# Manual
## Com-Server Highspeed

**W&T**

**W&T**

**Introduction**

The Com-Server models 58631, 58031, 58034, 58231, 58431, 58432 58331 and 58334 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 Highspeed models**

| Model No. | Network interface | Supply voltage | Serial interface | Housing |
|---|---|---|---|---|
| 58631 | 10/100BaseT autosensing | 12 - 24V AC/DC | 1 x Interface module RS232/RS422/RS485 | Top hat rail housing |
| 58031 | 10/100BaseT autosensing | 100-250V~ 50-60Hz | 1 x Interface module RS232/RS422/RS485 | Desktop metal housing |
| 58034 | 10/100BaseT autosensing | 100-250V~ 50-60Hz | 4 x Interface modules RS232/RS422/RS485 | Desktop metal housing |
| 58231 | 10/100BaseT autosensing | 5V +/-5% | 1 x Interface module RS232/RS422/RS485 | Compact metal housing |
| 58431 | 10/100BaseT autosensing | 5V +/-5% | 1 x TTL (optional RS232, RS422/485) | none (OEM) |
| 58432 | 10/100BaseT autosensing | 5V +/-5% | 1 x TTL + RS485 2-wire (optional RS232, RS422) | none (OEM, credit card format) |
| 58331 | 10/100BaseT autosensing | 5V +/-5% | 1 x Interface module RS232/RS422/RS485 | 19" version |
| 58334 | 10/100BaseT autosensing | 5V +/-5% | 4 x Interface modules RS232/RS422/RS485 | 19" version |

**W&T**

## Content

**W&T**

**W&T**

**W&T**

# 1 Quickstart

# 2   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. You 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!

■ Using the ARP command

■ Through the serial port

■ Using the RARP protocol

■ Using DHCP/BOOTP protocol

## 2.1   Assigning the IP using the ARP command

⚠️ *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:

```
58xxx      [Model]
EN=00c03d004a05 ──── Ethernet address
OK xxxxxx
```
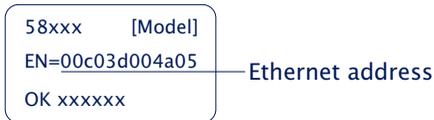
Insert a static entry into the ARP table of the computer using the following command line:

```
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]
```

ℹ️ *Older Windows systems only accept a static entry if there is a dynamic one already present. Here you should first ping another network station.*

⚠️ *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.*

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 will be established and the configuration menu is displayed in the Telnet window. All further settings are made here (see *Basic configuration of the Com-Server*).

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

## 2.2   Assigning the IP 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 A 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*).

## 2.2.1 Deactivating DHCP/BOOTP 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" (zero) 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 → SETP TCP/IP → BOOTP Client*.

## 2.3    Assigning the IP using a RARP server

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.

```
58xxx       [Model]
EN=00c03d004a05 ——— Ethernet address
OK xxxxxx
```
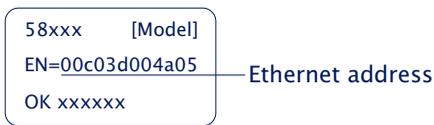
The Com-Server must be connected to the network in the same segment as the RARP server.

**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.

## 2.4   Assigning the IP using DHCP/BOOTP protocol

Many networks use DHCP (Dynamic Host Configuration Protocol) 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:

```
58xxx      [Model]
EN=00c03d004a05 ──── Ethernet address
OK xxxxxx
```

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* dhcpd.conf *into which a corresponding entry has to be made.*

### 2.4.1 Deactivating the 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" (zero), the BOOTP client of the Com Server is deactivated (see *Assigning the IP through the serial port*).

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.*

*i*

*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-Ethernet and Web-IO.

