

Manual

# Com-Server Mini/Industry

**Preliminary**

Unfortunately the English manual for our Com Server Highspeed is being worked on presently. Since the structure of the menu and the configuration of the Mini Com Servers described here is very similar, please use it until we have finished the manual.

As standard the Com Server 58631 is equipped with a serial interface which can be switched between RS232, RS422 and RS485 mode. Pinout and description of the DIP switches can be found on the last pages of this document.

**Rel. 1.22, Sept. 2001**

**Models:**

58211, 58411,  
58221, 58421,  
58611, 58621,  
58412

**Firmware:**

Valid as of 1.22

## Introduction

The Com-Server models 58211, 58411, 58221, 58421, 58611 and 58621 represent a uniform platform for linking serial interfaces such as RS232, RS422/485 to TCP/IP networks.

In addition to all the standard applications implemented in the Com-Servers, this reference manual also describes methods of integrating Com-Servers into your own applications.

### Com-Server Models

Model No.	Network interface	Supply voltage	Serial interface	Housing
58211	10BaseT	5V +/-5% AC adapter included	RS232 (optional RS422/485)	Top hat rail housing
58221	10/100BaseT autosensing	5V +/-5% AC adapter included	RS232 (optional RS422/485)	Top hat rail housing
58611	10BaseT	12-24V AC/DC	RS232 (optional RS422/485)	Top hat rail housing
58621	10/100BaseT autosensing	12-24V AC/DC	RS232 (optional RS422/485)	Top hat rail housing
58411	10BaseT	5V +/-5%	TTL (optional RS232, RS422/485)	none
58421	10/100BaseT autosensing	5V +/-5%	TTL (optional RS232, RS422/485)	none
58412	10BaseT	5V +/-5%	TTL and RS485	none (OEM, credit card format)

### Wiesemann & Theis GmbH, Wuppertal

Telephone: +49 202 2680-110, Fax: +49 202 2680-265

Microsoft, MS-DOS, Windows, Winsock, and Visual Basic are registered trademarks of Microsoft Corporation

Specifications subject to change and error

W&T 03/2001

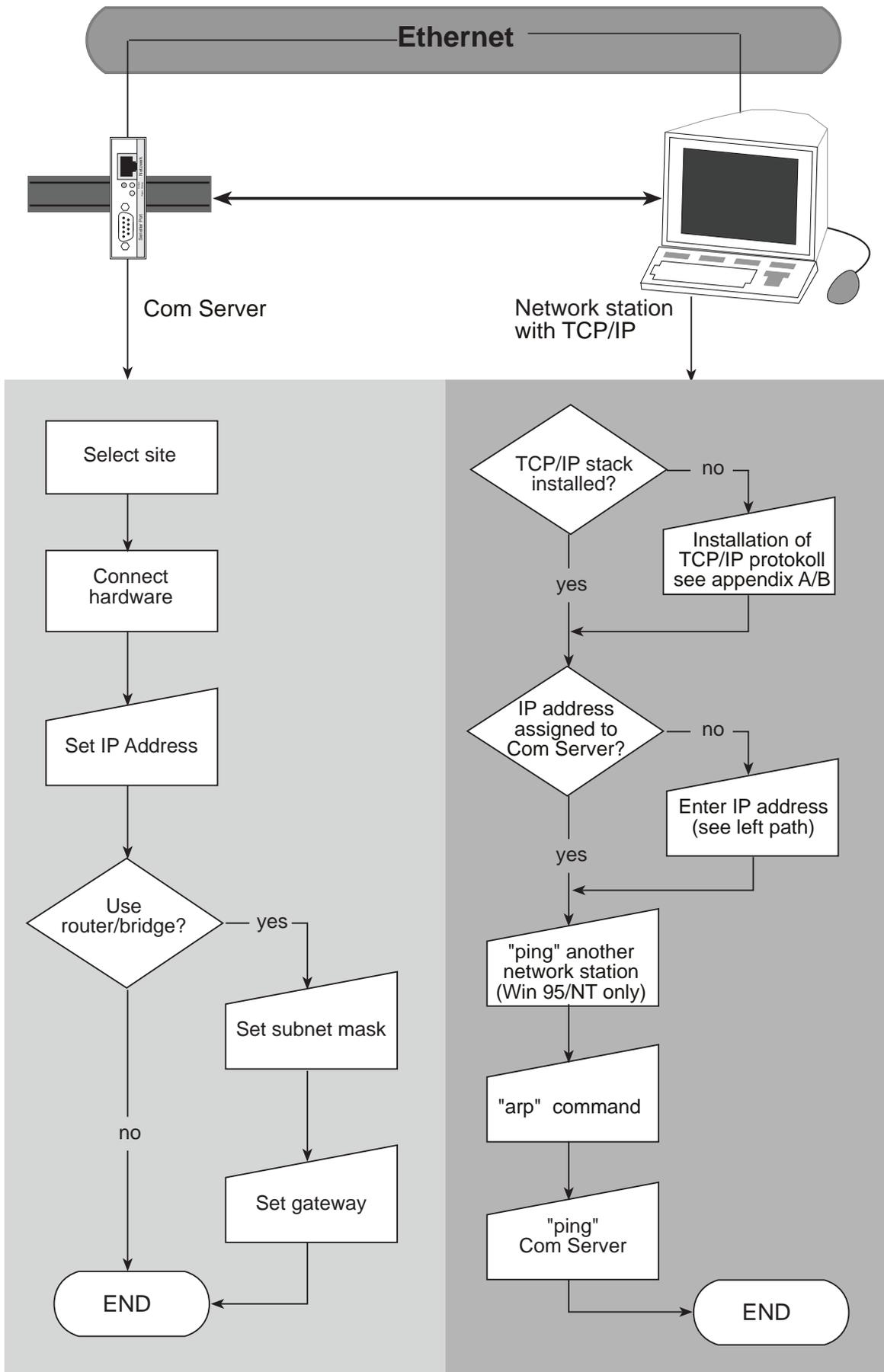
N:\Anleitung\58\_CS4\_5\Handbuch Rel 1\_21\english

---

<b>Introduction</b> .....	2
Com-Server Models .....	2
Quick Start .....	6
<b>1 Assigning the IP address</b> .....	7
1.1 Assigning using the "ARP" command (Windows 95/98/NT, UNIX) .....	7
1.2 Assigning through the serial port .....	8
1.3 Using the RARP server (UNIX) .....	9
1.4 Using DHCP-/BOOTP protocol .....	10
<b>2 Function of the Com-Server</b> .....	12
<b>3 Form factors</b> .....	13
3.1 Industry model .....	13
3.2 OEM model .....	14
3.3 OEM model 58412 (credit card format) .....	15
<b>4 Connections, controls and displays</b> .....	16
4.1 Ethernet connection .....	16
4.2 Serial connection .....	17
4.2.1 RS232 port .....	17
4.2.2 Serial TTL interface .....	18
4.2.3 RS422/485 interface (optional) .....	19
4.2.4 Interfaces for the OEM Com Server 58412 .....	22
4.3 Supply voltage .....	24
4.3.1 Supply voltage 5V (Models 58211, 58411, 58221, 58421) .....	24
4.3.2 Supply voltage 12–24V (Models 58611, 58621) .....	24
4.3.3 Power supply OEM model 58412 (credit card format) .....	24
4.4 LED displays .....	25
<b>5 Starting the Telnet configuration</b> .....	26
<b>6 Base configuration of the Com-Server</b> .....	27
6.1 Menu: INFO System .....	28
6.2 Menu: SAVE Setup → non-volatile memory .....	28
6.3 Menu: SETUP System .....	29
6.3.1 Menu: SETUP System → Setup TCP/IP .....	29
6.3.2 Menu: SETUP System → Set Password .....	30
6.3.3 Menu: SETUP System → Flash Update .....	30
6.3.4 Menu: SETUP System → Factory Defaults .....	30
6.3.5 Menu: SETUP System → Reset Com-Server .....	30

<b>7</b>	<b>The serial port as seen by the network</b> .....	31
7.1	Addressing in the TCP/IP network .....	32
<b>8</b>	<b>Configuration of the serial port</b> .....	33
8.1	The configuration menu .....	33
8.2	The serial parameters (Menu: UART Setup) .....	35
8.2.1	The handshake modes (SETUP Port 0 → UART Setup → Handshake) .....	35
<b>9</b>	<b>Configuring the TCP/IP modes (Menu: TCP/IP Mode)</b> .....	38
9.1	Configuring the TCP port number of the Com-Server .....	38
9.2	Data transfer per TCP/IP Sockets .....	39
9.2.1	TCP server mode (Default-Port number 8000) .....	39
9.2.2	TCP client mode (Menu: TCP Client) .....	39
9.2.3	Client/Server mode between Com-Server ports .....	41
9.2.4	UDP client mode (Menu: UDP Client)	
9.2.5	Setting the local UDP-Port number of the Com-Server .....	43
9.3	Serial socket interface .....	44
9.4	Data transfer per Telnet .....	45
9.4.1	Telnet server mode (Port number 23 or 6000) .....	45
9.4.2	Telnet client mode (Menu: Telnet-Client) .....	45
9.5	Data transfer per FTP .....	47
9.5.1	FTP server mode (Port number 21 or 7000) .....	47
9.5.2	FTP client mode (Menu: FTP-Client) .....	47
9.6	„Box to Box“ mode (Menu: Box to Box) .....	51
9.7	IP Bus mode (Menu: IP Bus Mode) .....	53
9.8	The Windows COM Port Redirector .....	54
9.9	SLIP router mode (Menu: SLIP Router) .....	55
9.9.1	Configuring the Com-Server via SLIP .....	56
<b>10</b>	<b>Status and error messages (Menu: Port State)</b> .....	58
<b>11</b>	<b>Expanded services of the Com-Server</b> .....	60
11.1	Control and service ports .....	60
11.2	Serial Port Control (Port number 9094) .....	60
11.2.1	The control structure .....	60
11.3	Reset Com-Server port (Port number 9084) .....	65
11.4	Software reset of the Com-Server (Port number 8888) .....	65
11.5	Uploading/downloading configuration data .....	66
11.5.1	Uploading/downloading the configuration under Windows .....	66
11.6	Inventory taking per UDP .....	68
11.6.1	The Info packet .....	68
11.7	SNMP management .....	70

<b>12</b>	<b>Firmware update of the Com-Server</b> .....	71
12.1	Where do I get the current firmware? .....	71
12.2	Network firmware update under Windows 9x/NT/2000 .....	72
12.2.1	Customer-specific factory defaults .....	72
12.3	Network firmware update under UNIX .....	74
12.4	Firmware update through the serial port .....	75
<b>Appendix</b>	.....	76
A	TCP/IP under Windows 9x .....	76
B	TCP/IP under Windows NT .....	77
C	Overview of the Port and Socket numbers factory set in the Com-Server .....	78
D	Serial assignment of the IP address under Windows .....	79
E	Web applications HTTP, SMTP, POP3 ... ..	81
F	Technical Data .....	82
<b>Index</b>	.....	83



## 1 Assigning the IP address

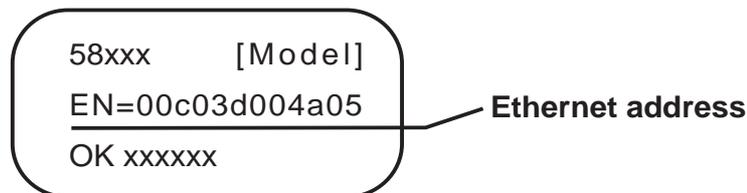
The Com-Server is factory set to IP address 0.0.0.0. Before you can make the entry in the Com-Server, you need to specify an IP address that is valid for your network. Your system administrator will provide you with this. If you have only a small network with no routing, use the IP address of your PC and simply change the last digit.

! *The IP address must be unique within the network!*

### 1.1 Assigning using the "ARP" command (Windows 95/98/NT, UNIX)

! *This method can only be used if the Com-Server does not already have an IP address, i.e. the entry is 0.0.0.0. To change an IP address, use one of the other methods described in this section or use the configuration menu over Telnet.*

Required is a computer which is located in the same network segment as the Com-Server and which has TCP/IP protocol installed. Read off the Ethernet address of the Com-Server from the sticker on the side of the housing:



Under Windows, first *ping* an other network station and then insert a static entry into the ARP table of the computer using the following command line:

```
arp -s [IP address] [MAC address]
```

e.g. under Windows: `arp -s 172.16.231.10 00-C0-3D-00-12-FF`

e.g. under SCO UNIX: `arp -s 172.16.231.10 00:C0:3D:00:12:FF`

Next use the following command line under „*Start* → *Run*“ to start a Telnet session on the configuration port of the Com-Server with the desired IP address:

```
telnet 172.16.232.10 1111 [Return]
```

The Com-Server takes the IP address of the first network packet sent to it as its own and saves it in non-volatile memory. The Telnet connection is not established and the configuration menu displayed in the Telnet window. All further settings are made here (see „Basic configuration of the Com-Server“).

! *The IP addresses must be **without** leading zeros in all Windows environments. Otherwise the entry is incorrectly interpreted by the system and an incorrect IP address is assigned to the Com-Server.*

## 1.2 Assigning through the serial port

After a Com-Server reset a time window of around 1-2 seconds is available, during which you can assign a new IP address by entering at least 3 „x“.

! *In contrast to the ARP method described above, this serial method functions regardless of whether the Com-Server already has an IP address or not. The procedure can be repeated as often as desired. Therefore use this method if you don't know the IP address or have forgotten it. Appendix D contains the detailed procedure under Windows using HyperTerminal.*

First connect the serial port of the Com-Server to a computer. For a standard PC or laptop, you will need a crossed RS232 cable (=Null modem cable, see „RS232 interface“).

The serial transmission parameters of the terminal program you use should be set to *9600 baud, no parity, 8 bits, 1 stop bit, no handshake*. Reset the Com-Server by interrupting the power. When the green status LED lights up, enter the letter „x“ at least three times on the terminal, until the Com-Server returns the prompt „IPno.+<Enter>“.

Use the usual format (xxx.xxx.xxx.xxx) to enter the IP address, and end the entry by pressing <Enter>. If the entry was accepted, the acknowledgement is the assigned IP address. Otherwise you will get a „FAIL“ message followed by the last current IP address.

All other settings such as gateway address, subnet mask etc. are done through the Telnet configuration menu (see „Basic configuration of the Com-Server“).

### Deactivating the DHCP-/BOOTP client through the serial port

The DHCP-/BOOTP function of the Com Server can be turned off as part of assigning the IP address serially. We recommend making use of this at all times except where use of DHCP/BOOTP is expressly required for the IP assignment.

To deactivate the DHCP/BOOTP client enter the option „0“ directly appended (no space!) to the IP address and confirm with <Enter>.

Example:

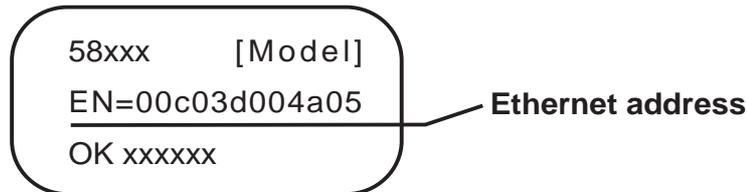
```
xxx                -> Com-Server
IP no.+<ENTER>:   <- Com-Server
172.17.231.99-0   -> Com-Server
172.17.231.99    <- Com-Server
```

**This function can later be reactivated through the Telnet configuration under „SETUP System → Setup TCP/IP ⊃ BOOTP Client (see section Basic configuration of the Com Server).**

### 1.3 Using the RARP server (UNIX)

UNIX environments especially use RARP protocol for centrally assigning IP addresses. TCP/IP devices that want to obtain an IP address send RARP requests with their Ethernet address as a broadcast over the network.

Activate the RARP server, and enter the Ethernet address of the Com-Server in the file */etc/ethers* and the IP address in the file */etc/hosts*.



The Com-Server must be connected to the network in the same segment as the RARP server.

**Example:**

Your Com-Server has the MAC address EN= 00C03D0012FF (sticker on the unit). You want to give it IP address 172.16.231.10 and the alias name WT\_1:

Entry in the file */etc/hosts*:           172.16.231.10           WT\_1  
 Entry in the file */etc/ethers*:       00:C0:3D:00:12:FF       WT\_1

If the Com-Server IP address is 0.0.0.0 (=factory default), RARP broadcasts are cyclically generated in order to obtain a valid address from any present RARP Daemon.

If the Com-Server already has a valid IP address, an RARP broadcast is generated after each reset. If a reply comes within 500 ms, the IP address contained in it is accepted. As in the case of assigning through the serial port, this method also allows you to overwrite a current IP address.

## 1.4 Using DHCP-/BOOTP protocol

Many networks use DHCP (**D**ynamic **H**ost **C**onfiguration **P**rotocol) or BOOTP for centralized, dynamic assigning of IP addresses. Which of the two protocols are used in any given situation makes no difference in connection with Com-Servers, since DHCP is simply a downward-compatible extension of BOOTP. DHCP servers thus also use requests from BOOTP clients. The following parameters can be assigned to the Com-Server using these protocols:

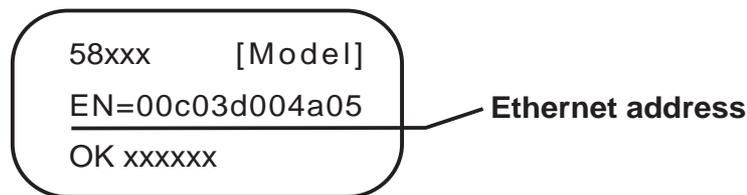
- **IP address**
- **Subnet mask**
- **Gateway address**

It is not possible to transfer other parameters or a lease-time.

### Function

To obtain an IP address, the Com-Server sends a corresponding BOOTP request as a broadcast over the network after each reset. The reply generated by the DHCP/BOOTP server contains the IP address as well as the subnet mask and gateway address. The Com-Server immediately places this information in its non-volatile memory.

To start up the Com-Server in DHCP/BOOTP networks, consult your system administrator. If DHCP is assigning the address, inform him that a reserved IP address is required. For maintaining the respective address database, the administrator needs the Ethernet address of the Com-Server found on the sticker:



Once the administrator has made the necessary entries, the Com-Server obtains the desired IP address automatically after each reset. To ensure accessibility of the Com-Server even should the DHCP/BOOTP server go down, the previous IP address is retained should there be no reply.

- ! In DHCP environments the assigned address **must** be reserved by means of a fixed link to the Com-Server's Ethernet address. Under Windows NT this is done in the DHCP Manager under menu item „Reservations“. Linux does this by providing the file „DHCPconf“ into which a corresponding entry has to be made.

**Deactivating DHCP-/BOOTP protocol**

A DHCP sever assigns IP addresses dynamically from an address pool provided by the administrator. This means DHCP-compatible devices usually receive a different IP address each time they are started. Since a constantly changing IP address with the Com Server is not desirable, the latter uses BOOTP protocol based on fixed assignments from Ethernet to IP address. DHCP servers should reply to BOOTP requests only if they have an explicit IP reservation for the Ethernet address of the sender.

Some DHCP servers (e.g. Windows 2000 servers) do however use both DHCP and BOOTP requests from the dynamic address pool. To prevent the Com Server from assigning the user unknown IP addresses in such environments, the following methods can be used:

- A reservation in the respective DHCP server must be made before connecting the Com Server to the network.
- Assigning the IP address for the Com Server through the serial port. Sending “xxx” to the Com Server during a new start takes you to the input mode for a new IP address. If you enter this followed by the string “-0”, the BOOTP client of the Com Server is deactivated (see section *Assigning over the serial interface*).

**In existing systems the BOOTP client of the Com Server can also be deactivated or reactivated at any time over the Telnet configuration under „*SETUP System → Setup TCP/IP → BOOTP Client* (see section *Base configuration of the Com Server*).**

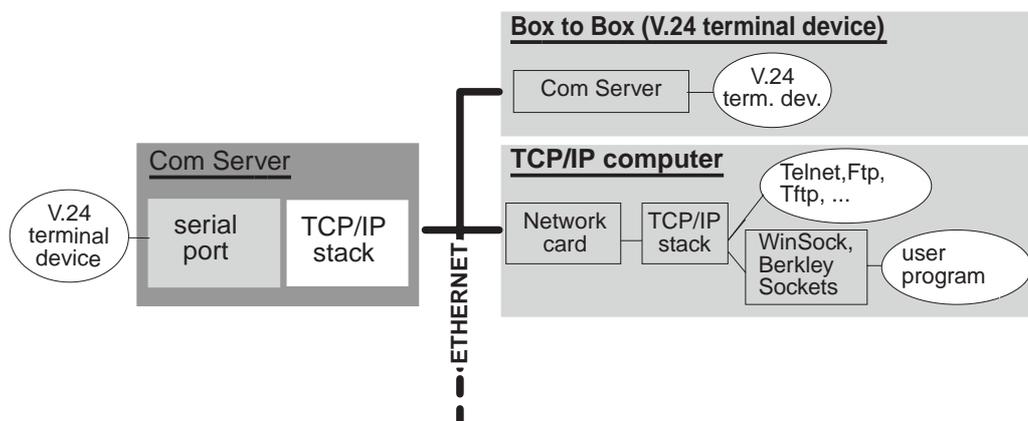
➡ *An explanation of the basic terms and concepts for addressing in the Internet and using DHCP and BOOTP can be found in our manual „TCP/IP for Beginners“.*

## 2 Function of the Com-Server

Com-Servers allow direct communication with various serial devices over an Ethernet network regardless of the computer world.

Measuring devices, controls and peripherals of all kinds having a serial port can be easily accessed through an Ethernet LAN. The connections are made through the existing cable without affecting any network operation which may be already running; repeaters, bridges or routers can also be used.

On the serial side, Com-Servers can be equipped with RS422 or RS485 in addition to the standard RS232 interface.



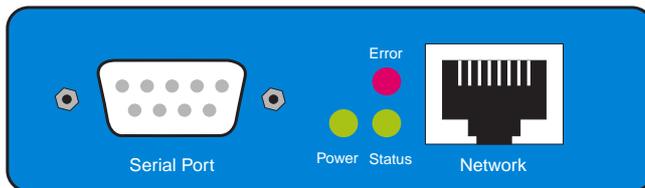
The Com-Server has a configuration port that is opened through the network using the *telnet* command. This offers diagnostic possibilities in addition to setting of the base and operating parameters.

The Com-Server is basically accessible using TCP/IP protocol. Which higher protocols in the TCP/IP suite (e.g. Telnet, FTP etc.) are used for data transmission depends on the actual application.

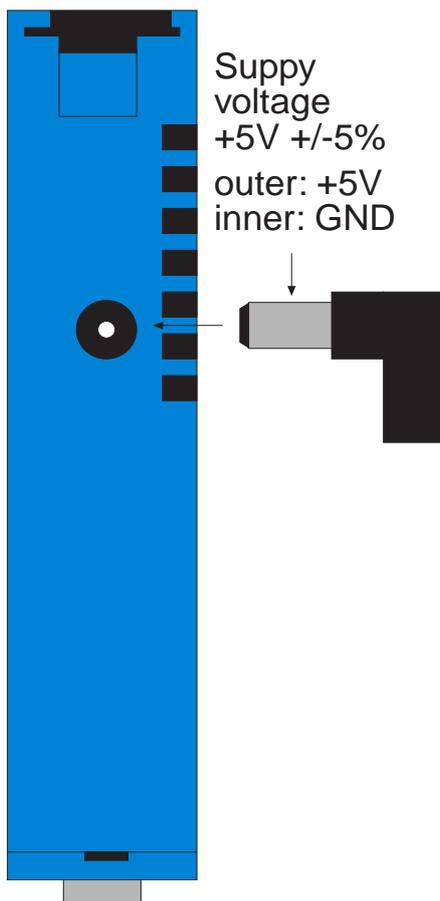
### 3 Form factors

#### 3.1 Industry model

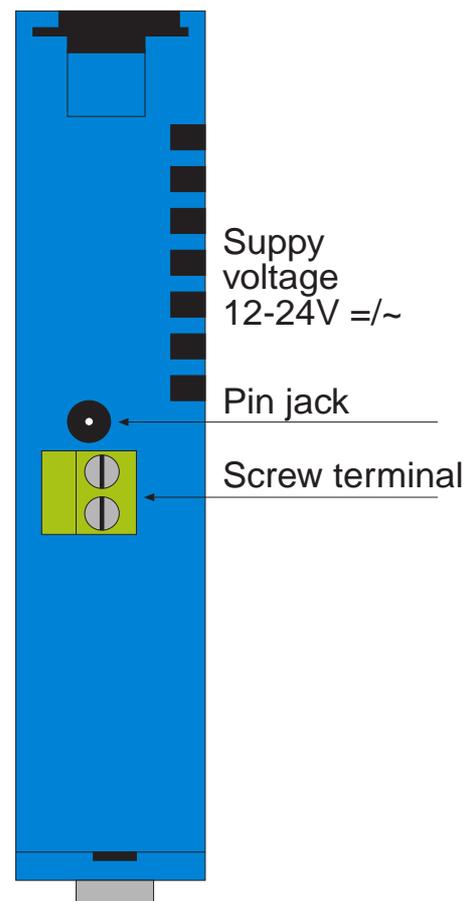
Front view 58211, 58221, 58611, 58621



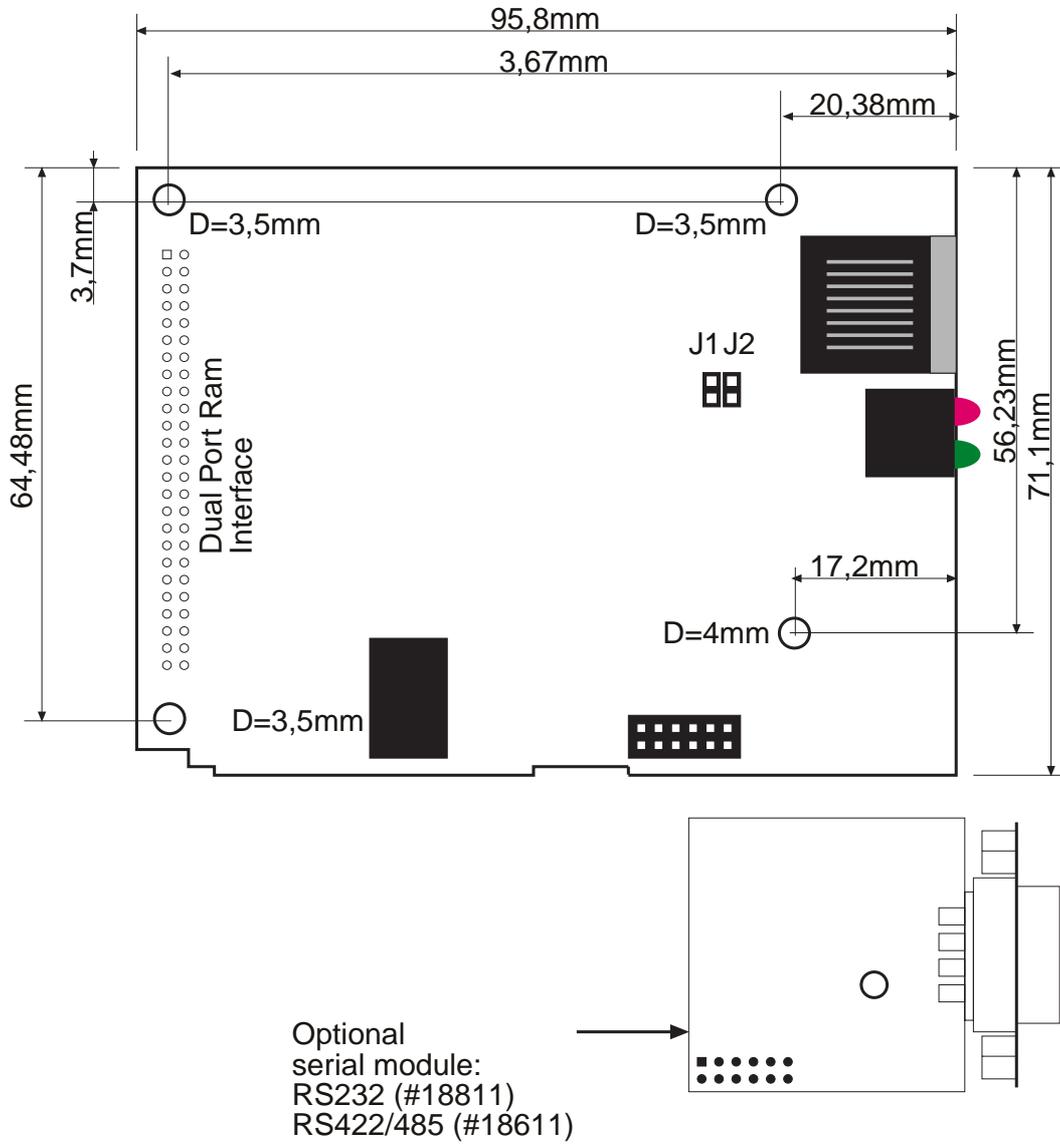
Bottom side 58211



Bottom side 24V models

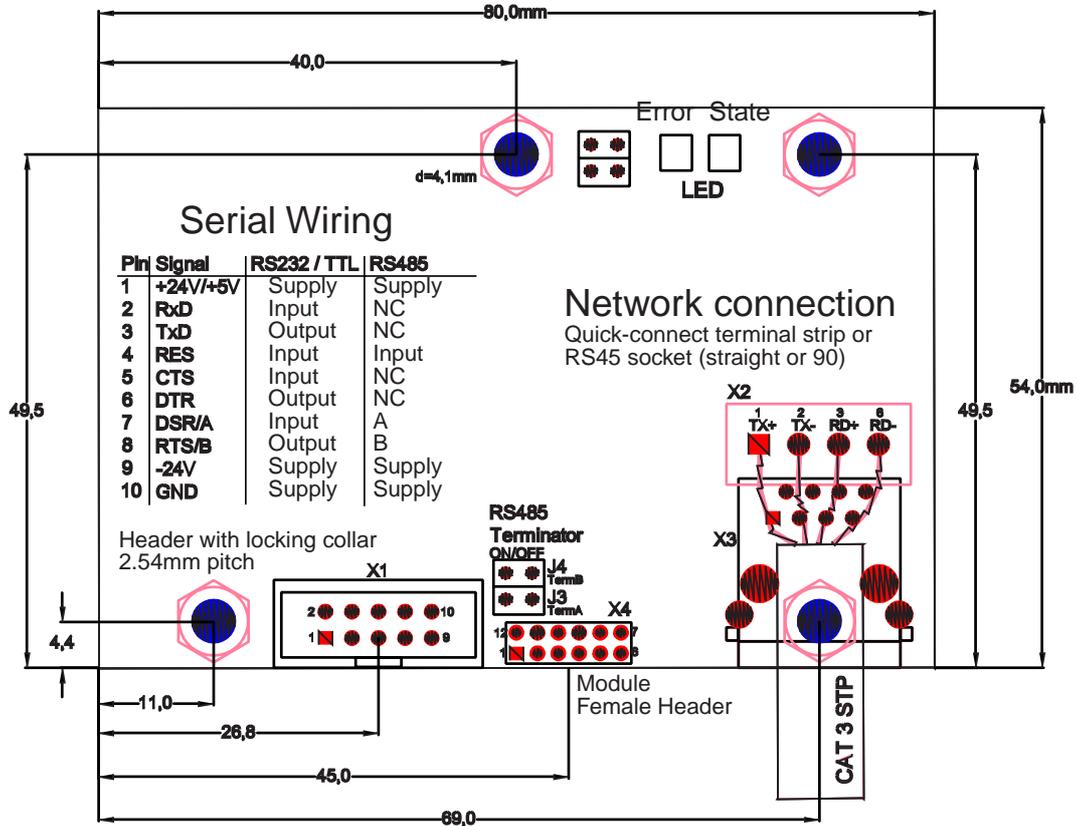


3.2 OEM model



### 3.3 OEM model 58412 (credit card format)

The OEM Com Server 58412 has both the standard serial interface with TTL levels as well as an onboard RS485. For a detailed description of these interfaces, see section Connections, operating elements and displays.



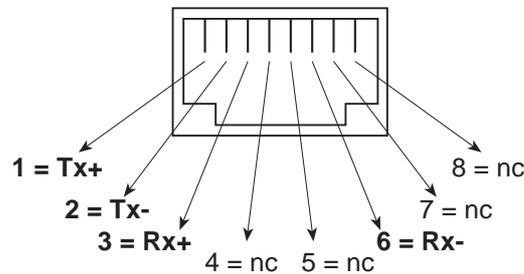
## 4 Connections, controls and displays

The Com-Server should be installed in a location such that a cable length of 100 meters on the network side is not exceeded.

Please note that all plugs may be connected except for turned-off terminal devices. The location of the individual connections can be found in illustrations in the „Form factors“ section.

### 4.1 Ethernet connection

For the network connection an IEEE 802.3-compatible terminal on a shielded RS45 connector on the front of the unit is provided. Here the Com-Server can be connected to a hub or switch. The wiring corresponds to a conformal MDI interface (AT&T258), so that you can use a 1:1 cable with a length of max. 100 meters.



RJ45 female (per AT&T256)

Depending on the Com-Server model the following standards are supported:

#### **10BaseT, 10 MBit/s** (Models 58211, 58411, 58611)

The units conform to 10BaseT at 10 Mbps. It is also possible however to use an autosensing hub or switch to incorporate into a 100BaseTx network. Such autosensing components automatically set themselves to the transmission speeds supported by the terminal device.

#### **100BaseTx and 10BaseT combined, 100/10 Mbps** (Models 58221, 58421, 58621)

These units support both 10BaseT as well as 100BaseTx at a bit rate of 100 Mbps and the option of full-duplex transmission. Switching between the two network speeds is automatic using the autosensing function of the Com-Server, corresponding to the possibilities of the hub or switch. Operating at 100 Mbps requires the use of suitable cable, at least conforming to Cat. 5/ISO Class D.

The current link status is indicated by the Error LED on the front panel. If this LED is flashing at a rate of approx. 1 second (at least once), the connection to the hub either does not exist or is experiencing a disturbance.

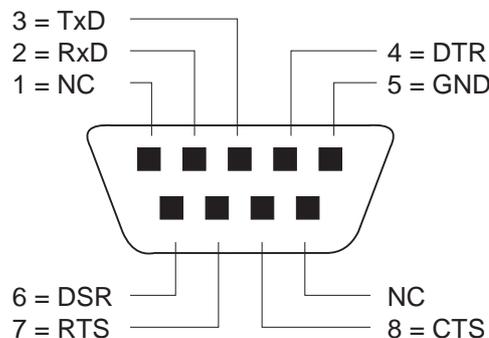
## 4.2 Serial connection

Model 58211, 58221, 58611 and 58621 Com Servers come standard with a serial RS232 interface. The modular design also allows use of an RS422/485 board. The OEM models 58411, 58421 and 58412 are factory supplied with serial TTL interfaces. The OEM model 58412 also provides an additional onboard RS485 interface.

### 4.2.1 RS232 port

The pin configuration of the RS232 port is identical to that of a PC, which allows you to use standard cables. Be sure that the Com-Server port and that on the serial terminal device are configured for the identical transmission parameters and handshake procedure.

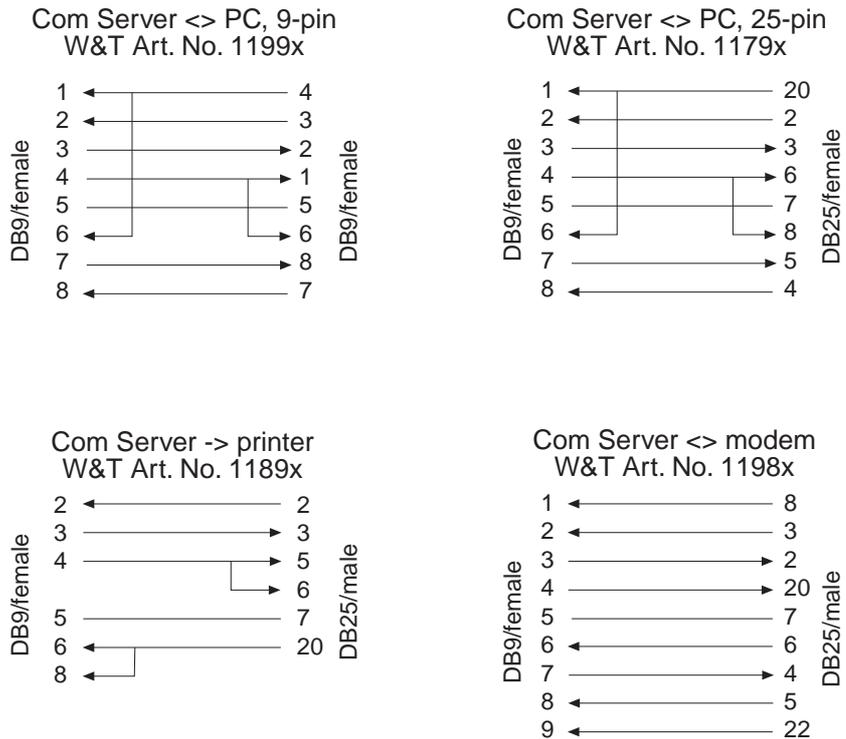
A detailed pin description as well as the required W&T cable for the main applications can be found on the following page.



Pin assignment for RS232, DB9 plug

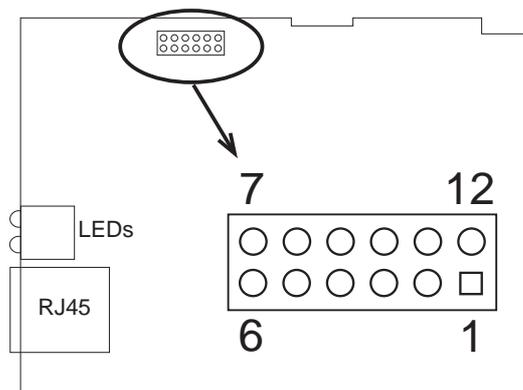
Pin	Direction	Signal	Description	Default funktion (1)
1	—	NC	Not connected	—
2	IN	RxD	Receive data	Data in
3	OUT	TxD	Transmit data	Data out
4	OUT	DTR	Data terminal ready	+12 V for existing TCP connection to a client or server
5	—	GND	Signal ground	—
6	IN	DSR	Data set ready	ignored
7	OUT	RTS	Ready to send	Handshake output: +12 V = ready to receive data -12 V = not ready to receive data
8	IN	CTS	Clear to send	Send data only at +3..12 V
9	—	NC	Not connected	—

(1) Applies only to the Hardware handshake setting



### 4.2.2 Serial TTL interface

The serial TTL interface on the OEM Com-Server is configured as a 12-pin post connector with 2mm spacing and permits direct connection to the TTL port of a UART. It is also the basis for using all W&T interface modules.



! *Wiring assignments for the OEM model 58412 „credit card format“ differ from those described here. For this model please consult section “Interfaces for the OEM Com Server 58412“.*

**Pin assignment and function TTL interface 58411 and 58421:**

TTL interface

Pin	Signal	Function
1	5V	Vcc
2	NC	—
3	RxD	Data in
4	TxD	Data out
5	Reset (1)	Reset in-/output
6	CTS	Input
7	DTR	Output
8	DSR	Input
9	RTS	Output
10	NC	—
11	NC	—
12	GND	Signal ground

- (1) The open collector feature of the Reset pin allows its use as a LOW-active in- and output. A Reset generated for example by the Com-Server's watchdog can be employed for your own uses. Applying a GND level for at least 100ms (by using a button for example) forces a reset of the Com-Server.

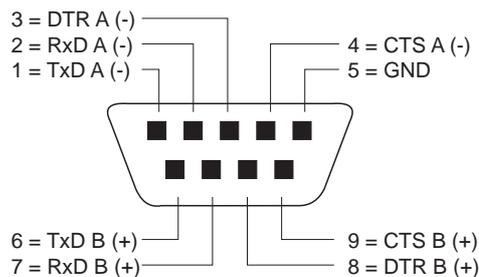
**4.2.3 RS422/485 interface (optional)**

(does not apply to model 58412; please consult section *Interfaces for the OEM Com Server 58412*)

As an option Com-Servers can be equipped with an RS422/485 interface. As opposed to RS232, the RS422 version allows transmission distances of up to 1000 meters. The RS485 mode allows you to integrate corresponding 2- or 4-wire bus systems into a TCP/IP network with the help of the Com-Server.

All RS422/485 signal lines are protected against static discharges of up to 15kV per IEC 801-2, Level 4.

! Especially when using longer cable lengths in industrial environments, potential differences need to be taken into consideration. To prevent the resulting transmission problems and hardware damage, we recommend the use of separate galvanic isolation, using for example the RS422/485 isolator Model 66201.



Pin assignment RS422/485, DB9 male

### Modes

The RS422/485 interface module can be set to three different modes using the DIP switch located on the module:

- **RS422, RS485 4-wire bus master**  
Provides one data and handshake channel each in each direction. The RS422/485 drivers and receivers are always active in this mode.
- **RS485 4-wire with automatic control**  
One data channel is available in each direction. The RS485 driver chip is automatically activated each time data is sent out and then disabled when data is finished sending. The receive channel is always active in this mode.
- **RS485 2-wire bus with automatic control**  
One data channel is available in each direction. The RS485 driver chip is automatically activated each time data is sent out and then disabled when data is finished sending. The receive channel is deactivated when the driver is on, and on when the driver is disabled.

See the following table for DIP switch mode selections:

Mode	SW1	SW2	SW3	SW4	SW5
RS422, RS485 4-wire bus master	OFF	OFF	OFF	OFF	ON
RS485 4-wire automatic control	OFF	ON	OFF	OFF	ON
RS485 2-wire automatic control	ON	ON	OFF	OFF	ON

### Opening the Com-Server

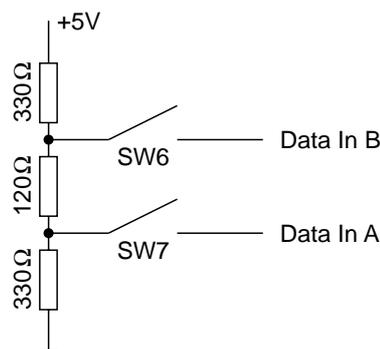
To open the Com-Server, plug the DB9 connector into the seriali port. After tightening the two mounting screws, pull on the DB9 plug to remove the circuit board from the housing.

### Handshake for RS485 modes

RS485 bus systems do not use flow control in the traditional sense for data integrity, but rather a logical protocol. The handshake procedure for the Com-Server must therefore be set to *NO* (see „The serial parameters (Menu: UART Setup)).

### Terminating

All RS485 modes require a termination network on the bus ssystem which ensures a defined rest state in the high-ohmic phases of bus operation. Connection of the bus system with a termination network can be done in the interface by closing the DIP switches 6 and 7 on the RS422/485 module:



**Wiring examples**

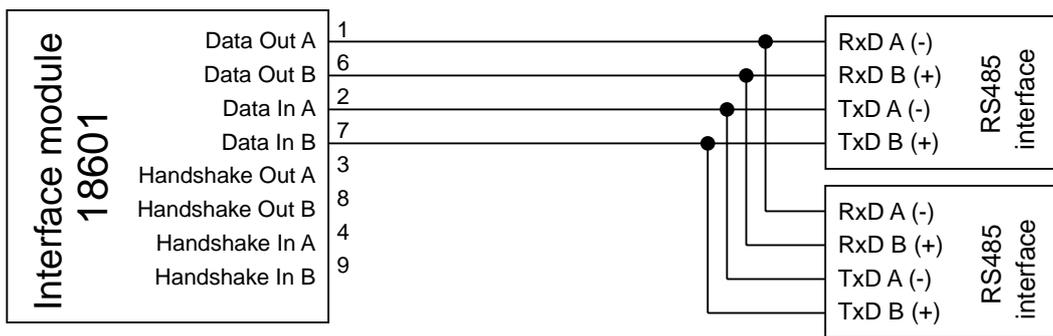
**RS422 connection with hardware handshake**

RS422/485 interface



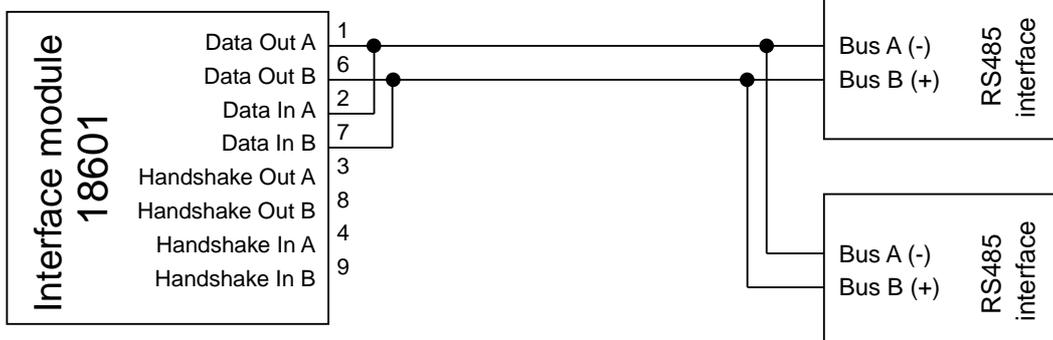
**RS485 connection (4-wire bus master)**

RS422/485 interface



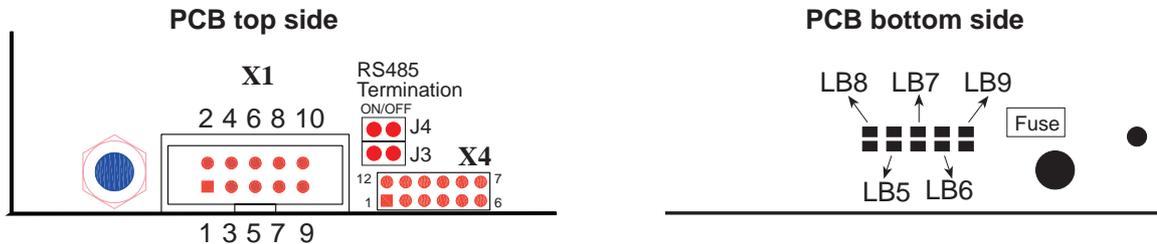
**RS485 2-wire connection**

RS422/485 interface



#### 4.2.4 Interfaces for the OEM Com Server 58412

The OEM model 58412 in credit card format has in addition to an RS232 interface with TTL levels an onboard RS485 interface. Both interfaces are located on the 10-pin connector X1 with the locking collar. Reconfigure using the solder jumpers on the underside of the board.