# 3   Form factors

■ Industry model

■ Office models

■ 19" models

■ OEM models

■ Compact model

## 3.1    Com-Server Highspeed Industry

Front view 58631



Error
Serial Port
Power  Status    Network

Bottom side 58631



Screw terminal

Supply
voltage
12-24V =/~
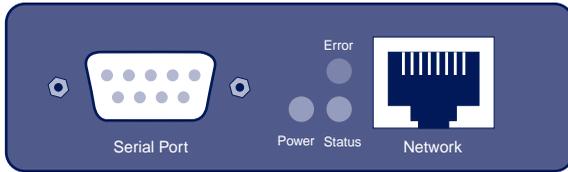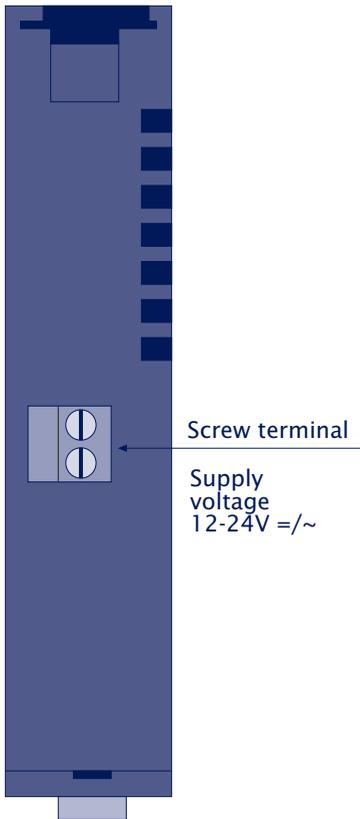
## 3.2 Com-Server Highspeed Office

### Front view 58031

Status  ● Power
Error
A Port

W&T
www.wut.de
Com-Server
Highspeed Office
**10/100 MBit**
#58031

Serial Port A

### Front view 58034

Status  ● Power
Error
D C B A Port

W&T
www.wut.de
Com-Server
Highspeed Office
**10/100 MBit**
#58034

Serial Port D    Serial Port C    Serial Port B    Serial Port A

### Back view 58031, 58034

Network

Reset

Supply voltage

## 3.3 Com-Server Highspeed 19"



Status LED

off = standby
on = error
blink = connection
flash = data

Error LED

off = OK
1x = check network
2x = check ser. data format
3x = check ser. handshake
on = self test / update

Port A
Port B
Port C
Port D
Power
Reset

10/100BaseT

Serial Port A

Serial Port B
Serial Port C
Serial Port D

W&T
www.wut.de
Com-Server
#5833x

58331 / 8TE

58334 / 12TE

## 3.4 Com-Server Highspeed OEM 58431



95,8mm

92,13mm

20,38mm

3,7mm

D=3,5mm

D=3,5mm

J1
J2

64,48mm

56,23mm

71,1mm

17,2mm

D=4mm

D=3,5mm

Optional serial module:
RS232/422/485
RS232
RS422/485

## 3.5 Com-Server Highspeed compact 58231

Under construction!

# 4    Connections, controls and displays

■ Pin assignments

■ Supply voltage

■ LED displays

## 4.1    Ethernet connection

All Com-Server models incorporate an IEEE 802.3-compatible network interface on a shielded RJ45 connector. The pin assignments correspond to an MDI port, so that the connection to the hub or switch is accomplished using a 1:1 shielded patch cable. The OEM model 58432 can also be optionally equipped with an LSA-Plus no-strip connect terminator module.

```
1 = Tx+                          8 = nc
  2 = Tx-                      7 = nc
    3 = Rx+                  6 = Rx-
        4 = nc   5 = nc
```
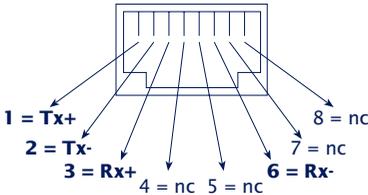
RJ45 female (AT&T256 standard)

### 10/100BaseT, autosensing
All models 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, correspponding 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.

### Galvanic isolation
The network connection is galvanically isolated from both the supply voltage and the serial port(s) at a minimum of 500V RMS.