#### Pin assignments X1

Pin	Signal	Function	
		RS232/TTL	RS485
1	+5V/+24V	Supply	Supply
2	RxD	Data In	(RxD)
3	TxD	Data Out	(TxD)
4	RES/ <sup>(1)</sup>	Reset Input/Output	Reset Input/output
5	CTS	Data In	(Data In)
6	DTR	Data Out	(Data Out)
7	DSR/A	Data In	Bus A/-
8	RTS/B	Data Out	Bus B/+
9	GND 24V	Supply	Supply
10	GND 5V	Supply	Supply

<sup>(1)</sup> The open collector configuration of the reset pins allows function as a LOW-active in- and output. A reset triggered for example by the watchdog in the Com Server can be used here for your own purposes. Applying a GND level for at least 100ms (using a button for example) forces a restart of the Com Server.

#### The serial TTL interface

This interface can be configured as the TTL port of a UART. The solder jumpers must be set as follows:

- LB5, LB6 = in
- LB7, LB8, LB9 = out

#### The RS485 interface with automatic control

The following configuration of solder jumpers establishes a 2-wire RS485 bus connection on pins 7 and 8 of X4.

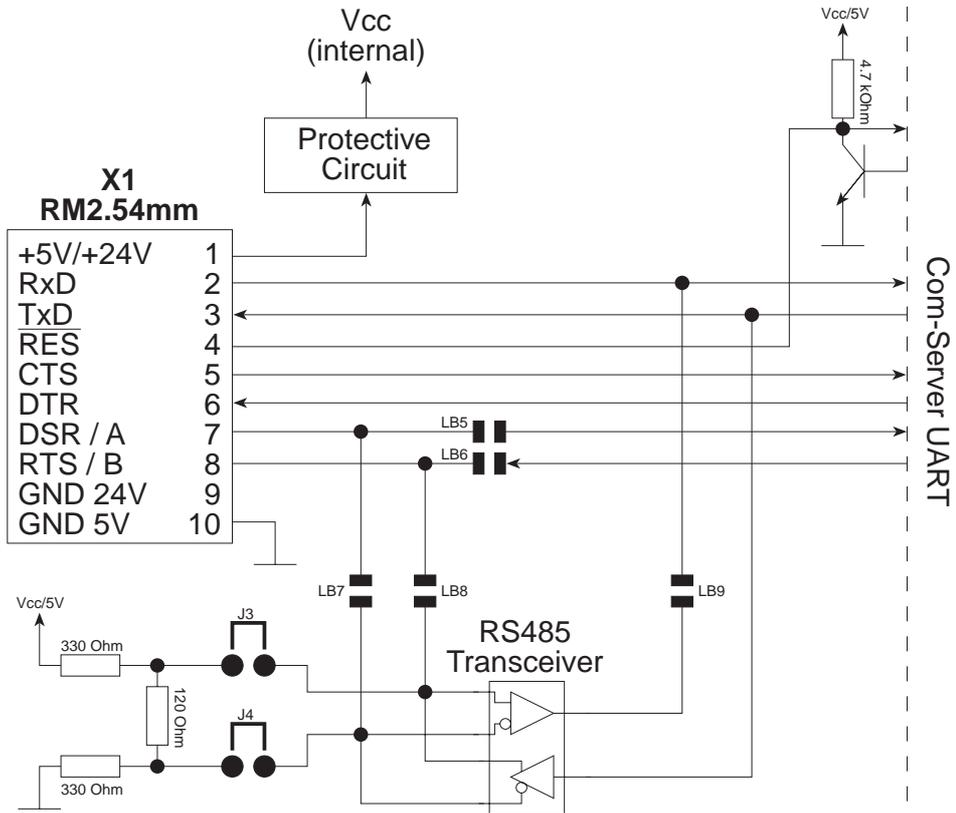
- LB5, LB6 = out
- LB7, LB8, LB9 = in

The RS485 driver ship is automatically enabled each time data is sent and disabled (high impedance state) again when data output is finished. The RS485 receiving channel is deactivated when the driver is on, but is switched on when the driver is in the high impedance state.

RS485 mode requires that the bus system be terminated with a termination network which ensures a defined idle state in the high impedance phases of bus mode. If the bus system in question does not already have a termination, one can be set using (inserting) jumpers J3 and J4.

! *The serial TTL interface and the RS485 bus can be used only in alternation.*

**Basic diagram of the serial interfaces**



### 4.3 Supply voltage

Depending on the model, Com-Servers are powered either by a regulated +5V or with an AC/DC voltage between 12V and 24V.

#### 4.3.1 Supply voltage 5 V (Models 58211, 58411, 58221, 58421)

The supply voltage for models 58211, 58411, 58221 and 58421 is brought in through the mains socket located at the bottom of the housing or on the circuit board edge. The feed voltage must be 5V +/-5%. The current draw of each model is indicated in the technical appendix.

#### 4.3.2 Supply voltage 12–24 V (Models 58611, 58621)

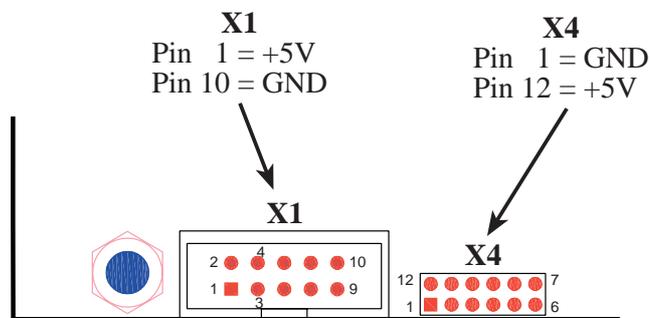
The supply voltage for models 58611 and 58621 can also be brought in through the mains socket for jack plugs located at the bottom of the housing or on the circuit board edge. DC voltage of any polarity or AC voltage may be used. Polarity reversal protection results in the following maximum and minimum values for the supply voltage:

- AC: 9Veff (- 5%) - 24Veff (+5%)
- DC: 12V (-5%) - 34V (+5%)

The current draw of each model is indicated in the technical appendix.

#### 4.3.3 Power supply OEM model 58412 (credit card format)

The supply voltage for OEM model 58412 is brought in through one of the two post connectors that are used for the serial signals as well. Supply voltage has to be 5V +/-5%. The current draw is indicated in the technical appendix.



## 4.4 LED displays

- **Power LED**  
Indicates supply voltage present. If the LED is not on, check for correct wiring of the power supply.
- **Status LED**  
Flashes whenever there is activity on the serial port. Periodic flashing indicates that the port has a valid connection to another network station. The status of the serial port can also be read out using the Com-Server's Telnet configuration tool.
- **Error LED**  
The Error LED uses various flashing patterns to indicate error conditions on the device or serial port. The error texts for the previous five serial faults and the associated system time (time between the last restart of the Com-Server and when the error occurred) can also be read out using the Telnet configuration tool.

**1 x flashing of the Error LED** = Check network connection

The Com-Server cannot receive a link pulse from a hub. Check the cable or the hub port.

**2 x flashing of the Error LED** = Check serial data format

The serial port received at least one character with a parity or framing error (=Parity-Error / Framing-Error), or the data register of the serial receiver ship was written even though the previous character was not read out. Check the correctness of the serial parameters, the handshake procedure and the connection cable.

**3 x flashing of the Error LED** = Check serial Handshake

The serial connected device is not responding to the handshake stop signal set by the Com-Server and continues to send data. The result can be that the serial ring memory is overwritten and data are lost. Check the handshake configuration of the device as well as the wiring of the connection cable.

**All LEDs on** = Self-test error

The self-test performed after each start or reset of the Com-Server could not be correctly finished.

This error can occur when you have prematurely broken off a software update and the full operating software could not be transferred. The Com-Server is no longer capable of being operated in this condition. Repeat the software update over the network (see „Firmware update of the Com-Server“), and address the Com-Server using its assigned IP address. If this does not eliminate the error or should the error occur irrespective of any prior software update, please return the unit to W&T for service.

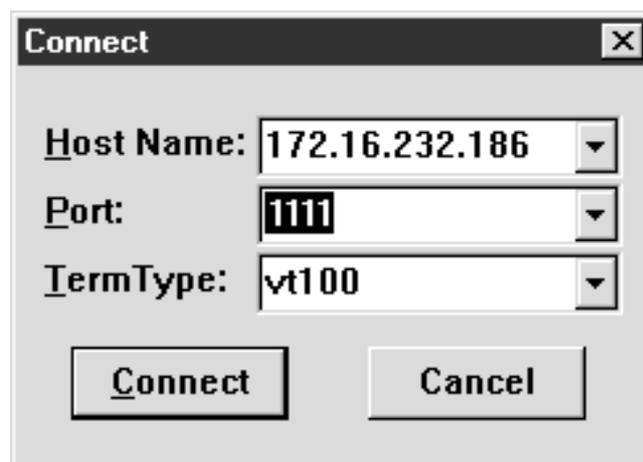
## 5 Starting the Telnet configuration

The Com-Server is configured using the *Telnet* network protocol. The prerequisite is that the Com-Server has already been assigned an IP address that is valid in this network at which it can be accessed (see "Assigning the IP address"). The configuration can be done from virtually any computer having a network connection and TCP/IP stack (see Appendix A/B). The *telnet.exe* program is always included with Windows 9x/NT/2000, and is usually located in the Windows main directory.

To select the configuration menu, start the Telnet client using the following parameters:

[Address] = IP address of the Com-Server  
1111 = Configuration port of the Com-Server

Under Windows 9x/NT you only need to use „Start → Run“ *telnet*. Then in the submenu *Network system...* of the *Connect* menu enter as the host name the IP address of the Com-Server and as the connection port number „1111“.



If the connection could be established, you will see the following menu in your Telnet window:

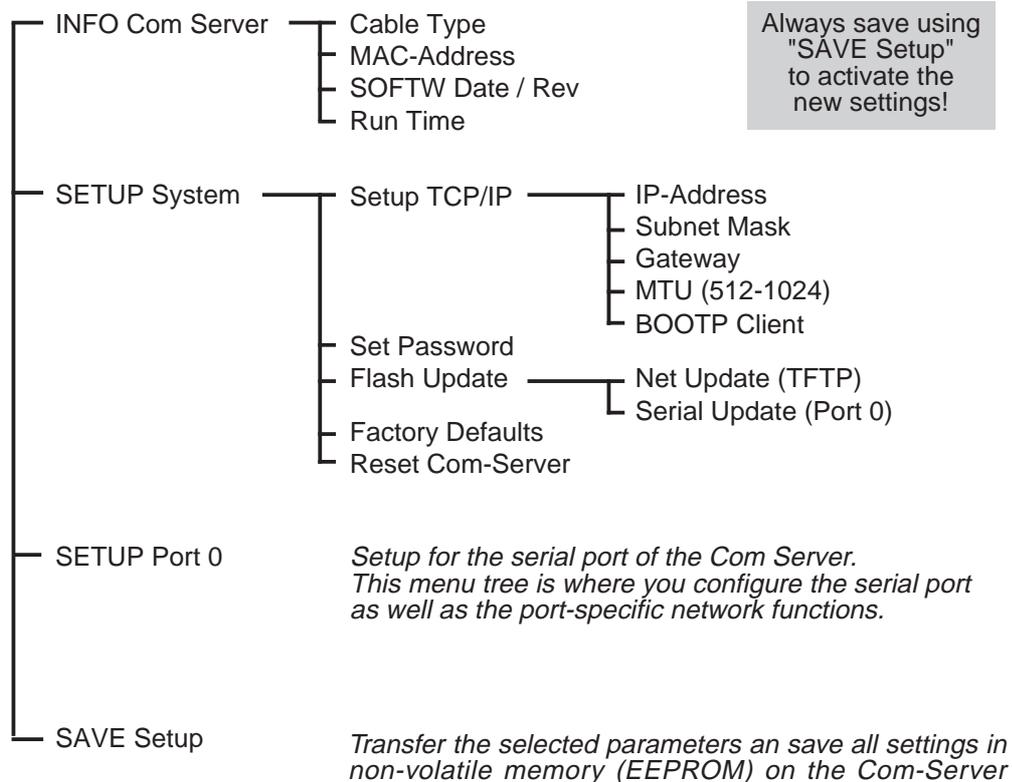
```
*****
*      MINI Com-Server      *
*****
1. INFO  System
2. SETUP System
3. SETUP Port 0 (Serial)
4. SAVE  Setup
```

## 6 Base configuration of the Com-Server

The following overview shows the configuration menu of the Com-Server. You will only see one level of the selected menu at a time on the monitor. To go to the next level, simply enter the number of the desired menu branch and press ENTER. Entering *q* or pressing ENTER brings you back to the previous menu level.

The last configured value of a menu item is shown in parentheses. If you make any changes, the new value will appear here the next time you open the menu. Entries do not become valid in the Com-Server until you have saved them using *SAVESetup*.

As long as you do not open this menu item, you can move around the entire menu and change values without really modifying anything.



- ! If you exit the configuration menu by closing the Telnet connection without first doing a SAVE Setup, the original configuration will be retained.

## 6.1 Menu: INFO System

This menu allows you to call up device-specific parameters such as the version number and creation date of the firmware, MAC address of the unit, etc.

- **Cable Type**  
Shows the Ethernet connection of the Com-Server (twisted pair)
- **MAC address**  
Shows the Ethernet address of the Com-Server. This number is factory set and registered. It cannot be changed.
- **SOFTW Date/Rev.**  
Shows the creation date and version number of the operating software in flash
- **HARDW Rev.**  
Shows the version status of the Com-Server hardware.
- **Run Time**  
Shows the time in hours and minutes since the last Com-Server restart.

## 6.2 Menu: SAVE Setup → non-volatile memory

Enter a y to the question "Save Changes?". If the entry was correct the message „Saving...“ appears on the monitor and the Com-Server saves all your changes in its non-volatile memory (EEPROM). Once the data are saved they are activated each time you turn the unit on or perform a reset.

If you enter something other than y or simply press the ENTER key, you are returned to the main menu without saving the changes.

## 6.3 Menu: SETUP System

This menu is for configuring all the parameters that pertain to the Com-Server operating system and are independent of the serial interface.

### 6.3.1 Menu: SETUP System → Setup TCP/IP

#### → IP-Address

Enter here the IP address if you want to change it. Please note that this number is not freely selectable, but rather needs to be specified based on the network address of the TCP/IP network. The format corresponds to the normal syntax (e.g. 172.16.231.5).

#### → Subnet Mask

The subnet mask only needs to be entered if the Com-Server will be making connections to another subnet. Enter the subnet mask of the subnet in which the Com-Server is located (e.g. 255.255.255.0). Please note: The IP address determines the network class. From this is derived a default subnet mask (e.g. 255.255.0.0 for a Class B network). You are only allowed to extend this to the right. Improper entries are automatically corrected when you save.

#### → Gateway

Enter here the IP address of the Gateway if you will be making connections to other subnets.

! *Changes to the system parameters „IP-Address“, „Subnet Mask“ and „Gateway“ cannot be activated right after saving. The Com-Server will use these values only after closing the current Telnet connection using „q“.*

#### → MTU – Maximum Transfer Unit (default: 512)

This value determines the maximum size of a TCP/IP packet. It refers to the number of bytes (excluding header) that can be sent in a packet. The smaller the MTU, the more network buffer overall is available in the Com-Server. The selectable range begins at 512 and ends at 1024 bytes. The values can be set in increments of 128 bytes (automatic correction).

#### → BOOTP Client

The factory setting is for DHCP-/BOOTP protocol activated (menu entry = 1) and the Com Server attempts to get an IP address after each reset of a DHCP or BOOTP server. Entering a “0” deactivates the function (see section Assigning the IP address using DHCP-/BOOTP protocol). Another method of deactivating the BOOTP client is in conjunction with IP assignment over the serial interface (see section Assigning over the serial interface).

! *Some DHCP servers also assign an IP address in response to BOOTP requests from their dynamic pool. To prevent the Com Server from getting a different IP address after every reset in such environments, you must deactivate the BOOTP protocol.*

➔ *For an explanation of basic terms and principles for addressing in the Internet, see our manual „TCP/IP for Beginners“.*

### 6.3.2 Menu: SETUP System → Set Password

Here you may specify an 8-character hexadecimal password for protecting the configuration menu from unauthorized access. The factory-set password is 00000000. With the password you have unrestricted access to the Com-Server's configuration menu.

You are prompted for a valid password as soon as the Telnet connection for the configuration port is opened. An incorrect entry denies access to the Com-Server's configuration port.

! *Resetting or changing any parameters including the password is only possible if you know the old password. Make sure you write it down and keep it in a safe location.*

### 6.3.3 Menu: SETUP System → Flash Update

Before you activate one of the following modes, make sure that you have quit any active network connections. Then confirm with „y“. The update mode is indicated by lighting of the Status LED on the Com-Server.

#### → Net Update (TFTP)

Activate this mode when you want to perform an update of the Com-Server operating software over the network using TFTP (see „Firmware update of the Com-Server“).

#### → Serial Update (Port 0)

Activate this mode if you want to send the files containing the new firmware serially. (see “Firmware update of the Com-Server“).

! *You may quit the update mode only by either completely performing the update or by means of a reset, i.e. turning off the power!*

### 6.3.4 Menu: SETUP System → Factory Defaults

Enter a y to reset all the parameters. The configuration is then in its factory preset state. The Com-Server closes the Telnet connection. Then it performs a software reset in order to activate the new configuration.

! *Resetting the non-volatile memory causes loss of **all** the settings which deviate from the default values, including the IP address.*

*The setting profile of the factory defaults can among other things be replaced by a customer-specific profile (see „Customer-specific configuration profiles“). In this case the customer-specific settings are activated after the reset.*

### 6.3.5 Menu: SETUP System → Reset Com-Server

Select this menu item to perform a software reset on the Com-Server. First your Telnet connection is properly closed.

! *All data from any still open network connections are lost!*

## 7 The serial port as seen by the network

The actual data are exchanged between the application and the Com-Server port using TCP/IP or UDP/IP. Which protocol is selected and the type of connection (Client or Server) for data transmission depends on the application. The following table shows the various modes of the serial Com-Server port.

### Menu: **SETUP Port 0** → **TCP/IP Mode**

"TCP-Server" Mode	–
"TELNET-Server" Mode	–
"FTP-Server" Mode	–
"TCP-Client" Mode	→ TCP-Client
"UDP-Client" Mode	→ UDP-Client
"TELNET-Client" Mode	→ Telnet-Client
"FTP-Client" Mode	→ FTP-Client
"Box to Box" Mode	→ Box to Box
"IP Bus Mode"	→ IP Bus Mode
SLIP-Router	→ SLIP Router
Serial Socket Interface	→ Ser. Socket Interface

Data transmission over the serial connection is full-duplex with a maximum speed of 57.6 K. Interface parameters and network transmission mode are configured over the network from any TCP/IP station.

In addition to simple transmission of the data, a dedicated TCP port provides functions for monitoring and configuring the serial interface (see „Enhanced services of the Com-Server“).

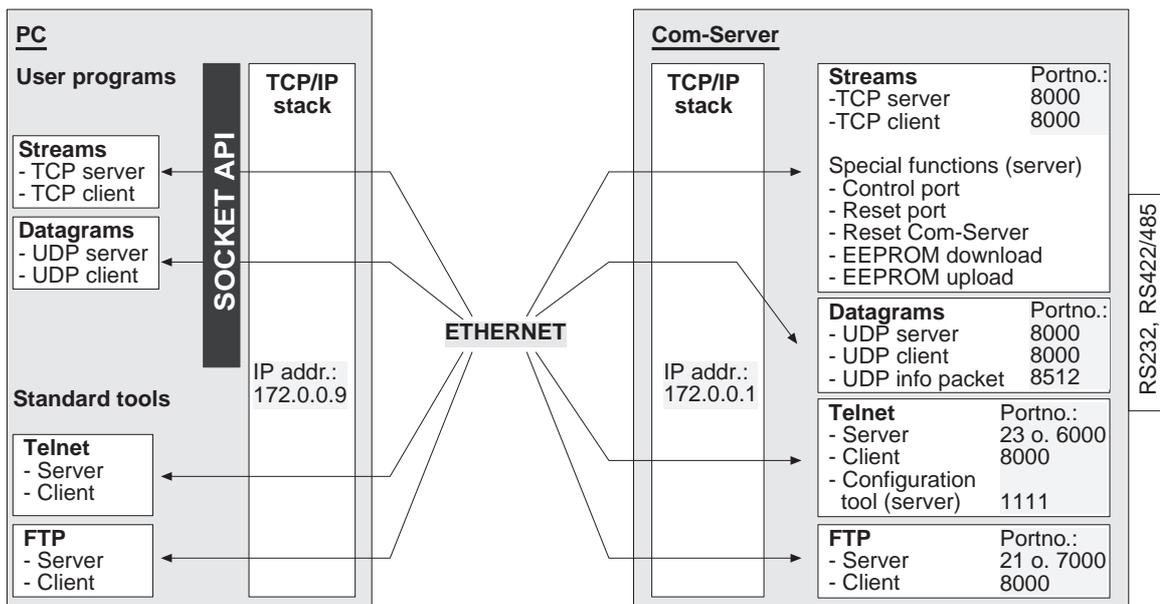
The Com-Server supports the following protocols:

- ARP, RARP, ICMP (Ping), (DHCP and SNMP in development)
- IP, TCP (Client/Server), UDP (Client/Server)
- FTP (Client/Server), TELNET (Client/Server)
- SLIP

## 7.1 Addressing in the TCP/IP network

Addressing in the TCP/IP network is done in two steps. First the network station itself is addressed with the IP address, then the services of this network station are addressed with TCP port numbers. Each IP address must be unique throughout the network, and each port number must be unique on the network station.