### Link status
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    RS232/422/485 combi-module

Com-Server models 58631, 58031, 58034, 58331, 58334 and 58231 come standard with an RS232/422/485 multi-port whose modes are described below.

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.

### 4.2.1  Opening the Com-Server

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

The office versions can be opened by removing the four upper screws on the enclosure sides.

In the case of the compact version 58231, the bump guard on the housing side with the serial port must first be removed. This reveals four screws, which are removed in order to remove the circuit board from the housing.

*i*

*Equipping Com-Servers with interface modules other than those already installed is a modification which must be performed only by W&T.*

### 4.2.2 Mode selection

Switching between the modes is accomplished using the internal DIL switches located on the respective interface module. The following table shows an overview of all modes:

| Mode | SW1 | SW2 | SW3 | SW4 | SW5 | SW6 | SW7 | SW8 |
|------|-----|-----|-----|-----|-----|-----|-----|-----|
| RS232 (1) | OFF | OFF | OFF | OFF | OFF | OFF | OFF | ON |
| RS422, RS485 4-wire bus master | OFF | OFF | OFF | OFF | ON | x | x | OFF |
| RS485 4-wire automatic control | OFF | ON | OFF | OFF | ON | x | x | OFF |
| RS485 2-wire automatic control | ON | ON | OFF | OFF | ON | x | x | OFF |

(1) Factory default

### 4.2.3 RS232 mode (factory default)

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.

**DIP switch setting**

| Mode | SW1 | SW2 | SW3 | SW4 | SW5 | SW6 | SW7 | SW8 |
|------|-----|-----|-----|-----|-----|-----|-----|-----|
| RS232 | OFF | OFF | OFF | OFF | OFF | OFF | OFF | ON |

⚠️ *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.*

## Pin assignment and function RS232, DB9

| Pin | Direction | Signal | Description | Default Funktion (1) |
|-----|-----------|--------|-------------|----------------------|
| 1 | (IN) | NC/DCD | Data carrier detect | 1-port models: Not connected<br>4-port models: Ignored |
| 2 | IN | RxD | Receive Data | Data in |
| 3 | OUT | TxD | Transmit Data | Data out |
| 4 | OUT | DTR | Data Terminal Ready | 12V 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<br>+12V = ready to receive data<br>- 12V = not ready to receive data |
| 8 | IN | CTS | Clear To Send | Send data only at +3...12V |
| 9 | (IN) | NC/RI | Ring indicator | 1-port models: Not connected<br>4-port models: Ignored |

*(1) Applies only to the hardware handshake setting*

## Wiring examples

### Com-Server <> PC, 9-pin
W&T Art. No. 1199x
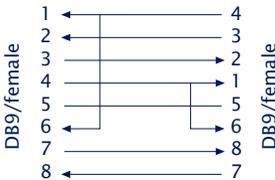


### Com-Server <> PC, 25-pin
W&T Art. No. 1179x
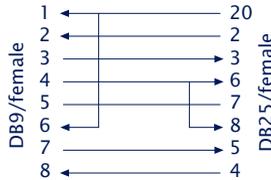


### Com-Server -> printer
W&T Art. No. 1189x
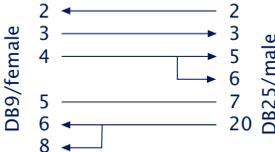


### Com-Server <> modem
W&T Art. No. 1198x
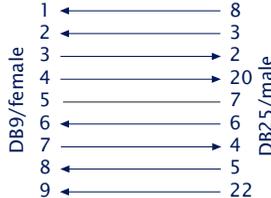
### 4.2.2 RS422/485 mode

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

⚠ *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.*

**DIP switch setting**

| Mode | SW1 | SW2 | SW3 | SW4 | SW5 | SW6 | SW7 | SW8 |
|------|-----|-----|-----|-----|-----|-----|-----|-----|
| RS422, RS485 4-wire bus master | OFF | OFF | OFF | OFF | ON | x | x | OFF |
| RS485 4-wire automatic control | OFF | ON | OFF | OFF | ON | x | x | OFF |
| RS485 2-wire automatic control | ON | ON | OFF | OFF | ON | x | x | OFF |