Addressing the serial ports is done analogously. The Com-Server is assigned an IP address upon installation (see „Assigning the IP address“). This address is used to access the Com-Server. The individual services are factory addressed using the following port numbers.



When the service starts a certain port number is assigned to a server process; the port numbers of client processes are normally dynamically assigned and change each time the program starts.

In the Com-Server all the processes (client and server) have factory assigned port numbers, except for the FTP client.

If in a particular case the factory assigned port number cannot be used, you may modify it (see „Configuration of the TCP/IP modes“).

## 8 Configuration of the serial port

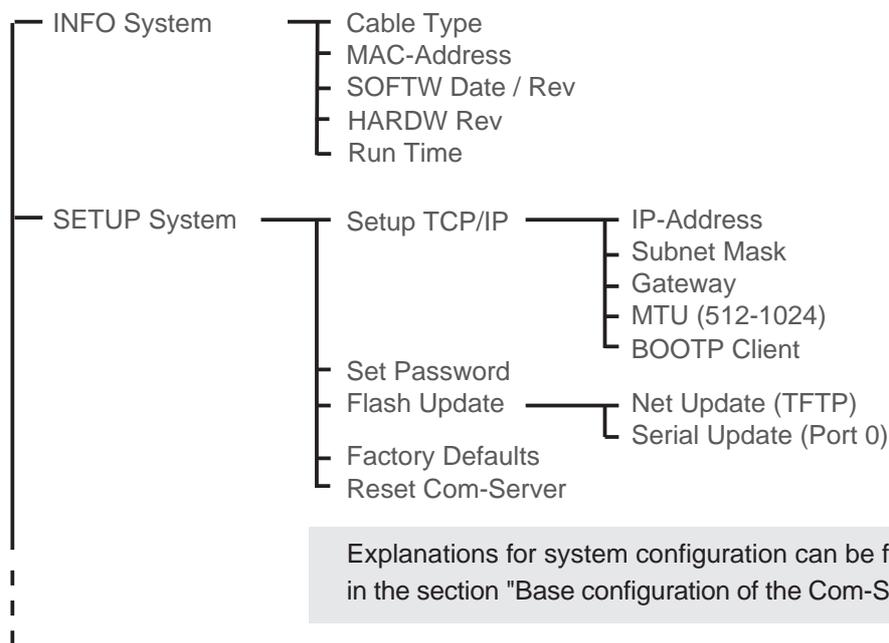
As in the case of the basic configuration, the various TCP/IP modes and interface parameters are set over the network using the Telnet protocol. This configuration can be performed from practically any computer having a network connection and TCP/IP protocol.

To select the configuration menu, start the Telnet client as follows (see „Starting the Telnet configuration“):

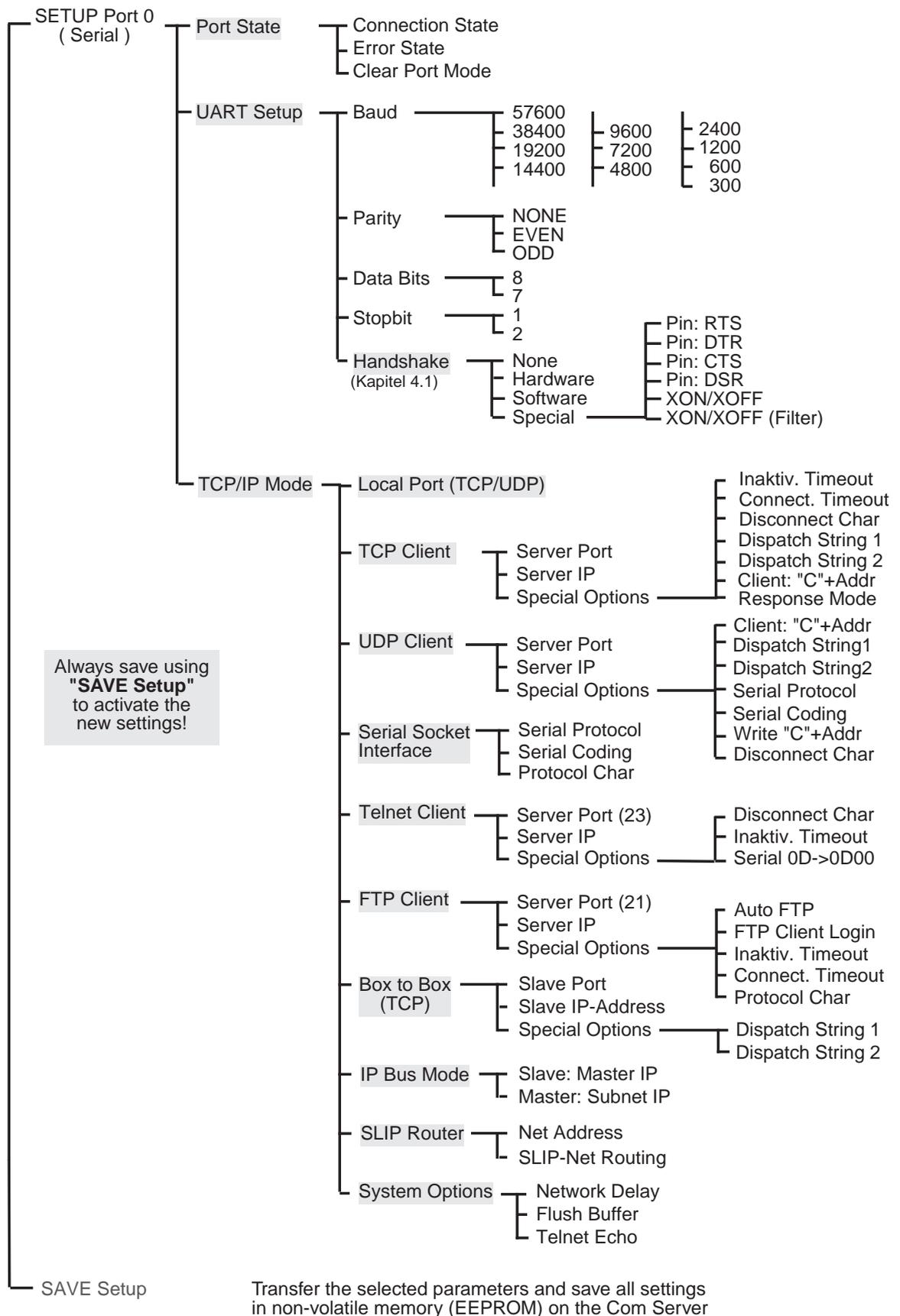
### 8.1 The configuration menu

The following overview shows the entire menu. You will only see one level at a time on the monitor; enter a digit to scroll forward, and press ENTER to move backwards.

All your settings are activated in the Com-Server only after you have saved them using the **SAVE Setup** menu. Until then you may move around the menu as you like, change values and set different modes. If you close the Telnet connection without saving, the Com-Server retains all the settings it had before you opened the configuration menu.



## Configuration of the serial port



## 8.2 The serial parameters (Menu: UART Setup)

Like any serial device, the Com-Server port must be set to match the communication parameters of its partner. All the RS232 parameters relevant for operation are configured in the *Setup Port 0 → UART Setup* menu. In addition to the usual settings for baud rate, data bits, parity and stop bits, here you can also specify the handshake procedure as well as the functions of the individual control signals.

! The settings only become active after saving with „SAVE Setup“. Please note that saving the parameters deletes all data currently located in the ComServer port buffers.

### 8.2.1 The handshake modes (SETUP Port 0 → UART Setup → Handshake)

This menu provides three standard modes for flow control of the serial port. As opposed to these predefined handshake profiles, you can use the *Special* submenu to specify the function of each control signal individually. This allows you for example to use hardware-controlled TCP connection control. Details on the control line functions are contained in the description of the submenu.

#### → Hardware Handshake

When using hardware handshake, the individual RS232 signals perform their factory-set functions as described in the section „RS232 Interface“. Please note that the meaning of the individual signals may be changed by the respective user software.

Function of the control signals:	RTS:	Flow Control
	DTR:	Show Connection
	CTS:	Flow Control
	DSR:	NO

#### → Software Handshake

The software handshake is implemented using the two ASCII characters (11H)=XON and (13H)=XOFF. In the standard setting *Software Handshake* these two codes are considered by the Com-Server as control characters and are filtered out from the actual data flow in both directions. This filtering can be turned off individually for each data direction in the *Special* submenu.

Function of the control signals:	RTS:	NO (Default HIGH)
	DTR:	NO (Default HIGH)
	CTS:	NO
	DSR:	NO

#### → NO Handshake

Turns off any hardware or software flow control for the serial interface. All the data are output to the connected terminal device regardless of the status of the inputs CTS and DSR. Any impending overflow of the serial input buffer is not signalled by the Com-Server. The setting *NO Handshake* makes sense for any data transmission that uses a serial protocol for data integrity or flow control.

Function of the control signals:	RTS:	NO	(Default HIGH)
	DTR:	NO	(Default HIGH)
	CTS:	NO	
	DSR:	NO	

→ **Special**

Here it is possible to configure the function of the control lines differently from the three predefined protocols. In addition, the filter function of the „Xon“ and „Xoff“ characters used for the software handshake can be influenced. Toggle between on and off by entering the menu number of the desired function. Settings made here are correspondingly overwritten by selecting one of the three predefined handshake modes.

→ **Pin: RTS & Pin: DTR**

The following functions can be assigned to the outputs RTS and DTR:

1. **Flow Control** If this function is turned on, the Com-Server handles the RS232 flow control through the corresponding output. The ready state for serial data is signaled by a HIGH signal (+3...12V). If the memory limit of the input buffer is reached, the line is set to LOW (-3...-12V).
2. **Show Connection** The output is switched to the serial port depending on an existing TCP connection. HIGH (+3...12V) indicates an existing connection.
3. **NO (Default=HIGH)** The outputs are not used by the Com-Server and carry a constant HIGH level (+3...12V). A network application can use the control port of the Com-Server (see „Enhanced services of the Com-Server“) to temporarily influence the level.
4. **NO (Default=LOW)** The outputs are not used by the Com-Server and carry a constant LOW level (-3...-12V). A network application can use the control port of the Com-Server (see „Enhanced services of the Com-Server“) to temporarily influence the level.

→ **PIN: CTS and PIN: DSR**

The inputs CTS and DSR can be assigned the following functions:

1. **Flow Control** The serial flow control is performed using the corresponding pin. Serial data are output only if the input was set to HIGH (+3...12V) by the connected terminal device.
2. **OPEN/CLOSE Connection** This option for TCP connection building and ending can be used to control connection building and ending in all Com-Server client operating modes. If the IP address and the TCP port are stored in the Com-Server, a HIGH level (+3...12V) on the corresponding input establishes a connection. The connection is closed with a LOW level (-3...-12V).
3. **Accept only by HIGH** Connection establishment of a TCP client is only accepted if a HIGH level (+3...12V) is present on the corresponding input. Likewise the connection is rejected when there is a LOW level.
4. **NO (Manual IN)** The selected input is ignored by the Com-Server and plays no role either in data transmission or connection control.

→ **XON/XOFF**

Here the handshake can be configured for XON/XOFF separately for each data direction. "XON/XOFF Receive" switches processing of this control character for serial receiving on: After receipt of an XOFF sent by the RS232 terminal device, the Com-Server stops serial data output until it receives an XON. If "XON/XOFF Receive" is turned on, the Com-Server generates an XOFF when it is no longer able to receive serial data.

→ **XON/XOFF (Filter)**

If XON/XOFF is turned off for flow control, this option can be used to separately determine for both data directions whether the Com-Server filters these control characters out of the actual data.

1. **Send-Filter (Data direction → serial)** If this option is turned on and XON or XOFF characters occur in the actual data, the Com-Server filters them out and does not pass them along to the terminal device. Select this mode generally for bi-directional RS232 connections, since otherwise you will have problems with data traffic. If the Send-Filter option is turned off, all the data including the XON and XOFF characters are sent unfiltered on the RS232 interface. This mode only makes sense if the connected terminal device is a graphics printer whose data stream can be expected to contain handshake characters.
2. **Receive-Filter (Data direction → network)** If this option is turned on, the Com-Server interprets the XON and XOFF characters sent by the serial terminal device as control bytes and does not insert them into the network data stream. This ensures that the recipient in the network only gets actual user data. If Receive-Filter is turned off, the XON and XOFF characters sent by the serial terminal device are transmitted to the network together with the actual user data. This mode requires of the respective recipient in the network that it be able to separate actual data from control data.

The currently selected parameters are shown in the top line of the „UART Setp“ menu. As soon as a parameter is changed, until the parameters are saved this display does not show the configuration with which the Com-Server port is currently working!

Display format: *[Baud rate], [Parity], [Data bits], [Stop bits], [Handshake]*

Possible values for the handshake variables are:

		<b>Send-Filter</b>	<b>Receive-Filter</b>
[N]	No handshake	----	----
[H]	Hardware handshake	----	----
[S]	Software handshake XON/XOFF	ON	ON
[Special]	Special setting configuration menu "Handshake-Special"	----	----

## 9 Configuring the TCP/IP modes (Menu: TCP/IP Mode)

There are several possibilities for transporting serial interface data. The Com-Server port supports the standard protocols FTP and Telnet. It also allows you to transport the data on the socket interface level directly as TCP streams or UDP datagrams - in other words, without any additional higher-order protocol. The illustration below shows the protocol layering in the Com-Server:

Standard Protocols:

Socket interface applications:

FTP Client/Server	Telnet Client/Server	Client/Server	Client/Server
		SOCKET API	
TCP Streams		UDP Datagrams	
IP (Internet Protocol)			ICMP
Ethernet			

Each mode is explained individually in the following sections. The server mode does not have to be configured separately; it is activated when a client of a TCP/IP station directs a connection request to the server in the Com-Server.

The client processes in the Com-Server need to be set up in the configuration menu of the Com-Server port.

### 9.1 Configuring the TCP port number of the Com-Server

Direct access via TCP or UDP to the serial port of the Com-Server is done using the port number stored in the menu branch *Setup Port 0 -> TCP/IP Mode -> Port Number*. For reasons of downward compatibility with older firmware versions, port 8000 is pre-configured at the factory.

It must be noted that only one TCP client at a time can be connected to the serial port of the Com-Server. If there is already a connection, any attempt from another client is rejected until this first connection has been properly closed or ended by means of a reset.

**!** *This configuration method is intended only for direct socket access per TCP or UDP. Higher-order services like TELNET or FTP are not reachable in this way. If the configurable port number is set to one of the standard values (e.g. 21 for FTP), the associated service is no longer reachable and the data area of the TCP or UDP protocol is passed on transparently to the serial terminal device.*

## 9.2 Data transfer per TCP/IP Sockets

Using the socket API (under Windows *WinSock*, under UNIX *Berkley Sockets*, ...) it is possible to implement a variety of applications in the form of client or server processes on TCP/IP computers. The API offers all the functionality for transporting data over the network. You can customize your application program - that is, the further processing and evaluation of the data - for your particular requirements.

The Com-Server provides both of the protocols possible on the socket level for data transfer: Client/server process with TCP sockets (streams) or UDP sockets (datagrams). The following comparison shows the essential features of both protocols at a glance.

Feature	UDP Protocol	TCP Protocol
Data integrity using checksum	Yes	Yes
Connection control using TCP/IP stack (opening and closing the connection)	No	Yes
Guarantee correct packet sequence using TCP/IP stack	No	Yes
Limit number of active connections on the computer	No with a socket multiple ports can be polled	One socket per connection to a serial port is needed

TCP is a connection-oriented protocol, i.e. during data transfer there is a fixed connection between client and server. TCP has all the mechanisms for opening a connection, closing it and ensuring errorless data transfer over the network.

UDP has none of these mechanisms, but it is often faster, since there is no connection opening and closing and there can be no timeout situations. If a packet gets lost, data transfer continues uninterrupted as long as there is not a higher-order protocol to provide for repetitions.

### 9.2.1 TCP server mode (Default-Port number 8000)

No other settings are necessary for the TCP server mode if you want to use the default port 8000. A connection to the Com-Server port can be opened from any TCP/IP station in the network. If the port is in use by a client process, further connection requests are rejected until the active connection is closed.

The connection is controlled by the user program (client process), which opens (*connect()*) and closes (*close()*) the connection. Once a connection is established, data can be exchanged bidirectionally between the two processes. The Com-Server output all the data from the LAN to the serial port and in return reads in all the data from the serial port in order to convey it to your client process.

! When there is no connection, the entry „FREE“ must be visible in the Com-Server port menu SETUP Port 0 → Port State → Connection State. If a connection is active, the entry „In Use Port number <IP address>“ appears there!

### 9.2.2 TCP client mode (Menu: TCP Client)

The Com-Server port is configured as a TCP client (data transport in streams) in the submenu *TCP-Client*.

In this mode the Com-Server port is able to as a client itself to open a connection to the configured TCP server and close it again after an expired timeout. The connection is opened when data are received at the serial port. If there is no connection to the TCP server currently active, the port

itself can also be a TCP server and accept connections from clients on its default TCP port or on the port configured in the menu branch "*Setup Port 0 → TCP/IP Mode → Port Number*".

**Menu settings: SETUP Port 0 → TCP/IP Mode → TCP Client**

→ **Server Port**

Port number that the application (TCP server process) addresses on the computer.

*Format: decimal*

→ **Server IP**

IP address of the computer on which your application (TCP server process) is active.

*Format: Dot notation*

→ **Special Options → Dispatch Str. 1 & Dispatch Str. 2 (default: 000H)**

This mode allows compact sending of serial data to the network; the serial data stream is not broken down randomly, rather serial packets can be transferred over the network in their original context as a packet.

In Dispatch Str. 1 and/or Dispatch Str. 2 enter in hexadecimal format two characters each to be used for finding the serial data stream. Only when one of the two strings has been found are the data packed into network packets by the serial port. If only one character is to be searched for, Dispatch Str. 2 must be set to „0000“. If you for example configure Dispatch Str. 1 to „3100“ and Dispatch Str. 2 to „0000“, only the character „1“ will be searched for.

*Format: 16-bit integer hexadecimal, Host-Order (leading Low-Byte)*

*Dispatch Str. 1 and Dispatch Str. 2 = 0 deactivates the mode*

→ **Special Options → Inactivity Timeout (default: 30)**

Here you can specify the value for a timer. When the specified time expires, the Com-Server port closes the connection. The timer is reset if there is an active network connection when data are being exchanged. If no data are set within the specified time, the Com-Server port closes the connection to the TCP server.

*1 Tick: 1 second*

*Format: decimal*

*InactivityTimeout = 0 deactivates the connection closing after a timeout.*

→ **Special Options → Connection Timeout (default: 300)**

This value is a connection timeout that is only effective together with an activated *Inactivity Timeout* . After the *Inactivity Timeout* expires, the Com-Server tries to send any still existing, unsent user data for the duration of the *Connection Timeout*. If it receives no reply from the TCP server within this time, you may assume the connection is „hanging“; the data are then rejected and the connection reset. To prevent unintended loss of data, make this value sufficiently large.

*1 Tick: 1 second*

*Format: decimal*

*Connection Timeout = 0 deactivates resetting of the connection after a timeout.*

→ **Special Options → Disconnect Char (default: 0)**

If the Com-Server receives the character defined here at the serial port, the connection to the TCP server is closed. The character itself is not sent to the TCP server. Default value is 0.

*Format: decimal*

*Disconnect Char = 0 deactivates this mode*

→ **Special Options → Client: "C"+Addr (default: deactive)**

Activating this switch activates the TCP client mode to alternating TCP servers. The address of the TCP server need not be permanently configured, rather it is send as an ASCII string in front on the user data on the serial port. The string for addressing is not included in the send.

There are two possibilities for string format:

1. The parameters *Server Port* and *Server IP* are zero.  
Format: C IP address, Port number <CR> (e.g. "C172.16.231.101,4800<CR>")
2. The parameter *Server Port* contains the Default Port (e.g. 4800), *Server IP* contains the first three bytes of the IP address (e.g.172.16.231.0).  
Format: C 4.Byte IP address <CR> (e.g. "C101<CR>")

! The string is not allowed to contain spaces!

→ **Special Options → Response Mode (default: deactive)**

In *Response Mode* the application is informed of the network-side connection status of the Com-Server in the serial terminal device by the output of special characters. This function is only available in TCP client mode.

The following messages are output on the serial port:

- C (connected)** The connection was successfully opened. There is a TCP connection to the server.
- N (not connected)** The connection was not successfully opened. There is no response from the TCP server.
- D (disconnected)** The connection attempt was rejected by the TCP server, or a previously existing connection was normally closed by the TCP server or Com-Server.
- Ixxx.xxx.xxx.xxx** (invoked by ...) A client application on the indicated station successfully opened a TCP connection to the Com-Server (e.g. I172.20.20.1)

Once you have entered all the parameters in the Com-Server port menu, press the ENTER key several times to return to the main menu and save the entries with *SAVE Setup*. The message „Saving...“ appears, then the TCP client mode is activated. The entry „TCP-Client“ now appears in the *SETUP Port.. → Port State → Connection State* menu. The current status of the connection can be read in this menu at any time.

**Deactivating TCP client mode**

Set the option *Server Port* in the menu *SETUP Port.. → TCP/IP Mode → TCP Client* to zero, or use the *SETUP Port.. → Port State → Clear Port Mode* menu. The entry „FREE“ must now appear in the *SETUP Port.. → Port State → Connection State* menu.

**9.2.3 Client/Server mode between Com-Server ports**

There are various ways of connecting two or more Com-Server ports together in this mode. All involved Com-Server ports must be configured for TCP client mode and may then alternately connect to each other when there are data on the serial port. After the data has been send the connection is closed either by the *Inactivity Timeout* or the *Disconnect Character*.

Serial addressing (Parameter *Client: "C"+Addr*) makes altlernating connections between several Com-Server ports possible.

## Configuring the TCP/IP modes

- **Application:** A control program polls a measuring device; the measuring device is passive.