**Pin assignment and function RS422/485, DB9**

| Pin | Direction | Signal | Description | Default function (1) |
|-----|-----------|--------|-------------|----------------------|
| 1 | OUT | TxD A/- | Transmit Data A | Data output |
| 2 | IN | RxD A/- | Receive Data A | Data input |
| 3 | OUT | RTS A/- | Ready To Send A | Handshake output |
| 4 | IN | CTS A/- | Clear To Send A | Handshake input |
| 5 | --- | GND | Signal Ground | --- |
| 6 | OUT | TxD B/+ | Transmit Data B | Data output |
| 7 | IN | RxD B/+ | Receive Data B | Data input |
| 8 | OUT | RTS B/+ | Ready To Send B | Handshake output |
| 9 | IN | CTS B/+ | Clear To Send B | Handshake input |

*(1) with factory default setting*

## Modes

The DIL switches are used to set the following modes:

- **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.

## 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 ssytem 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 module:
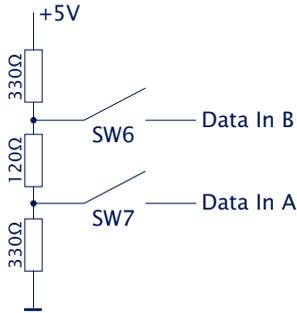


## Wiring examples

### RS422 connection with hardware handshake
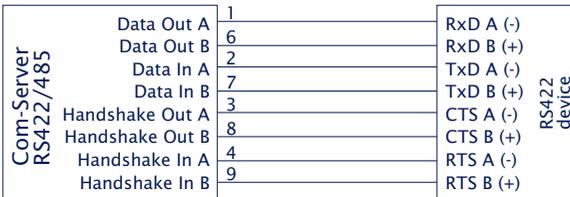


### RS485 connection (4-wire bus master)
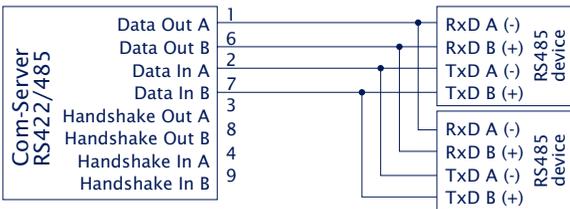


### RS485 connection 2-wire
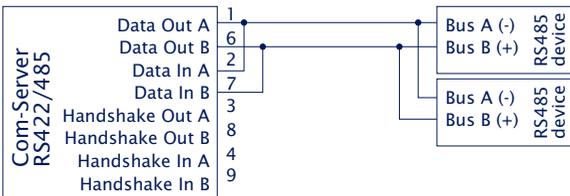
## 4.3 Interface for the OEM-Com-Server 58431

The OEM model 58431 has a UART interface that operates at TTL levels, which is implemented on a 2mm square post header. As an option, interface modules for RS232, RS422/485 or the multi-port can be connected.
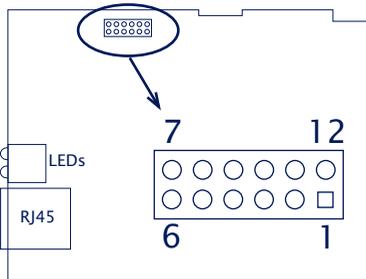


⚠️ *Wiring assignment for the OEM model 58432 „credit card format" differ from those described here. For this model please consult the belonging special section*

## Pin assignment and function TTL interface 58431

| 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 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.*

## 4.4 Interfaces for the OEM-Com-Server 58432

The OEM model 58432 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.