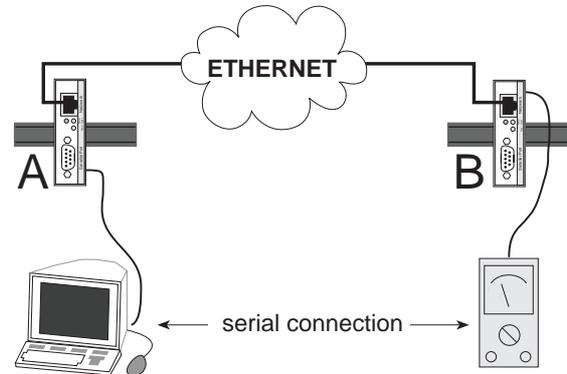
**Configuration menu:**  
→ TCP/IP Mode → TCP Client

**Com-Server A**

- Server Port: 8000
- Server IP: *IP-Address Com-Server B*
- Client "C"+Addr: deaktive

**Com-Server B**

- Server Port: 0000
- Server IP: 0.0.0.0
- Client "C"+Addr: deaktive



- **Application:** A control program polls a measuring device; the measuring device can actively send data to the control program.

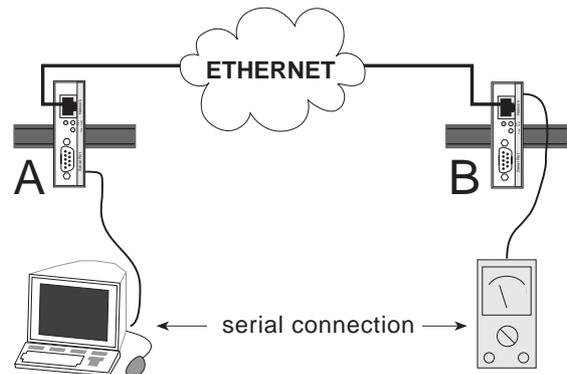
**Configuration menu:**  
→ TCP/IP Mode → TCP Client

**Com-Server A**

- Server Port: 8000
- Server IP: *IP-Address Com-Server B*
- Client "C"+Addr: deaktive

**Com-Server B**

- Server Port: 8000
- Server IP: *IP-Address Com-Server A*
- Client "C"+Addr: deaktive



- **Application:** A control program polls multiple measuring devices; the measuring devices can (e.g. in case of error) independently send data to the control program.

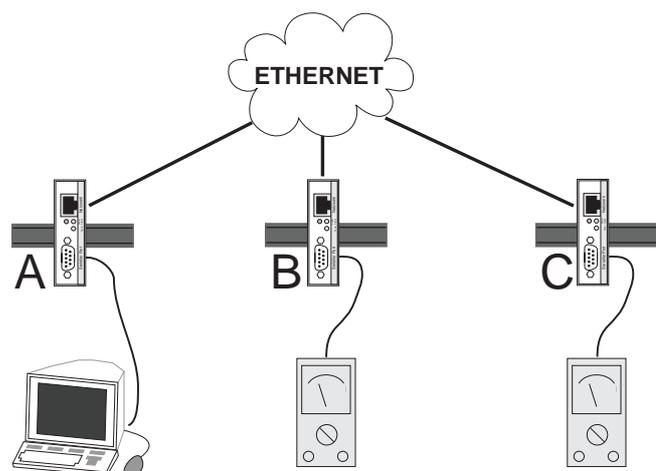
**Configuration menu:**  
→ TCP/IP Mode → TCP Client

**Com-Server A**

- Server Port: 8000
- Server IP: *IP Address Com-Server B*
- Client "C"+Addr: aktive

**Com-Server B, C, ...**

- Server Port: 0000
- Server IP: 0.0.0.0
- Client "C"+Addr: deaktive



The control program addresses the measuring devices through the serial port in the following format: C *ip\_address, port\_number* <CR>

Example: C172.16.231.8,8000<CR> Opens connection to Com-Server port 0 with IP address 172.16.231.8

#### 9.2.4 UDP client mode (Menu: UDP Client) (Default-Port number 8000)

The term „UDP Client“ is somewhat misleading, since UDP has no mechanisms for opening and closing connections. Whether a process behaves like a server or like a client is determined only by the protocol layer which is above UDP (Application Layer).

! *A Com-Server port must always (whether as client or as server) be configured for data exchange using UDP datagrams in order to ensure a clear arrangement for independent data transfers from the Com-Server port. Accepted are all UDP datagrams that are addressed to the configured local port number.*

#### 9.2.5 Setting the local UDP-Port number of the Com-Server

Access to the serial port on the Com-Server per UDP takes place through the default UDP port 8000. If this port cannot be used by the client application, you may use menu branch *Setup Port 0 -> TCP/IP Mode -> Local Port* to specify a different port on which the Com-Server accepts UDP packets addressed to it.

##### Menu settings: **SETUP Port..→ TCP/IP Mode → UDP Client**

##### → **Server Port**

Port number that the applicaiton (UDP server process) addresses on the computer.

##### → **Server IP**

IP address of the computer on which the application is active.

##### → **Special Options → Dispatch Str. 1 & Dispatch Str. 2 (default 0000H)**

This mode allows compact sending of serial data to the network; the serial data stream is not broken down randomly, rather serial packets can be transferred over the network in their original context as a packet.

In Dispatch Str. 1 and/or Dispatch Str. 2 enter in hexadecimal format two characters each to be used for finding the serial data stream. Only when one of the two strings has been found are the data packed into network packets by the serial port. If only one character is to be searched for, Dispatch Str. 2 must be set to „0000“. If you for example configure Dispatch Str. 1 to „3100“ and Dispatch Str. 2 to „0000“, only the character „1“ will be searched for.

*Format: 16-bit integer hexadecimal, Host-Order (leading Low-Byte)*

*Dispatch Str. 1 and Dispatch Str. 2 = 0 deactivates the mode*

##### → **Special Options → Serial Protocol (default: deactive)**

##### → **Special Options → Serial Coding (default: deactive)**

The protocol mode activatable using these two menu items has been replaced by the „Serial Socket Interface“ (see section „Serial Socket Interface“). For reasons of downward compatibility the functions are still available, but for new applications only the „Serial Socket Interface“ should be used.

##### → **Special Options → Client: "C"+Addr (default: deactive)**

Activating this switch activates UDP client mode to alternating TCP servers. The address of the TCP server need not be permanently configured, rather it is send as an ASCII string in front on the user data on the serial port. The string for addressing is not included in the send.

There are two possibilities for string format:

1. The parameters *Server Port* and *Server IP* are zero.  
Format: C *IP address, Port number* <CR> (e.g. "C172.16.231.101,4800<CR>")
2. The parameter *Server Port* contains the Default Port (e.g. 4800), *Server IP* contains the first three bytes of the IP address (e.g. 172.16.231.0).  
Format: C *4.Byte IP address* <CR> (e.g. "C101<CR>")

! *The string is not allowed to contain spaces!*

→ **Special Options → Write "C"+Addr (default: deactive)**

Set this switch to active (1) to output the sender address in ,ASCII format before outputting the data for a UDP datagram to the serial port. The string always contains 22 characters.

Format: C *IP address, Port number* (e.g. "C172.016.231.101,04800")

→ **Special Options → Disconnect Char (default: 0)**

This parameter is only processed if the parameter *Client: "C"+Addr* is activated. If the Com-Server receives the character configured here at its serial port, it deletes the last serially received UDP server address. The character itself is not sent. The preset value is 0.

*Format: decimal*

*Disconnect Char = 0 deactivates this mode*

After you have entered all the parameters in the Com-Server port menu, pressing the ENTER key repeatedly to return to the main menu and save the entries using *SAVE Setup*. The „Saving...“ message indicates that the UPD-Client mode is activated. Now the entry „UDP-Client“ appears in the *SETUP Port.. → Port State → Connection State* menu. The current status of the connection can be read in this menu at any time.

### 9.3 Serial socket interface

The Serial Socket Interface enables defined communication between Com-Servers and the device connected to their serial port. Comparable to the TCP/IP socket interface in Windows or UNIX system, this interface offers the following possibilities for the serial port.

- ✓ Opening and closing up to 8 handles (data connections) at the same time
- ✓ Specifying the type of network connection (UDP or TCP) and the connection parameters (destination IP address and port plus local port of the Com-Server)
- ✓ Transmitting IP broadcasts
- ✓ The Com-Server independently informs the serial terminal device as part of the protocol of changes in the system or connection status
- ✓ Packet-oriented transmission of serial data in both directions, i.e. sending of packets with packet start characters, packet headers, data, packet end characters
- ✓ Transmission of the network destination address (IP address, UDP port) to the Com-Server and the network sender to the serial terminal device with the data
- ✓ Parameterizable packet frame characters (packet end/packet start) or character doubling procedure for not having to exclude characters from the stream
- ✓ Configurint the Com-Server serial port
- ✓ Configuring the Com-Server network parameters
- ✓ Reading the configurations

The detailed protocol documentation can be downloaded from our Web site.

## 9.4 Data transfer per Telnet

Telnet is a terminal emulation protocol. Data transmission is character-oriented. Telnet allows bi-directional character exchange between the TCP/IP computer and the serial terminal device connected to the Com-Server.

### 9.4.1 Telnet server mode (Port number 23 or 6000)

No pre-settings are necessary on the Com-Server. You can access the device just like any other station in the network under its name or IP address.

**Telnet invocation:** `telnet [IP-Number]` or `telnet [Host-Name]`

If the connection can be opened, a message from the TELNET server appears on the screen. From here on data can be read in or output a character at a time from the keyboard.

All data received from the serial port are output on the screen and - if desired - recorded in a protocol file. All keyboard entries are output as data a character at a time on the serial port.

The echo behavior of the Com-Server can be configured in the menu branch *SETUP Port 0* → *TCP/IP Mode* → *System Options*.

### 9.4.2 Telnet client mode (Menu: Telnet-Client)

This mode is the simplest way to implement terminal emulation on a serial terminal device and thereby to effect direct communication with the TCP/IP station on which the Telnet server is active. The Com-Server opens the connection as soon as the serial port has received a character.

**Menu settings:** *SETUP Port 0* → *TCP/IP Mode* → *Telnet Client*

→ **Server Port (23)**

Port number that the Telnet server addresses (23 = Standard Telnet-Port)

→ **Server IP**

IP address of the TCP/IP computer on which the Telnet server is active.

→ **Special Options** → **Inactivity Timeout (default: 30)**

Here you can specify the value for a timer. When the specified time expires, the Com-Server port closes the connection. The timer is reset if there is an active network connection when data are being exchanged. If no data are set within the specified time, the Com-Server port closes the connection to the TCP server.

*1 Tick: 1 second*

*Format: decimal*

*InactivityTimeout = 0 deactivates the connection closing after a timeout.*

→ **Special Options** → **Disconnect Char (default: 0)**

When the Com-Server port receives the character configured here at its serial port, the Com-Server port closes the connection to the Telnet server. It is important that this value not be used within a Telnet session, since this would result in premature closing. The character itself is not sent. The factory set value is 0.

*Format: decimal*

*Disconnect Char = 0 deactivates this mode*

→ **Special Options → Serial 0D->0D00 (default: 0)**

If this switch is activated, an 00H is appended to the serially received character 0DH: in other words, 0DH 00H is sent over the network. This option may have to be activated for sending binary files.

To activate the Telnet-Client mode, enter the desired parameters and then press the ENTER key repeatedly to return to the main menu and save the entries using *SAVE Setup*. The message „Saving...“ appears. The selected mode is shown in the *SETUP Port.. → Port State → Connection State* menu; the current status of the connection can also be read at any time in this menu.

**Deactivating Telnet client mode**

Set the option *Server Port* in the menu *SETUP Port.. → TCP/IP Mode → Telnet Client* to zero, or use the *SETUP Port.. → Port State → Clear Port Mode* menu. The entry „FREE“ must now appear in the *SETUP Port.. → Port State → Connection State* menu.

## 9.5 Data transfer per FTP

Data exchange per FTP is file-oriented; this means it is possible to output file contents on the Com-Server serial port or to read characters into files from the serial port. The Com-Server can function both as an FTP server and as an FTP client.

### 9.5.1 FTP server mode (Port number 21 or 7000)

This mode should be used if the data to be sent exist as files and the action will always be pre-settings on the Com-Server are necessary for this mode. You can access the unit just like any other station in the network under its name or IP address.

**FTP invocation:** `ftp [IP-Number]` or `ftp [Host-Name]`

After entering the `ftp` command you can acknowledge the prompt for a login name with „ENTER“.

**The following commands are available:**

<code>PUT [local file] [remote file]</code>	- Sends the file <i>local file</i> to the RS232
<code>GET [remote file] [local file]</code>	- Gets characters from the RS232 in <i>local file</i>
<code>ASCII</code>	- Send in ASCII mode
<code>IMAGE</code>	- Send in binary mode
<code>QUIT</code>	- Ends the FTP session

**Notes on [remote file]:**

**PUT:** Enter any desired character, since no *remote file* exists.

**GET:** Again enter any desired letter, or encode a timeout criterion for the data transfer. Enter a number no longer than 3 digits (one tick corresponds to one second). If *[remote file]* does not contain a value in this range, the connection will be broken off 30 seconds after the last retrieved character.

e.g.: `GET 5 /user/cs_in` (breaks the connection off after 5 seconds of no data traffic)

For the exact syntax of your function invocations, please consult the user's manual for your FTP software.

! The entry "FREE" must be located in the SETUP Port 0 → Port State → Connection State menu before a connection can be opened.

### 9.5.2 FTP client mode (Menu: FTP-Client)

This mode enables access to files on any TCP/IP computer having an activated FTP server. The action must always be initiated by the serial terminal device.

The address of the FTP server (Port number and IP address) must in any case be configured in the Com-Server. For the sequence of the FTP protocol (Login, file command, ...) there are two possibilities: The automatic FTP client and the FTP client with serial protocol.

**Menu settings: SETUP Port 0 → TCP/IP Mode → FTP Client**

→ **Server Port (21)**

Port number that the FTP server addresses (Standard-FTP-Port: 21)

→ **Server IP**

IP address of the TCP/IP computer on which the FTP-Server is active.

Once you have entered all the parameters in the Com-Server port menu, press the ENTER key several times to return to the main menu and save the entries with *SAVE Setup*. The message „Saving...“ appears, then the FDP client mode is activated. The entry „FDP-Client“ now appears

in the *SETUP Port.. → Port State → Connection State* menu. The current status of the connection can be read in this menu at any time.

### The automatic FTP client

This mode is recommended whenever you need to execute the same command over and over again. The FTP commands are configured in the Com-Server. The latter opens a connection to the FTP server when it receives data at the serial port. It sends the login and runs the configured file command (getting serial data and saving as a file or reading a file from the FTP server and outputting at the serial port). Then the connection is closed. Configure the following parameters:

#### → Special Options → Auto FTP

Set this switch to „active“ to activate the automatic FTP client.

#### → Special Options → FTP Client Login

Enter the FTP commands one after the other.

Login:	Login-Name for the FTP-Server
Password:	Login-Password for the FTP-Server
[TYPE A/TYPE I]:	Transmission mode (ASCII/binary)
[STOR/APPE/RETR/LIST] [dir/file]:	File command (see command list at end of sect.)

If a command sequence was already configured, it is shown when opening the menu.

! *A maximum of 80 characters are available for all commands! Commands are case sensitive.*

#### → Special Options → Inactivity Timeout (default: 30)

Configure a timeout for the FTP commands APPE(nd) and STOR for breaking off the connection. If no serial data are received during this time, the FTP client closes the connection to the FTP server.

*1 Tick: 1 second*

*Format: decimal*

*Inactivity timeout = 0 means infinite (in this case the parameter „Protocol Char“ must be configured!)*

#### → Special Options → Connection Timeout (default: 300)

This value is a connection timeout that is only effective together with an activated *Inactivity Timeout*. After the *Inactivity Timeout* expires, the Com-Server tries to send any still existing, unsent user data for the duration of the *Connection Timeout*. If it receives no reply from the TCP server within this time, you may assume the connection is „hanging“; the data are then rejected and the connection reset. To prevent unintended loss of data, make this value sufficiently large.

*1 Tick: 1 second*

*Format: decimal*

*Connection Timeout = 0 deactivates resetting of the connection after timeout.*

#### → Special Options → Protocol Char (default: 0)

If the Com-Server receives the character defined here at the serial port, the connection to the FTP server is closed. The character itself is not sent. Default value is 0.

*Format: decimal*

*Disconnect Char = 0 deactivates this mode (in this case the parameter „Inactivity Timeout“ must be configured!)*

		Open connection	Close connection
LIST	Directory	The connection is opened when any character is received at the serial port. This character is not sent.	The connection is closed when the FTP server has sent all data.
RETR	File		
APPE	File	The connection is opened when the first character to be sent is received at the serial port.	The connection is closed when no serial data have arrived during the time period specified under <i>Inactivity Timeout</i> or when <i>Protocol Char</i> was received.
STOR	File		

**FTP client with serial protocol**

This mode is recommended where constantly changing commands with different files need to be executed and where terminal entry is possible or there is a programmable serial device.

The sequence of the FTP commands is given to the Com-Server port on the serial interface. The Com-Server gets the command sequence, opens the connection to the FTP server and gives an OK to start data transmission. Configure the following parameters:

→ **Special Options → Auto FTP**

Set this switch to "deactive".

→ **Special Options → Protocol Char ( default: 0 )**

Choose and enter here a character with which the sequence of FTP commands will be separated from the user data stream. The pre-adjustment is 0. With an ASCII-data transfer (type A) the receipt of the protocol character in the data stream causes a shut down of the connection. If the transmission takes place with TYPE I (binary), the occurrence of the protocol character in the data stream will be ignored. The only possibility for a disconnection in this case is by way of an inactivity timeout.

*Format: decimal*

**FTP commands for the FTP client**

TYPE	A or I	Activation of ASCII or binary mode
STOR	File	Saves all serially received data in <i>File</i>
APPE	File	Saves all serially received data to the file end of <i>File</i>
RETR	File	Outputs <i>File</i> to the serial port
DELE	File	Deletes <i>File</i>
LIST	Directory	Outputs the directory contents on the serial port
RESET		Generates a software reset of the Com-Server. This command may only be used if there is no active connection to the FTP server

**Please note: Commands in ALL CAPS!**

**Command string format (entering over the serial interface)**

<i>login</i> <lf>	<i>login</i> <lf>
<i>password</i> <lf>	<i>password</i> <lf>
<i>TYPE I</i> or <i>TYPE A</i> <lf>	<i>DELE remote file/dir</i> <lf>
<i>STOR, RETR, APPE</i> or <i>LIST remote file/dir</i> <lf>	< <i>Protocol Char</i> >
< <i>Protocol Char</i> >	
<i>RESET</i> <lf>	
< <i>Protocol Char</i> >	

The commands are separated by 0AH; this hex value corresponds to a Line Feed and can be generated from the keyboard with **CTRL J**.

! A maximum of 128 characters are available for the complete login string!

**Opening and closing the connection**

1. Send the command string to the serial port. If the complete string was received, the Com-Server begins to open the connection and process the commands.
2. If the functions can be executed one after the other, you get the string *OK + Protocol Char*.
3. Then the actual user data are sent, depending on the command either from the serial port to the network or the reverse.
4. In case of error the FTP server sends the error code + *Protocol Char + Protocol Char*.
5. The connection is automatically closed in either case, and you receive the end code + *Protocol Char* from the FTP server.

**Examples:**

1. User *egon* having the password *happy* wants to output the file */etc/hosts* in ASCII format. The value „003“ (Ctrl C) is entered as *Protocol Char*. The protocol would look as follows:

Serial terminal device: *egon*<lf>*happy*<lf>TYPE A<lf>RETR */etc/hosts*<lf>  
<Ctrl C>

Com-Server-Port: OK<Ctrl C> [contents of file */etc/hosts*]<Ctrl C>  
221 Goodbye<Ctrl C>

2. User *egon* having the password *happy* would like to insert binary data to the end of the file */usr/egon/config*. The value „003“ (Ctrl C) is entered as *Protocol Char*. The protocol would look as follows:

Serial terminal device: *egon*<lf>*happy*<lf>TYPE I<lf>APPE */usr/egon/config*<lf>  
<Ctrl C>

Com-Server-Port: OK<Ctrl C>  
Serial terminal device: [binary data]  
Com-Server-Port: 221 Goodbye<Ctrl C>

3. User *egon* having the password *happy* would like to view the file */usr/privat* and transfer the data in ASCII. The file does not exist however. The protocol would look as follows:

Serial terminal device: *egon*<lf>*happy*<lf>TYPE A<lf>RETR */usr/privat*<lf>  
<Ctrl C>

Com-Server-Port: 550 */usr/privat: No such file or directory*<Ctrl C>  
<Ctrl C> 221 Goodbye<Ctrl C>

**Deactivating FTP client mode**

Set the entry *Server Port* in menu *SETUP Port 0 → TCP/IP Mode → FTP Client* to zero, or use the menu *SETUP Port.. → Port State → Clear Port Mode*. You must see the entry „FREE“ on the *SETUP Port.. → Port State → Connection State* menu. You must see the entry „FREE“ on the *SETUP Port.. → Port State → Connection State* menu.

## 9.6 „Box to Box“ mode (Menu: Box to Box)

In „Box to Box“ mode any two Com-Server serial ports can be logically linked to each other over the network. The two connected serial terminal devices are in constant online contact in this mode. Any additional data traffic or other network protocols have no effect on the connection.

In this mode a **permanent TCP connection** is opened between the Master port and the Slave port. It makes no difference which port is the Master and which the Slave. The Master port functions as a TCP client and is therefore responsible for opening (after a configuration or reset) and closing (after deactivating „Box to Box“ mode).

### Menu settings: SETUP Port..→ TCP/IP Mode → Box to Box

! Only the Master port is configured for „Box to Box“ mode; The Slave IP address and Slave port number are set **only on the Master port!** Settings in the Special Options submenu and parameterizing of the serial port can and must be done on both ports.

#### → Slave Port (set only for Master port)

Port of the serial Com-Server Slave port. The serial port of a Com-Server can easily be „connected“ with one of the up to four serial ports of another Com-Server.

#### → Slave IP-Address (set only for Master port)

IP address of the Com-Server in which the Slave port is located.

#### → Special Options → Dispatch Str. 1 & Dispatch Str. 2 (default:0000H)

This mode allows compact sending of serial data to the network; the serial data stream is not broken down randomly, rather serial packets can be transferred over the network in their original context as a packet.

In Dispatch Str. 1 and/or Dispatch Str. 2 enter in hexadecimal format two characters each to be used for finding the serial data stream. Only when one of the two strings has been found are the data packed into network packets by the serial port. If only one character is to be searched for, Dispatch Str. 2 must be set to „0000“. If you for example configure Dispatch Str. 1 to „3100“ and Dispatch Str. 2 to „0000“, only the character „1“ will be searched for.

*Format: 16-bit integer hexadecimal, Host-Order (leading Low-Byte)*

*Dispatch Str. 1 and Dispatch Str. 2 = 0 deactivates the mode*

Once you have entered all the parameters in the Com-Server port menu, press the ENTER key several times to return to the main menu and save the entries with *SAVE Setup*. The message „Saving...“ appears, then „Box to Box“ mode is activated. The entry „Box to Box Master“ now appears in the *SETUP Port.. → Port State → Connection State* menu. The connection status can be read at any time in this menu both for the Master and for the Slave port.

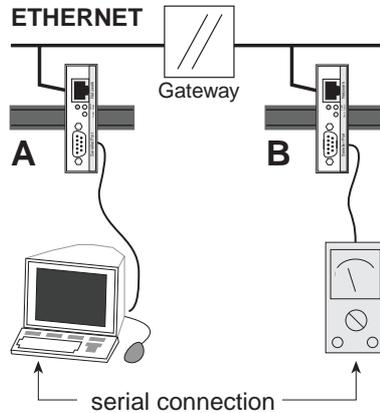
**Sample configuration for „Box-to-Box“ mode**

IP address: 172.16.231.8  
Port number of Port 0: A

Master port:

SETUP Port 0

- ↳ TCP/IP Mode
  - ↳ Box to Box
    - ↳ Slave Port **A**
    - ↳ Slave IP-Address **172.016.231.005**
    - ↳ Special Options
      - ↳ Dispatch Str.1 **0D0A**
      - ↳ Dispatch Str.2 **0000**



IP address: 172.16.231.5  
Portnumber of port 0: A

Slave port:

SETUP Port 0

- ↳ TCP/IP Mode
  - ↳ Box to Box
    - ↳ Slave Port **-**
    - ↳ Slave IP-Address **000.000.000.000**
    - ↳ Special Options
      - ↳ Dispatch Str.1 **3132**
      - ↳ Dispatch Str.2 **0D0A**

The Slave address is only configured on the Master Com-Server. The *Dispatch Strings* may be set on both ports however.

**Typical applications**

- Replacing complex serial star topology wiring, such as in small business systems.
- Serial connections over longer distances. A pair of Com-Servers replaces two line drivers and offers in addition error correction using the Ethernet transmission procedure.
- Serial remote connection using already existing Ethernet-Internetwork connections (routers, bridges, etc.).
- Implementing serial connections with frequently changing operating sites without additional cable expenditure - simply plug into the Ethernet cable!

**Deactivating „Box to Box“ mode**

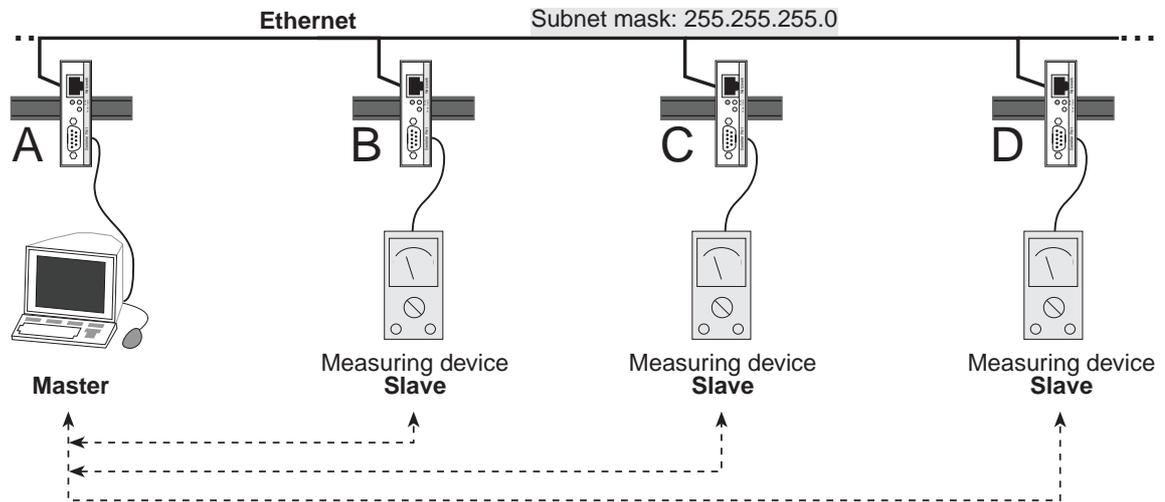
Just as in the case of configuration, „Box to Box“ mode only needs to be deactivated at the Master port in order to properly close the TCP connection. The Slave port and the Master port must both be located on the Ethernet.

Set the entry *Slave Port* in the *SETUP Port.. → TCP/IP Mode → Box to Box* menu to zero („-“), or use the menu *SETUP Port.. → Port State → Clear Port Mode*.

If for any reason the Slave port is not released, you may also use the menu *SETUP Port.. → Port State → Clear Port Mode* in the Slave port menu.

## 9.7 IP Bus mode (Menu: IP Bus Mode)

Several serial Com-Server ports can be logically linked together over the network in the form of a Master-Slave bus. The connected serial terminal devices are in constant online contact in this mode. Any additional data traffic or other network protocols have no effect on the connection.



**IP address:**

Com-Server A:  
172.16.231.1

Com-Server B  
172.16.231.36

Com-Server C  
172.16.231.89

Com-Server D  
172.16.231.14

**Menu: SETUP Port .. → TCP/IP Mode → IP Bus Mode**

→Master: Subnet IP  
172.16.231.0

→Slave: Master IP  
172.16.231.1

→Slave: Master IP  
172.16.231.1

→Slave: Master IP  
172.16.231.1

The Master sends to all Slaves and receives from all Slaves. Slaves are not able to exchange data between each other however.

**Important:**

- The serial devices must send their data using their own proofing protocol. Since the Com-Servers use Internet Protocol (IP) as the network transmission protocol, they cannot ensure data integrity.
- Since data are sent from the Master to the Slaves per broadcast, all the devices which are to comprise a bus must belong to the same subnet. This means the subnet mask and the network portion of the IP address must be identical in all the Com-Servers.

**Menu settings: SETUP Port..→ TCP/IP Mode → IP Bus Mode**

**!** You may not configure the IP Bus mode in the same Telnet session in which the IP address, the subnet mask or the gateway address for the Com-Server was already changed. After you make such settings you must first close the Telnet connection using „q“ and then reopen it.

**Activating the Master**

→ **Master: Subnet IP**

Enter here the network address of the subnet in which the Master and Slave are located. It is sufficient to enter a value other than zero (e.g. 1.0.0.0) and confirm with ENTER. The Com-Server processes the subnet IP address itself from its own IP number and the subnet mask (AND operation) and enters it.

### Activating the Slave

#### → Slave: Master IP

Set on all Com-Server ports that are connected to Slaves the full IP address of the Com-Server having the Master port.

In both cases the change must be saved after entering the IP or subnet IP address in order to activate the mode. Press ENTER repeatedly to return to the main menu and save using *SAVE Setup*. The message „Saving..“ appears, and *IP Bus Mode* is activated. The entry *Bus Master* or *Bus Slave* appears in the *SETUP Port 0 → Port State → Connection State* menu.

### Deactivating IP Bus Mode

Set the respective IP address in the *SETUP Port.. → TCP/IP Mode → IP Bus Mode* menu to zero (0.0.0.0), or use the menu *SETUP Port.. → Port State → Clear Port Mode*.

## 9.8 The Windows COM Port Redirector

The COM Port Redirector available for Windows 9x, NT and 2000 installs virtual COM ports in the respective system. These act like normal local ports as far as a serial application is concerned, but are in fact located on Com-Servers elsewhere in the network.

The COM Port Redirector is not included standard with Com-Servers. A free testing license for the current version is however available at our Web site ([HTTP://www.wut.de](http://www.wut.de)) for downloading.

Download links can be reached through the Web data sheets for the individual Com-Server models.

Additional licensing is computer-oriented, i.e. the COM Port Redirector can be used to access any desired number of serial ports located on Com-Servers. If more than one computer communicates with a Com-Server however, each station requires a separate license.

### Settings on the Com-Server

As far as the Com-Server is concerned, only the three network-specific parameters IP address, subnet mask and gateway address need to be configured in order to use a COM Port Redirector. It is not necessary to set serial transmission parameters such as baud rate. These are determined by the serial application and sent to the Com-Server over the network.

### Installing and configuring the COM Port Redirector

Information on installing and configuring a COM Port Redirector can be found in the corresponding online help.

### 9.9 SLIP router mode (Menu: SLIP Router)

In this mode the Com-Servers functions as a router. All network packets whose destination address is within the configured subnet are routed over the serial interface using SLIP. All arriving serial SLIP packets are passed along to the local Ethernet network.

**Menu settings: SETUP Port.. → TCP/IP Mode → SLIP Router**

! You may not configure for SLIP routing in the same Telnet session in which the IP address, the subnet mask or the gateway address for the Com-Server was already changed. After you make such settings you must first close the Telnet connection using „q“ and then reopen it.

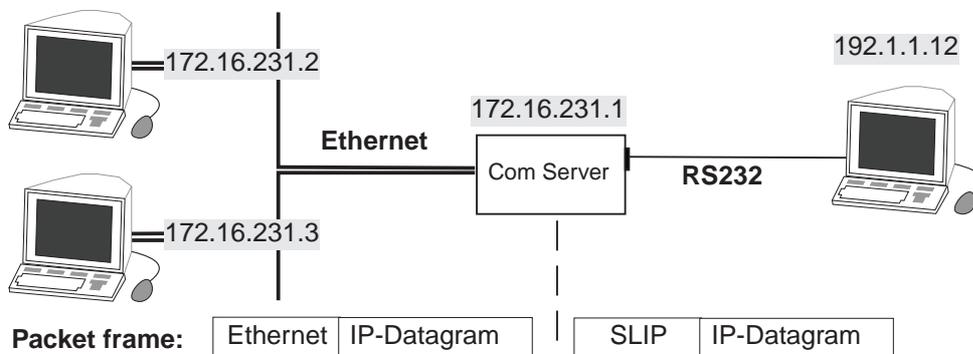
→ **Net Address**

Enter here the network address of the serially connected subnet you want to route to via SLIP.

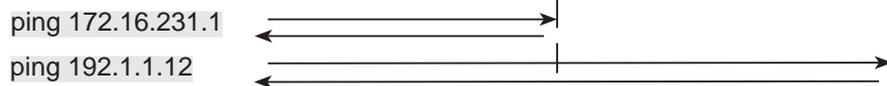
→ **SLIP-Net Routing (default: 1)**

If this parameter has a value of 1, the Com-Server functions as described like a router for the subnet specified under *Net Address*. If the parameter is 0, the Com-Server is transparent, i.e. all packets directed to the IP address of the Com-Server are forwarded as SLIP packets to the serial port. The destination IP address (IP address of the Com-Server) is replaced by the parameter *Net Address*. This allows you to assign individual IP addresses to the connected SLIP computers (=Net Address) without having to assign a separate subnet for each connection.

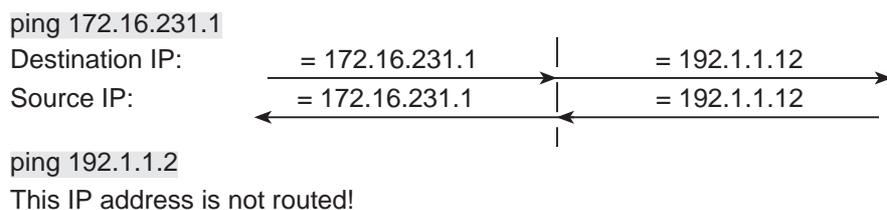
! Excluded from this procedure are packets for the Telnet configuration port 1111 of the Com-Server!



... → SLIP Router → Net Address = 192.1.1.0 / → SLIP-Net Routing = 1



... → SLIP Router → Net Address = 192.1.1.12 / → SLIP-Net Routing = 0



**Deactivating SLIP router mode**

Set the parameter Net Address to zero (0.0.0.0) in the SETUP Port..→ TCP/IP Mode → SLIP Router, or use the menu SETUP Port..→ Port State → Clear Port Mode.

**9.9.1 Configuring the Com-Server via SLIP**

It is possible to open a UDP connection for configuring some Com-Server parameters through the serial connection between the Com-Server and the connected SIP computer.

The SLIP packets must be directed to the IP address 10.0.0.1 and UDP port 1111. These packets are not forwarded to the Ethernet, but rather are processed by the Com-Server and answered as appropriate.

The packets consist of IP and UDP headers as well as the configuration data, and are exchanged between the Com-Server and the SLIP computer per SLIP.

**Format of the configuration data**

The configuration data always start with the TYPE field, which specifies the packet type, followed by the LEN field, which specifies the length of the configuration data in bytes. Next follows a list of parameters having any length, whose content and length is indicated by a defined number (PARAM\_NO).

TYPE	LEN	PARAM_NO	PARAMETER	PARAM_NO	PARAMETER	...
BYTE	BYTE	BYTE	TYPDEF	BYTE	TYPDEF	

**The parameter list**

The following Com-Server parameters can be written and/or read:

PARAM_NO	Parameter name	Data type	RD/WR
1	Setup TCP/IP → IP-Address	long (32 bit)	RD+WR
2	Setup TCP/IP → Subnet Mask	long (32 bit)	RD+WR
3	Setup TCP/IP → Gateway	long (32 bit)	RD+WR
4	Setup TCP/IP → MTU (Max. Transfer Unit)	unsigned int (16 bit)	RD+WR
5	TCP/IP Mode → SLIP Router → Net-Address	long (32 bit)	RD+WR
6	TCP/IP Mode → SLIP Router → SLIP-Net Routing	unsigned int (16 bit)	RD+WR
16	MAC-Address	char[6] (6 bytes)	RD
17	Software-Version	unsigned int (16 bit)	RD

Parameters of data type long or unsigned int are transmitted in net order, first the High byte, last the Low byte.

**Packet types (byte TYPE)**

The following three packet types are used for sending configuration data:

- **TYPE = 1: Writing parameters in the Com-Server**

This packet is sent by the SLIP computer to the Com-Server. The Com-Server carries out the configuration and deletes the packet. A packet for configuring the IP address and subnet mask would look as follows:

0	1	2	3	7	8	12
<b>TYPE</b>	<b>LEN</b>	<b>PARAM_NO</b>	<b>IP address</b>	<b>PARAM_NO</b>	<b>Subnet mask</b>	
1	10	1	hex: AC 10 EF 0 (172.16.231.1)	2	hex: FF FF FF 00 (255.255.255.0)	

- **TYPE = 2: Request for reading parameters**

This packet is sent by the SLIP computer to the Com-Server. The Com-Server sends a response type (TYPE=3) with the contents of the requested parameters. The packet contains the fields TYPE and LEN and a list of the desired parameter numbers (PARAM\_NO). A packet for reading the MTU and the MAC address would look as follows:

0	1	2	3	4
<b>TYPE</b>	<b>LEN</b>	<b>PARAM_NO</b>	<b>PARAM_NO</b>	
2	2	4	16	

- **TYPE = 3: Response to a request for reading parameters**

The Com-Server uses this packet to reply to a request for reading parameters (TYPE=2). It is used for sending the requested parameter contents. This packet is constructed like TRYPE 1. The reply to a request for parameters MTU and MAC address would look as follows:

0	1	2	3	5	6	12
<b>TYPE</b>	<b>LEN</b>	<b>PARAM_NO</b>	<b>MTU</b>	<b>PARAM_NO</b>	<b>MAC-Address</b>	
3	10	4	hex: 02 00 (512)	16	hex: 00 C0 3D 00 30 DB (00-C0-3D-00-30-DB)	

## 10 Status and error messages (Menu: Port State)

In this menu you will find information about the configured TCP/IP mode of the Com-Server port, the status of the network connection and a listing of error which occurred.

### → Connection State

This menu permits online connection control. The display is organized as follows:

```

Connection State
FREE
    
```

The port is in standard mode and has no connection.

```

Connection State
In Use: Port 2000 (172.016.231.001)
    
```

The port is in server mode and has a connection to the process having port number 2000 on the TCP/IP station having IP address 172.16.231.1

→ **Client address**

```

Connection State
TCP Client
Locked: Port 2000 (172.016.231.001)
    
```

The port is configured as a TCP client and has a connection to the server process having address port no. 2000, IP address 172.16.231.1

→ **Configured TCP/IP mode**

→ **Server address**

→ **Connection state client mode:**

- Unlock: The port is configured for the indicated mode but has **no active** connection.
- Locked: The port is configured for the indicated mode and has an **active** connection to the configured server.
- Scanning: The Com-Server wants to open a connection and **searches** for the configured server.  
(in "Box to Box" mode the slave or master port)

The display is updated by exiting and then reselecting the menu item *Connection State*.

### → Error State

This list shows the errors which occurred on the port. If more than five errors have occurred since the Com-Server was last reset or since the last time the error table was cleared, the oldest entry is always overwritten. The most recent entry is in first position and the oldest in last position. Preceding each entry is the time when the error occurred in hours and minutes since the last time the Com-Server was reset. You can clear the error table in order to have all the entries available again.

"CTS/DSR/RLSD Time Out"

A timer value can be preset for the Com Server for each of these three serial input signals. This value begins to run when the corresponding input assumes the LOW (-12V) state and is reset as soon as a HIGH level (+12V) is present. If this does not occur within the configured time, this error message is output. The reason may be for example an unconnected, deselected, defective or improperly configured serial terminal device. The factory setting for all inputs is for no timer.

"No halt on XOFF / RTS / DTR"

The connected serial device does not respond to the stop signal set by the Com Server and continues to send data. The result may be overwriting of the serial ring buffer and loss of data. Please check to be sure the handshake configurations for the devices agree and the connection cable is properly wired.

"Overrun Error"

The data register of the serial receiver component was written even though the previous character could not be read out. Since this is a purely device-internal event, this error message is strongly indicative of a Com Server hardware error.

"Parity Error"

A character received on the serial port has an incorrect or missing parity bit. Please check that the transmission parameters for the Com Server and the corresponding serial terminal device are the same. Parity errors can also be caused by use of excessively long cable.

"Framing Error"

A character received on the serial port does not fit in the time frame that results from the configured transmission parameters (baud rate, start bit, data bits, parity bit, stop bits). Please check here also that the Com Server and serial terminal device configurations are the same.

→ **Clear Port Mode**

Use this command to simply and easily revert the port to standard TCP/IP mode. The previously configured TCP/IP mode (TCP-, UDP-, Telnet-, FTP-Client, Box to Box, Bus IP Mode) is deleted.

! The changes made using Clear Port Mode are also *effective without SAVE Setup* after exiting the menu, i.e. they are saved directly in the EEPROM.

## 11 Expanded services of the Com-Server

### 11.1 Control and service ports

In the preceding sections we explained all the standard processes that are implemented in the Com-Server. These client/server processes cover the majority of the application possibilities for the Com-Server. Implementation of more complex scenarios often requires however that the software be individually adapted.

For users who make use of socket programming, the Com-Server offers additional functions beyond pure data transfer.

### 11.2 Serial Port Control (Port number 9094)

The TCP client and TCP server processes for sending data to the serial port of the Com-Server do not allow influencing the serial port itself. But some applications make it necessary that the status and configuration of the port be known and capable of being changed at all times.

Parallel with the data connection on Port 8000, a control connection can be opened on Port 9094. Through this connection it is possible to read out the current status of the port (handshake lines and error states) or to despatch commands. The user data itself is only transported through the actual data connection.

! *The control connection can only be opened if the Com-Server port functions as a TCP server or TCP client.*

The data- and control port (Port 8000 and 9094) are independent of each other. They may be individually opened and closed whenever desired.

#### 11.2.1 The control structure

For the purpose of exchanging information and commands, data records having a defined length and structure are exchanged through this connection.

To request an info structure (completely filled data record from the Com-Server port), you only need to send any desired character to the control port. To despatch a command, the Com-Server expects the complete structure (30 bytes) in a TCP packet. If no allocatable data (e.g. just half of a structure) pass through the control port, the reply is still an info structure. The declaration of the structures is made in C language.

! *Note for all structures:*

The info structure has a fixed length of 30 bytes and is comprised of the following individual structures:

```
#pragma pack(1)    //packs the structure components into 1-byte limits
```

```
typedef struct _rem_box_cntrl
{
    char        zero_1;
    COM_ERROR   _ce;
    COM_STAT    _cs;
    BOX_CNTRL   _bc;
    char        zero_2;
} REM_BOX_CNTRL;
```

```
#pragma pack()
```

### zero\_1/zero\_2:

The two characters *zero\_1* and *zero\_2* are the start and end characters of the structure and must always be zero.

### COM\_ERROR:

The *COM\_ERROR* structure has a length of one WORD (16-bit integer) and contains all the error states of the serial port.

```
typedef struct _com_error
{
    union
    {
        word error_flags;
        struct _err_flags
        {
            word f_data      : 1; //not used/reserved
            word f_net       : 2; //not used/reserved
            word f_com       : 1; //Set when COM port error detected
            word f_break     : 1; //Reflect the break flag
            word f_cts_time  : 1; //Time out while waiting on CTS
            word f_dsr_time  : 1; //Time out while waiting on DSR
            word f_rlsd_time : 1; //Time out while waiting on RLSD (CD)
            word f_overrun   : 1; //Overrun error
            word f_parity    : 1; //Parity error
            word f_frame     : 1; //Framing error
            word f_status    : 1; //not used/reserved
            word no_use_1    : 1; //not used
            word no_use_2    : 1; //not used
            word f_rx_over   : 1; //Ring buffer overrun after handshake
            word no_use_3    : 1; //not used
        };
    };
} COM_ERROR;
```

### COM\_STAT:

The *COM\_STAT* structure has a length of three WORDS (16-bit integer) and contains the status of the handshake lines as well as the contents of the send and receive buffer of the serial port. It also allows you to directly influence the handshake lines and buffer.

```
typedef struct _com_stat
{
    union
    {
        word com_flags;
        struct _com_flags
        {
            word cts_hold      :1; //CTS line           - LowByte
            word dsr_hold      :1; //DSR line           |
            word ri_hold       :1; //not used/reserved  |is set with every
            word rlsd_hold     :1; //reserved/reserved  |received packet
            word dtr_hold      :1; //DTR line           |
            word rts_hold      :1; //RTS line           |
            word x_receive     :1; //XOFF received      |
            word x_send        :1; //XOFF was send      -

            word dummy         :2; //not used           - HighByte
            word send_xoff     :1; //Send XOFF asynchron  |
            word flush_rd      :1; //Flush serial input buffer |
            word flush_wr      :1; //Flush serial output buffer |
            word set_rts_dtr   :1; //set RTS to rts_hold and  |
                                //DTR to dtr_hold           |
            word set_break     :1; //Independent setting break mode |
            word clear_break   :1; //Independent clearing break mode -
        };

        word cbInQue;          //Receive byte count of COM ring buffer
        word cbOutQue;         //Transmit byte count of COM ring buffer
    }
} COM_STAT;
```

### Flushing buffers and influencing the handshake:

- 1) Copy the complete structure of an info packet received by the Com-Server port and fill in the HighByte of the *COM\_STAT* structure.
- 2) All commands whose flags have the value „1“ are executed. If you set the flag *set\_rts\_dtr*, be sure that you also set the flags *rts\_hold* and *dtr\_hold* in the LowByte correspondingly as well.

**!** Depending on how the Com-Server and the *box\_cntrl.f\_flags* structure are configured, the RTS and DTR signals will be used for a LOCK/UNLOCK display or for serial flow control. In this case the Com-Server itself takes over control of these pins. The command flag *set\_rts\_dtr* should therefore be used only for the following setting of the *box\_cntrl.f\_flags* structure:

```
f_rts_disable and f_dtr_disable = 1
f_inx_dtr and f_inx_rts         = 0
```

**BOX\_CNTRL:**

The *BOX\_CNTRL* structure (20 bytes) is used to store the port configuration (baud rate, data bits, parity, stop bits, handshaking, timing values, etc.). Here you can influence the parameters and how they are saved.

```
typedef struct _box_cntrl
{
    char baud;          //Baud rate for channel
                        // 0 = 57600  5 = 4800
                        // 1 = 38400  6 = 2400
                        // 2 = 19200  7 = 1200
                        //20 = 14400  8 = 600
                        // 3 = 9600   9 = 300

    char bits;         //000x.xxxx  data bits, stop bits, parity
                        // 10 = 7 data bits
                        // 11 = 8 data bits
                        // 0 = 1
                        // 1 = 1½, 2 stop bits
                        // 1 = parity enable
                        // 0 = odd, 1 = even parity

    word  RLS_time_out; //Timer before f_rlsd_time will be set
    word  CTS_time_out; //Timer before f_cts_time will be set
    word  DSR_time_out; //Timer before f_dsr_time will be set
    char  XONChar;     //Char excepted as XON
    char  XOFFChar;    //Char excepted as XOFF
    word  hs_on_limit;  //if number of free bytes in ring buffer
                        // > hs_on_limit then clearing handshake stop
    word  hs_off_limit; //if number of free bytes in ring buffer
                        // < hs_off_limit then setting handshake stop
    char  PEChar;      //Replace this char if serial parity error (function
                        //first mus be enabled by setting f_flags.f_pechar=1)

    struct _commands
    {
        unsigned char save_command :4; //Save COM-Configuration
                                        //0 = no save
                                        //1 = save without EEPROM Update
                                        //2 = save with EEPROM Update

        unsigned char clear_error :1; //1 = clear error in display/lamps
        unsigned char set_fact_def :1; //1 = set factory defaults and reset
        unsigned char free_cmd :2; //not used
    };

    union
    {
        word hs_flags;
        struct hs_flags
        {
            word f_cts_connect :1; //Connect/Disconnect with CTS (HIGH/LOW)
            word f_dsr_connect :1; //Connect/Disconnect with DSR (HIGH/LOW)
            word f_cts_accept :1; //Accept Connection only by CTS=HIGH
            word f_dsr_accept :1; //Accept Connection only by DSR=HIGH
            word no_use0 :12; // not used
        };
    };
};
```

```

};
union
{
  word f_flags;
  struct _f_flags
  {
    word f_rts_disable :1; //RTS will not change at LOCK/UNLOCK
    word f_dtr_disable :1; //DTR will not change at LOCK/UNLOCK
    word f_outx        :1; //Enable softw. handshake while sending
    word f_inx         :1; //Enable softw. handshake while receiving
    word f_outx_cts    :1; //Enable hardware handshake on CTS
    word f_outx_dsr    :1; //Enable hardware handshake on DSR
    word f_inx_dtr     :1; //Enable hardware handshake on DTR
    word f_inx_rts     :1; //Enable hardware handshake on RTS
    word f_parity      :1; //Enable parity check & error report
    word f_pechar      :1; //Enable replacement of received char
    word f_inxfilter   :1; //Enable xon/xoff filter while receiving
    word f_outxfilter  :1; //Enable xon/xoff filter while sending
    word f_rts_default :1; //1 = While RTS is not used, RTS is active
    word f_dtr_default :1; //1 = While DTR is not used, DTR is active
    word f_user_time   :1; //not used
    word clr_err_char  :1; //1= If Com-Server is in client mode,
                          //serial chars with framing or parity
                          //errors will not open the connection
  };
};
} BOX_CNTRL;

```

### Configuring the serial port:

- 1) Copy the complete structure of an info packet received by the Com-Server and fill in the BOX\_CNTRL structure. This allows you to read out the default values and only enter the values you want to change.
- 2) To work with the configuration, the value *save\_command* in the structure has to be set. "save\_command = 2" causes the Com-Server to save this configuration in non-volatile memory and to use it again after a restart. A 1 does not overwrite the non-volatile memory, i.e. after a restart the old configuration is used.

### Functions of the RS232 outputs RTS and DTR

If the flags *f\_rts\_disable* and *f\_dtr\_disable* are set to 0, the corresponding RS232 output uses an enable level to signal to signal an active connection between the serial port and a client in the network. If the Com-Server itself is configured as a TCP client, a constant enable level is output, regardless of whether there is an active connection or not.

Alternately a 1 in the flags *f\_inx\_rts* and *f\_inx\_dtr* can assign the function of serial flow control to the respective RS232 output. If flow control is enabled, the corresponding flag for the *LOCK/UNLOCK* display must have the value 1 (= *LOCK/UNLOCK* display disabled).

If the respective RS232 output does not have the function of either *LOCK/UNLOCK* display or flow control, the flags *f\_rts\_default* and *f\_dtr\_default* can be used to specify the default state after a Com-Server reset (1=Enable, 0=lock level).

### 11.3 Reset Com-Server port (Port number 9084)

For special applications this port makes it possible to carry out a forced reset of the Com-Server port: The parameters for the current connection are deleted, and the current connection partner (host) is notified by means of a *TCP/IP RST*.

**Example:**

The TCP/IP station having address 172.16.231.100 has a connection to the TCP server (Port 8000) of the Com-Server having IP address 172.16.231.8. The host fails or is disconnected due to a cable break. The Com-Server port remains blocked as long as the TCP/IP station does not report.

To reset this port for a new connection, a *connect()* must be made to Port 9084 of the Com-Server. The Com-Server will accept the connection and immediately initiate a *close()* (close the connection). At the same time it sends a RESET signal to TCP/IP station 172.16.231.100 and deletes its parameters.

The next *connect()* to Port 8000 is now replied to by the Com-Server.

- ! *When invoking this service, the contents of the serial in- and output buffer are lost. The port can be opened by any desired station and should therefore only be used in extreme cases! No data are permitted to be transported through this connection! The port is opened by a host and then immediately closed by the Com-Server.*

### 11.4 Software reset of the Com-Server (Port number 8888)

Socket 8888 has been implemented in case the Com-Server ever needs to be completely reset. If a connection is opened on this port, the Com-Server immediately closes it again and then performs a software reset.

- ! *Following this reset all the buffer contents are deleted and any still active connections are closed - the Com-Server is in its base state! This reset can be performed from any station and should be done only in extreme cases!*

## 11.5 Uploading/downloading configuration data (Port numbers: 8003/read, 8004/write)

These services allow you to read out the configuration data for the Com-Server stored in the EEPROM and send it to another Com-Server. Especially when installing a large number of Com-Servers having the same settings this method eliminates having to open a Telnet configuration connection for each individual unit.

### Reading out the configuration data on Port 8003

All the client application needs to do is open a TCP connection to Port 8003 of the Com-Server. The Com-Server accepts it, automatically sends its 512-byte long configuration and then closes the connection to the client.

The client application can store this data now in a binary file and use it for downloading to other Com-Servers.

### Writing configuration data on Port 8004

Uploading to a Com-Server is analogous to reading out the configuration data. After opening a TCP connection to Port 8004 of the Com-Server, the latter expects the 512 bytes for a new configuration. After the client closes the TCP connection, the Com-Server saves the data in its EEPROM and performs a reset.

The value of the IP address remains unaffected by a configuration upload.

### Application

To operate multiple Com-Servers having the same configuration, you must first assign each unit with its own IP address. Then use Telnet to fully configure a Com-Server with the desired parameters and use Port 8003 to read the EEPROM out. The file thus created can now be uploaded to the remaining Com-Servers.

! *The procedure described here is a substitute for the relatively time-consuming configuration using a Telnet session. All the settings are saved in the EEPROM and are still available after a reset or power loss. Only a factory default reset overwrites the settings assigned by the factory set standard values (see „Network-Firmware update under Windows“).*

### 11.5.1 Uploading/downloading the configuration under Windows

For Windows computers you can also up- or download the configuration data using the W&T tool, which is also used for the firmware update. You can find a corresponding download link on the Web data sheet pages for the Com-Servers at the following address: <http://www.wut.de>

#### Reading out the configuration data

As in the case of direct uploading and downloading using sockets, here also you must first use Telnet to completely configure a Com-Server to the desired parameters.

After starting the tool the menu path „CS programming → Make a \*.cfg file“ will bring you to the input screen for creating a configuration file. Enter the IP address of the Com-Server from which you want to read the configuration data in the appropriate field. If the specified Com-Server is accessible, the input fields in the middle are automatically filled in. Now enter the file name under which you want to save the configuration and then click on the button „Write file“.

! *By activating the option „Apply generic settings“ the tool replaces the IP address, subnet mask and gateway address with the default values **after** the configuration data are downloaded.*

**Writing the configuration data**

In order to load an existing configuration file into a Com-Server, the latter must already have an IP address. After starting the update tool the menu path „CS programming → EEPROM“ brings you to the input screen for a configuration upload. After specifying the IP address and the desired file name, click on the „Start“ button to send.

- ! *The upload overwrites all the Com-Server settings with the contents of the sent configuration file. Only the IP address remains unchanged.*

**Factory default reset**

Clicking on the „Reset“ button in the input screen for the configuration upload resets the specified Com-Server to its factory default settings. Any previously uploaded configuration file thus becomes invalid.

## 11.6 Inventory taking per UDP

Larger installations often require that an inventory be taken of all the Com-Servers currently in the network. This can be done using UDP Port 8512. After receiving a network packet directed to this port, the Com-Server replies to the respective sender with an info packet. This contains the network-relevant basic parameters as well as information such as the current connection status.

Whether a packet was received by a Com-Server per broadcast or directly addressed to it makes no difference in generating the info structure. The only factor is that UDP-Port No. 8512 is used.

### 11.6.1 The Info packet

Each info packet consists of 2-5 data structures. *UDP\_BOX\_INFO* contains all the data specific to the Com-Server such as MAC and IP address. The 2nd-5th structure *PORT\_DESCRIPTOR* can be used to obtain the specified mode or current connection status for the individual ports. The total length of the two structures is derived as follows:

Length: **(22 + (port\_no × 10)) Bytes**

```
typedef struct _UDP_BOX_INFO    // Parameter in Hostorder = Low byte first
{
    unsigned char mac_addr[6];  // MAC address of Com-Server
    unsigned long IP_number;    // IP address of Com-Server
    unsigned long gateway;     // Gateway
    unsigned long subnet_mask; // Subnet Mask
    unsigned int  MTU;         // MTU
    unsigned int  port_anz;    // Number of ports in Com-Server

    PORT_DESCRIPTOR pd[port_anz];
} UDP_BOX_INFO;

typedef struct _PORT_DESCRIPTOR
{
    unsigned int  state;        // 0=FREE, 1=CONNECT, 3=WAITES
    unsigned int  mode;        // 0x0001 = TCP Client
                                // 0x0002 = TELNET Client
                                // 0x0003 = FTP Client
                                // 0x0004 = Box 2 Box Client (aktiv)
                                // 0x0005 = UDP-Send/Receive Mode
                                // 0x0008 = Box 2 Box Server (passiv)
                                // 0x0010 = SLIP-Router
                                // 0x0020 = PPP-Router (in development)
                                // 0x0030 = Box2Box Bus System: SlaveBox
                                // 0x0040 = Box2Box Bus System: MasterBox

    unsigned long remote_IP;   // if state==CONNECT, else 0
    unsigned int  remote_port; // if state==CONNECT, else 0
} PORT_DESCRIPTOR;
```

### **UDP\_BOX\_INFO**

Each info packet begins with the structure *UDP\_BOX\_INFO*, which has a fixed length of 22 bytes. The first 6 bytes contain the Ethernet or MAC address of the Com-Server in the usual convention: 3 bytes manufacturer's ID (Com-Server = 00:c0:3d) followed by 3 bytes of unique device ID.

The following 14 bytes contain in order the values for IP address, gateway address, subnet mask and MTU.

The last two bytes contain the integer value with the number of serial ports available on the Com-Server. The value determines the number of following structures of type *UDP\_BOX\_INFO* and thereby the total length of the info packet.

### **PORT\_DESCRIPTOR**

The info packet contains a 10-byte long structure of type *PORT\_DESCRIPTOR* for each serial port on the Com-Server. From this you can obtain the current connection status as well as the configured mode and respective communications partner.

! All variables of types „Integer“ and „Long“ are mapped in host order, i.e. the low-value bytes come first. The IP address 172.17.2.3 appears for example in byte order 3, 2, 17, 172.

*UDP packets are connectionless and unsecure datagrams. Especially when using broadcasts the Com-Server's own request as well as reply packet can get lost. To reliably ascertain all the Com-Servers installed in a subnet you should therefore repeat the broadcast if necessary.*

## 11.7 SNMP management

Many networks are managed through a central network management system via SNMP protocol. A full description of the Management Information Base (MIB) of the Com Server would exceed the limits of this Manual. Documentation including the ASN.1 file is available for downloading at our Web site <http://www.wut.de>.

Before downloading please write down the 5-digit model number of your Com Server. From the homepage you can then go to the product overview sorted by article numbers, which then allows you to select the Web data sheet for your particular Com Server model. Follow the link to the current version of the document.

## 12 Firmware update of the Com-Server

Since the operating software is under constant development, this device also allows you to perform a firmware update. The following upload variations are available depending on your computer environment.

- Firmware update through the network under Windows 9x/NT/2000
- Firmware update through the network with TFTP under UNIX
- Firmware update through the serial interface

### 12.1 Where do I get the current firmware?

The most current firmware including update tools and a revision list is published on our Web site at the following address: <http://www.wut.de>

Please write down the 5-character model code found on the Com-Server before downloading. From the homepage you can access the product overview sorted by article numbers, which then takes you directly to the Web data sheet for the respective Com-Server model. Follow the link to the current version of the firmware.

- ! ***Never interrupt the update process by pulling the power plug or pressing the reset button. The Com-Server will be non-operational after an incomplete update. Never mix files having different version numbers in the file name. This will cause the unit to malfunction. Send all the files one after the other. The Com-Server recognizes when all the files have been sent and that the new operation software is complete. It then performs an autonomous reset.***

## 12.2 Network firmware update under Windows 9x/NT/2000

Required is a PC running Windows with a network connection and activated TCP/IP stack. For the update process you need two files which, as already mentioned, are available for download from our Web site at <http://www.wut.de>.

- The executable update tool for transferring the firmware to the Com-Server.
- The file with the new firmware for transferring to the Com-Server.

The update process is described in the following steps. Please check our Web site for any possible supplementary notes.

1. Close **all** connection which may be still active on the Com-Server. The update process flushes all buffers, thereby deleting all data!
2. Start a Telnet session on the configuration port of the Com-Server.

```
telnet [IP address] 1111
```

In the *SETUP* menu select *System* → *Flash Update* → *Net Update*, and confirm with *y*. The Com-Server then closes the network connection and the corresponding Telnet session must be confirmed with „OK“. The green Status LED indicates that the Com-Server is now in update mode.

3. Now start the update tool. The menu path *CS programming* → *Flash* takes you to the input screen for uploading new firmware.
4. Enter the IP address of the Com-Server as well as the name of the firmware file in the appropriate fields. In the Options field „Output“ please activate only the item „Firmware“ (see „Customer-specific factory defaults“).
5. Now click on the *Start* button. The update will take a few seconds. When it is finished, a message window will display a corresponding message.
6. Check the configuration window of the Com-Server to verify that the new operating software was added. The new version number of the firmware should be indicated in the *INFO System* → *SOFTW Date/Rev* menu.

If the previous version is still shown, the file with the new operating software is corrupted. Please contact your dealer.

! *The procedure described here for the update applies only to firmware Version 1.14 and higher. Com-Servers with a lower version should be updated to at least 1.14. Note that **only** the destination IP address as well as the name of the new firmware are allowed to be entered in the input screen for the update. However both „Firmware“ and „Configuration“ **must** be activated in the Options field.*

### 12.2.1 Customer-specific factory defaults

When downloading new firmware using the update tool it is possible also to send a configuration file to the Com-Server. This file may contain a special setup profile deviating from the standard (e.g. serial parameters, operating modes etc.). In contrast to the procedure described in the section „Up-/Downloading configuration data“, the settings sent here are used as new factory defaults for the Com-Server.

#### Creating a configuration file

As described in „Up-/Downloading configuration data“, first you must read out a configuration file with the desired configuration profile from a Com-Server. Note especially the values for IP address, subnet mask and gateway address indicated in the Options block „Location parameters“.

The update tool writes these to the cfg file, after which they are used as factory defaults for the upload described here.

### **Transferring the configuration file including new firmware**

After starting the update tool, the menu path *CS programming* → *Flash* takes you to the input screen for uploading new firmware. In addition to the destination IP address for the Com-Server please enter also the names of the new firmware and configuration file in the appropriate fields. „*Firmware*“ **and** „*Configuration*“ must be activated in the „*Output*“ option field before you click on the „*Start*“ button. After successful transfer the desired settings can now be activated by a factory default reset of the Com-Server.

### **Transferring only the configuration file without new firmware**

After starting the update tool, the menu path *CS programming* → *Flash* takes you to the input screen for uploading new firmware. In addition to the destination IP address of the Com-Server, please enter only the name of the configuration file in the corresponding field. **Only** „*Configuration*“ may be activated in the „*Output*“ option field before you click on the „*Start*“ button. After successful transfer the desired settings can now be activated by a factory default reset of the Com-Server.

! *Configuring customer-specific factory defaults is only possible with firmware Version 1.14 and higher. Com-Servers having a lower version must first be updated to at least Version 1.14. Note here that **only** the destination IP address and the name of the new firmware may be entered in the input screen for the update. Both „*Firmware*“ and „*Configuration*“ **must** be activated in the options field however.*

## 12.3 Network firmware update under UNIX

Required is a computer with a network connection and a TCP/IP stack that provides Telnet and TFTP network protocols. When updating with TFTP protocol the firmware is split into 4 files that are sent to the Com-Server one after the other.

1. Start the remote configuration tool of the Com-Server over Telnet.  

```
telnet [IP address] 1111
```

 In the *SETUP*: menu select *System* → *Flash Update* → *Net Update* and confirm with „y“. The Com-Server closes the Telnet connection. The green Status LED indicates that the Com-Server is in update mode.
2. Now use the TFTP command to send the first file in binary mode to the Com-Server. While the data are being sent over the network the Status LED will be on. Then the Com-Server goes into programming mode and the Error LED comes on. This process may take several seconds. Wait until the Error LED goes off and the Status LED comes on again. Repeat this process for all four files.
3. The Com-Server recognizes when all files have been sent and independently does a restart. If the green Status LED comes on again after all the files have been sent, repeat Step 2 again in full. Sending a file twice does not result in an update error. The Com-Server waits until all the necessary files have been sent.

### Example: SCO UNIX

Enter the following commands after the corresponding prompt:

```
# tftp
tftp> connect [ip_number/host_name]
tftp> binary
tftp> put C4r1_1.4_1 [remote filename](remote filename = any letter)
```

Now wait until the green Status LED comes on again. Then use the *put* command to send the remaining files, and close the TFTP connection.

```
tftp> quit
#
```

4. Check the configuration menu of the Com-Server to make sure it took the new operating software. The version number of the new firmware should now be seen in the *INFO Com-Server* → *SOFTW Date/Rev* menu.

If the previous version is still displayed, then one or more of the files with the new operating software is corrupted. Please use our Hotline to get in touch with us.

## 12.4 Firmware update through the serial port

The only requirement is a computer having a configurable serial terminal. As in the case of updating per TFTP, here again the firmware is split into 4 files sent one after the other to the Com-Server. The update process is described in the following steps. Please follow these steps exactly.

1. Connect the serial port of the Com-Server to the serial port of the computer from which you want to perform the update.
2. Configure the computer's serial port as follows:  
*9600 baud, no parity, 8 bits, 1 stop bit*
3. Start the remote configuration tool of the Com-Server per Telnet:  
`telnet [IP address] 1111`  
Select *SETUP System* → *Flash Update* → *Serial Update* in the configuration menu and confirm with *y*. The Com-Server closes the Telnet connection. The green Status LED indicates that the Com-Server is now in update mode.
4. Now use the COPY command to send the first file to the Com-Server. While the data are being sent over the serial interface the Status LED flashes rhythmically. This may take up to 3 minutes. Then the Com-Server goes into programming mode and the Error LED comes on. Wait until the Status LED comes on again and stays on. Repeat this procedure for all four files.
5. The Com-Server recognizes when all the files have been sent and automatically performs a reset. If the Status LED comes on again after all the files have been sent, repeat Step 3 in full. Sending a file twice does not result in an update error. The Com-Server waits until all the necessary files have been sent.
6. Check the configuration menu of the Com-Server to make sure it took the new operating software. The version number of the new firmware should now be seen in the *INFO Com-Server* → *SOFTW Date/Rev* menu.

If the previous version is still displayed, then one or more of the files with the new operating software is corrupted. Please use our Hotline to get in touch with us.

## APPENDIX A

### TCP/IP under Windows 9x

To activate TCP/IP you need a computer having a network card and Windows 9x. TCP/IP is bundled with Windows 9x, so it only needs to be configured for the local network.

- Obtain a valid IP address from your network administrator.
- Go to the *Start* menu and select the submenu *Settings* → *Control Panel* → *Network*.
- Select the *Configuration* tab and *Add*, then *Protocol* and *Manufacturers: Microsoft* and *Protocol: TCP/IP*.
- Click on *OK*. This returns you to *Configuration*, and TCP/IP will now appear in the list of network components.
- Select *TCP/IP*, and click on *Properties*. On the *IP Address* tab enter the IP address for this computer as well as the subnet mask of the network in which it is located.

If you also want to have connections to other networks, use the *Gateway* tab to enter the IP address or the gateway through which you want to route your connections.

If you have only a local network and are not using gateways or routers, a few simple rules will apply to the subnet mask. Note the first place in the IP address and select the subnet mask accordingly:

1 – 126:	255.0.0.0
128 – 191:	255.255.0.0
192 – 254:	255.255.255.0

- Confirm with *OK*. You are now prompted to insert the Windows 9x CD. The drivers are updated, after which you must restart the computer in order to activate the new configuration.

---

## APPENDIX B

### TCP/IP under Windows NT

To activate TCP/IP you need a computer having a network card and Windows NT. TCP/IP is bundled with Windows 9x, so it only needs to be configured for the local network.

- Obtain a valid IP address from your network administrator.
- Go to the *Start* menu and select the submenu *Settings* → *Control Panel* → *Network*.
- Select the *Protocols* tab and *Add*, click on *Add* and then select *TWCP/IP Protocol* from the list.
- Click on *OK*. You will be asked whether you want to use a DHCP server for assigning the IP address. Check with your network administrator, or answer with *NO*. Now insert your Windows NT CD to install the required drivers.

After this you will be back on the *Protocols* tab. In the list of network components you will now find the entry *TCP/IP Prptocol*. Now click on *Close*!

- The links are updated and the properties window for TCP/IP appears. Enter here an IP address which is valied for this network. The subnet mask. is automatically determined from the IP address.

If you want to have connections to other networks as well, you must also etner the subnet mask specified for your network and a standard gateway. Obtain this information as well from your system administrator.

- Confirm with *OK* and restart the computer to activate the configuration.

## APPENDIX C

### Overview of the Port and Socket numbers factory set in the Com-Server

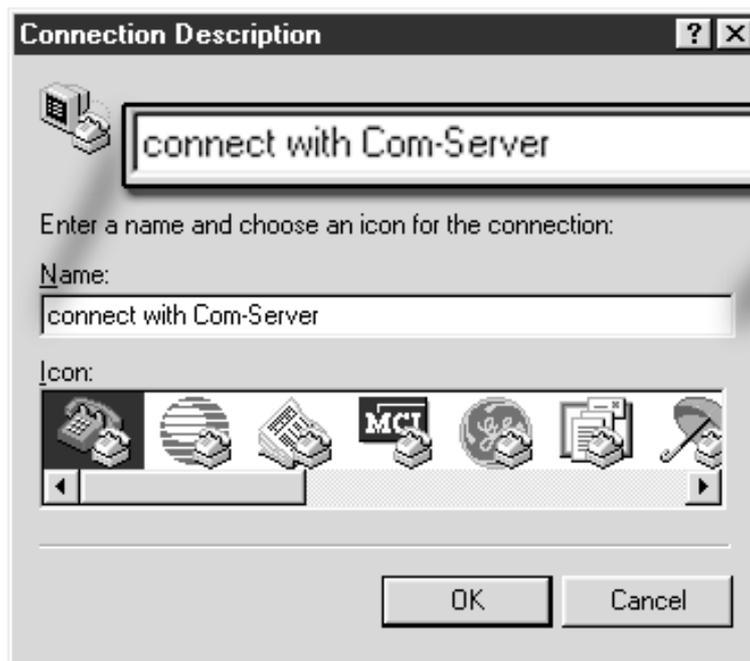
Port/Socket number	Application
6000, 23	TELNET Server Port A
7000, 21	FTP Server Port A
8000	Data Sockets (Client und Server) Port A
1111	TELNET configuration port
9094	Control port Port A
9084	Reset Port Status via Socket Port A
8888	Reset Com-Server
8003	Read configuration data via socket
8004	Write configuration data via socket
UDP 8512	Retrieve an info packet

## APPENDIX D

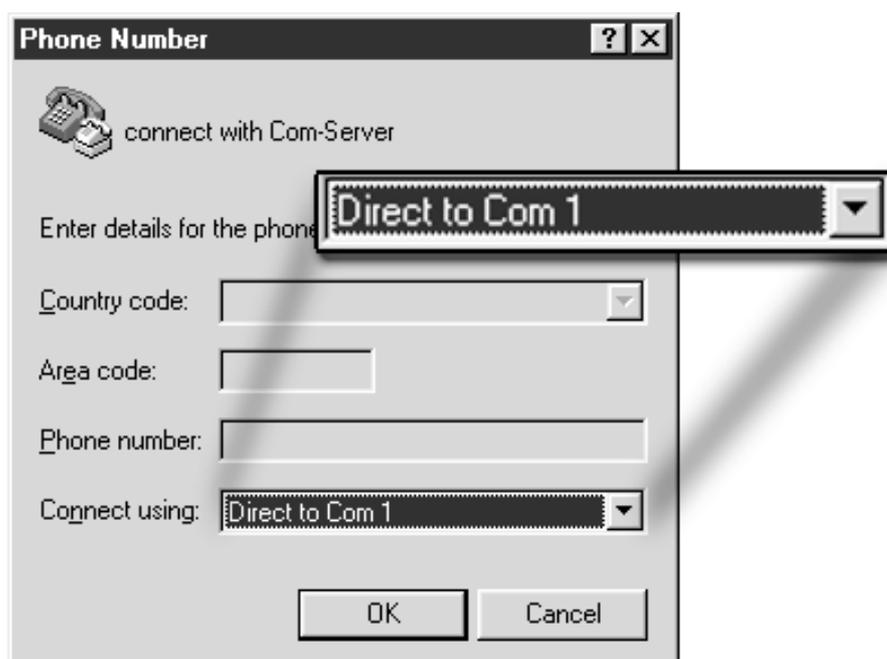
### Serial assignment of the IP address under Windows

To serially assign the IP address you can use Hyperterminal, which is bundled with Windows. Start this procedure under *Start → Programs → Accessories → Hyperterminal*.

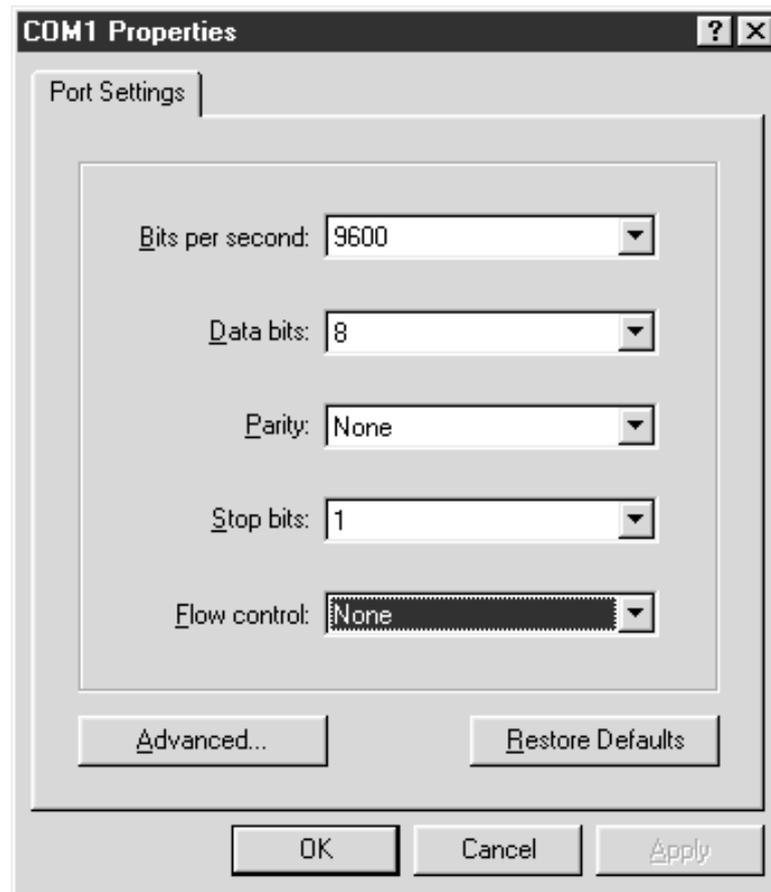
- 1.) In the first window you assign a name for the connection you wish to build. this allows you to directly start Hyperterminal with the correct transmission parameters the next time you want to assign an IP address:



- 2.) In the lower selection box of the following window please select only the COM port to which the Com-Server is connected.



- 3.) The transmission parameters are 9600 baud, 8 data bits, no parity, no protocol:



- 4.) After you have confirmed the settings with OK, you will find yourself in the actual terminal window. All keyboard entries are output through the selected COM port. Now turn the Com-Server on and off to reset it and while it is starting up hold down the „x“ key (lower case, no Caps Lock!). After about 2-3 seconds the prompt „IP no.:+<ENTER>:“ will appear in Hyperterminal.

Now enter the IP address you want to assign in the usual dot notation (e.g. 172.17.10.10). There will be no immediate echo of characters, so the individual inputs will not be seen on the monitor. Only after you have pressed Return to confirm is the completely entered IP address returned by the Com-Server.

If you have made improper entries or typing mistakes, the Com-Server will return „FAIL“ followed by its actual current IP address. Step 4 must then be repeated.

## APPENDIX E

### Web applications HTTP, SMTP, POP3 ...

TCP/IP protocol is the basis for all applications used in the Internet. As in the case for example of HTTP, SMTP or POP3 these are often relatively simply constructed, non-time critical ASCII protocols which are sent within the TCP data range. Distinctions are made among the individual services by using the respective TCP port number. A Web browser uses for example HTTP protocol under port number 80 for connecting to a Web server.

The freely configurable local port number of the Com-Server (see „Configuring the TCP/IP modes“) allows your own serial devices to be made Web-compatible with relatively little programming effort. The complicated lower protocol layers Ethernet, IP and TCP are fully handled by the Com-Server. Higher order protocols contained here are sent transparently to the serially connected device for further processing.

#### HTTP example

When a Web browser requests a particular site, the sequence as far as the network is concerned is as follows:

- Resolve the specified URL into the IP address using DNS
- Build and establish a TCP connection on Port 80 (=HTTP)

Both steps are done without any help from the serial terminal device using the TCP/IP stack of the Com-Server. Only now is the desired Web site retrieved using HTTP protocol, which the Com-Server passes along transparently. Specifically the serial device receives the following string:

```
HTTP 1.1 GET /filename [CR/LF]
n Options [CR/LF]
[CR/LF]
```

In the simplest case it is now sufficient to return the desired data after processing the file name and then close the TCP connection. There are two ways to close the connection: per RS232 control line (see „Configuring the RS232 parameters“ (Menu: UART Setup)) or using a timeout (see „TCP Client mode“ (Menu: TCP Client)).

For additional information and examples for integrating your own serial devices into the Web using the Com-Server, see our Web site at <http://www.wut.de>.

## APPENDIX F

### Technical Data

<b>Supply voltage</b>	
Type 58211	typ. 220mA, max. 270mA
Type 58411	typ. 200mA, max. 250mA
Type 58412 @ 5V/DC	typ. 200mA, max. 250mA
Type 58221	typ. 495mA, max. 590mA
Type 58421	typ. 475mA, max. 580mA
Type 58611 @ 24V/DC	typ. 65mA, max. 90mA
Type 58621 @ 24V/DC	typ. 160mA, max. 220mA
<b>Permissible ambient temperature (mounted with free air convection)</b>	
Type 58211, 58221, 58611, 58621	0–60°Celsius
<b>Permissible ambient temperature (mounted in rows without spacing)</b>	
Type 58211, 58611	0–60°Celsius
Type 58221, 58621	0–50°Celsius
<b>Permissible ambient temperature OEM models (ambient temperature of the board)</b>	
Type 58411, 58412, 58421	0–70° Celsius
<b>Permissible relative humidity (non condensing)</b>	
all models	0–95%
<b>Dimensions</b>	
Type 58211, 58221, 58611, 58621	105 x 75 x 22mm
Type 58411, 58421	see sec. 3.2
Type 58412	see sec. 3.3
<b>Weight</b>	
Type 58211, 58221, 58611, 58621	ca. 150g
Type 58611, 58621	ca. 100g

**Symbols**

100BaseT 16  
10BaseT 16  
2-wire bus 20  
4-wire bus 20

**A**

APPE 48, 49  
ARP 7  
ASCII 47

**B**

base configuration 27  
baud rate 33, 63  
BOOTP 8, 29  
Box to Box 51  
box to box 52  
BOX\_CNTRL 63

**C**

clear port mode 59  
client process 32  
COM Port Redirector (Windows) 54  
configuration data 66  
configuration menu 27, 33  
configuration of serial port 33  
connection status 58  
connection timeout 40, 48  
connections 16  
control port 60  
control structure 60  
CTS 35, 36

**D**

data bits 33, 63  
DHCP 8, 10, 29  
disconnect char 40, 44, 45  
dispatch 40, 43, 51  
DSR 35, 36  
DTR 35, 36, 64

**E**

error state 58  
ethernet connection 16

**F**

factory defaults 30, 72  
firmware 28, 71  
firmware update 30, 71  
flow control 36  
form factors  
    industry model 13  
    OEM model 14  
framing error 25  
FTP 47  
FTP client 47  
    login 48  
    with serial protocol 49  
FTP commands  
    APPE 48, 49  
    ASCII 47  
    GET 47  
    IMAGE 47  
    LIST 48, 49  
    PUT 47  
    QUIT 47  
    RETR 48, 49  
    STOR 48, 49  
    TYPE A 48, 49  
    TYPE I 48, 49  
FTP server 47

**G**

gateway 29  
GET 47

**H**

handshake 25  
    hardware handshake 35  
    NO handshake 35  
    software handshake 35  
handshake lines 62  
HTTP 81

**I**

IMAGE 47  
inactivity timeout 40, 45, 48  
info packet 68  
info System 28  
interface parameters 33  
inventory taking (UDP) 68  
IP address 29

assignment of 79  
IP bus mode 53

## L

LED displays 25  
error LED 25  
power LED 25  
status LED 25  
LIST 48, 49

## M

MAC address 28  
master port 51  
Master-Slave bus 53  
MTU 29

## P

parity 33, 63  
parity error 25  
password 30  
POP3 81  
port state 58  
PUT 47

## Q

QUIT 47

## R

RARP 9  
receive filter 37  
reset 30, 65  
Reset-Pin 22  
response mode 41  
RETR 48, 49  
router 29  
RS232 17  
cable 8  
interface 17  
RS422/485 interface 19  
RTS 35, 36, 64

## S

save setup 28  
send filter 37  
serial port 8  
serial socket interface 44

server process 32  
setup system 29  
show connection 36  
slave port 51  
SLIP router 55  
SMTP 81  
SNMP 70  
socket number 38, 78  
sockets 39  
stop bits 63  
STOR 48, 49  
subnet mask 29  
supply voltage 24

## T

TCP  
client 39  
port number 38  
server 39  
TCP/IP mode 38  
Telnet  
client 45  
configuration 26  
terminating 20  
timeout  
connection t. 40, 48  
inactivity t. 40, 45, 48  
TYPE A 48, 49  
TYPE I 48, 49

## U

UART setup 35  
UDP 39, 43  
client 43  
portnummer 43

## W

wiring examples 21

## X

XON/XOFF 36

**RS232/RS422/RS485 Interface Modules, #188x3****Function**

The W&T Interface Modules 18803, 18813 and 18833 provide an RS232, RS422 or RS485 interface for devices equipped with a serial TTL port. The Interface Module is configurable for the various operating modes via an 8-position DIL switch. Modules 18803 and 18813/18833 differ only in the arrangement of the post connector for the serial TTL port, and are in every other respect functionally identical.

**Important installation note**

When installing or replacing the Interface Modules, visually inspect to ensure that the module does not cause a short circuit with adjacent components.

To use the Interface Modules with W&T PC cards, remove the wire jumper between the SUB-D male connector housing and the module. Otherwise the galvanic isolation between the interface and the PC may be defeated.

**Supply voltage**

The Interface Modules require a regulated supply voltage of 5V DC  $\pm 5\%$ . (18803 and 18813) and 3,3V DC  $\pm 5\%$  (18833) respectively. The no-load current draw of the modules is approx. 40mA (typ.); any additional current requirement from an external load must of course be taken into account when dimensioning the power supply.

**Protection against static charges**

All external signal lines use ESD-immune interface components to provide protection against static discharges of up to 15kV per IEC 801-2, Level 4.

## Pin configuration

The RS232/RS422/RS485 connection for the modules is configured as a 9-pin male SUB-D connector, with the TTL interface formatted as 12-pin male post connector. Refer to the following table for connector pin assignments:

### TTL interface

pin#	signal	RS232	RS422/485
1	5V $\pm$ 5%	Vcc	Vcc
2	RI	output	"low" level
3	RxD	output	output
4	TxD	input	input
5	n.c.	n.c.	n.c.
6	CTS	output	output
7	DTR	input	input
8	DSR	output	"low" level
9	RTS	input	input
10	DCD	output	"low" level
11	12V $\pm$ 10%	n.c.	n.c.
12	GND	signal GND	signal GND

*Pin 1 of the TTL interface is indicated by a rectangular soldering pad.*

### RS232 interface

pin#	signal	function
1	DCD	input
2	RxD	input
3	TxD	output
4	DTR	output
5	GND	GND
6	DSR	input
7	RTS	output
8	CTS	input
9	RI	input

### RS422/RS485 interface

pin#	signal	function
1	TXD A	output
2	RxD A	input
3	DTR A	output
4	CTS A	input
5	GND	GND
6	TXD B	output
7	RxD B	input
8	DTR B	output
9	CTS B	input

## Operating modes

The Interface Modules are configurable for the following modes using DIL switches:

### RS232

The Interface Module converts all available TTL data and handshake signals into RS232 signals. This mode provides one Rx/D and Tx/D channel each in the respective direction, along with six handshake channels (RTS, CTS, DSR, DCD, DTR and RI).

### RS422

The Interface Module supports one data and one handshake channel each (selectable DTR or RTS handshake output). The RS422 sender/receivers are always active.

### RS485

One data channel in each direction is always available in all RS485 modes. These modes differ only in how the RS485 transceivers are controlled.

#### RS485 4-wire bus master

In this mode the master uses a conductor pair to send requests to the slaves, which in turn send their replies to the master on an additional common conductor pair. The RS485 transceivers are always active in this mode, whereby the master can always send and is constantly listening for the slaves.

**RS485 4-wire mode with handshake control**

The RS485 driver chip is turned on with a TTL Low level on the „DTR“ or „RTS“ line, whereas a TTL High level on this line puts the driver in a high-impedance state. The receiving channel in this mode is always active.

**RS485 2-wire mode with handshake control**

The RS485 driver chip is turned on with a TTL Low level on the „DTR“ or „RTS“ line, whereas a TTL High level on this line puts the driver in a high-impedance state. The receiving channel is deactivated when the driver is on, and enabled when the driver is in the high-impedance state.

**RS485 4-wire mode with automatic control**

The RS485 driver chip is automatically activated whenever data are output and brought to the high-impedance state when data output is finished. The receiving channel in this mode is always active.

**RS485 2-wire mode with automatic control**

The RS485 driver chip is automatically activated when data are output and brought to the high-impedance state when data output is finished. The receiving channel is deactivated when the driver is turned on, and enabled when the driver is in the high-impedance state.

The DIL switch settings can be found in the following table:

Operating mode	SW1	SW2	SW3	SW4	SW5	SW6	SW7	SW8
RS232	OFF	ON						
RS422, RS485, 4-wire bus master DTR handshake	OFF	OFF	OFF	ON	OFF	*	*	OFF
RS422, RS485, 4-wire bus master RTS handshake	OFF	OFF	OFF	OFF	ON	*	*	OFF
RS485, 4-wire / 2-wire with echo DTR control	OFF	OFF	ON	ON	OFF	*	*	OFF
RS485, 2-wire without echo DTR control	ON	OFF	ON	ON	OFF	*	*	OFF
RS485, 4-wire / 2-wire with echo RTS control	OFF	OFF	ON	OFF	ON	*	*	OFF
RS485, 2-Draht without echo RTS control	ON	OFF	ON	OFF	ON	*	*	OFF
RS485, 4-wire / 2-wire with echo automatic control	OFF	ON	OFF	ON	OFF	*	*	OFF
RS485, 2-wire without echo automatic control	ON	ON	OFF	ON	OFF	*	*	OFF

*\*) Terminating the bus system when required.*

### Important Note

The terminating DIL switches SW6 and SW7 must never be in the ON position when using the module in RS232 mode. This will result in a significant increase in the current draw and may cause the RS232 driver to fail.



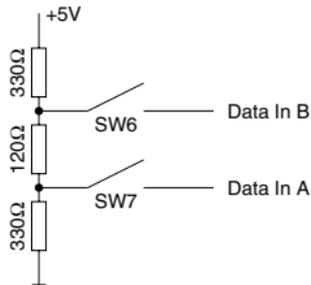
## Termination

All RS485 modes require termination of the bus system with a termination network. The resistor combination integrated in the module performs two tasks in RS485 applications:

1. The connected line is terminated corresponding to its impedance, which prevents signal reflections at the cable end.
2. In the high-impedance phases of bus operation a defined quiescent state is ensured.

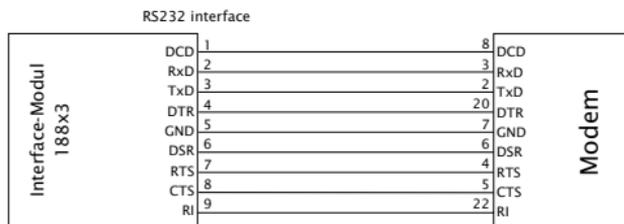
The bus system is allowed to be terminated with the termination network only in RS485 and RS422 modes, but not in RS232 mode.

Closing DIL switches 6 and 7 on the Interface Modul connects the bus terminals to the following resistance network:

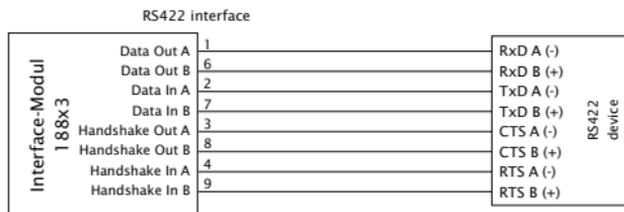


## Connection examples

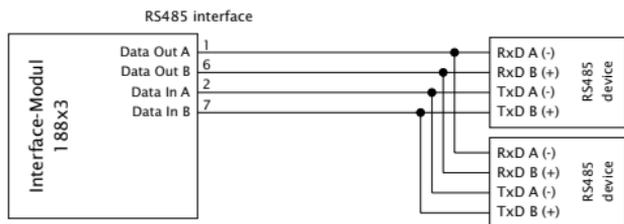
### RS232 application with hardware handshake



### RS422 application with hardware handshake



### RS485 4-wire bus master application



### RS485 2-wire application