**PCB top side**

**PCB bottom side**
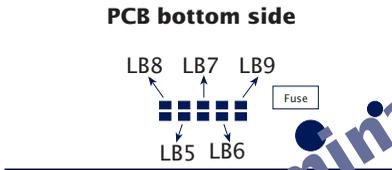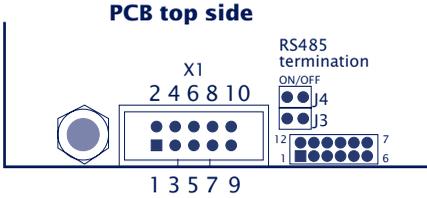
**Pin assignment X1**

Preliminary

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

*(1) The open collector feature of the reset pin 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
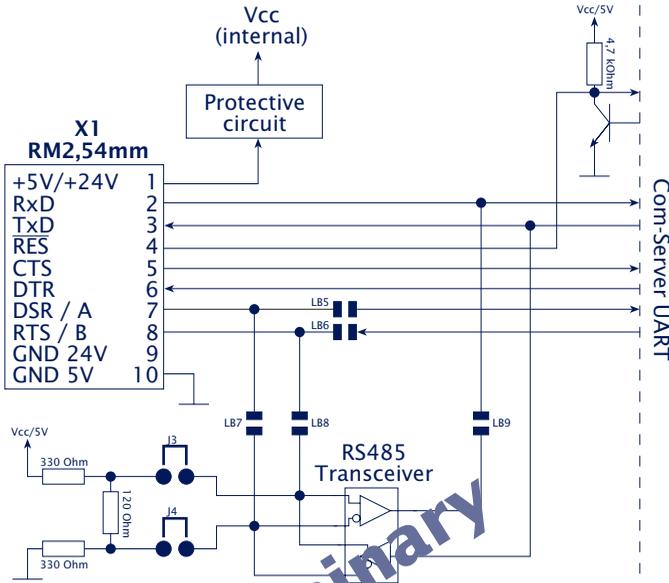
## 4.5 Supply voltage

### 4.5.1  Com-Server Highspeed Industry

The supply voltage for Model 58631 can be brought in on the adjacent screw terminals on the underside of the housing. DC voltage of any polarity or AC voltage may be used. The reverse polarity protection results in the following various maximum and minimum values for the supply voltage:

- AC: 9Veff (- 10%) - 24Veff (+10%)
- DC: 12V (-10%) - 40V (+10%)

If the supplied AC adapter for office applications is used, the connector pins can be connected to the screw terminals.

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

### 4.5.2  Com-Server Highspeed Office

The office models 58031 and 58034 include an integrated wide-range power supply for an AC voltage of 100-250V/50-60Hz. Mechanically the connection is made via a cold-device cable.

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

### 4.5.3  Com-Server Highspeed OEM and compact 58431, 58432, 58231

**58231, 58431**

The supply is brought in on a mains receptacle for hollow pin plugs located in the edge of the board. The feed voltage must be 5V ±5%.

Pin jack 5,5mm
Polarity:    inner = GND
            outer = +5V



**58432**

The supply voltage 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%.
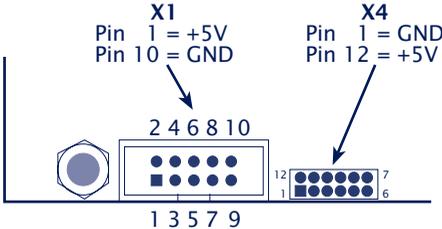


The current draw is indicated in the technical appendix.

### 4.5.4 Com-Server Highspeed 19" 58331, 58334

The Com-Server Highspeed 19" models get their supply voltage through the 96-pin VG connector. Use the following table to determine how jumpers a-c in the corresponding row depending on which bus system is used.

| Jumper | Pin VG strip A | B | C | Com-Server A | B | C | Standard (all three jumpers of row closed) |
|--------|------|-----|------|------|-----|------|-------------------------------------------|
| Row 1 | 9c | 32c | 31c | GND | +5V | +12V | VME-Bus J1 |
| Row 2 | 2b | 1b | n.c. | GND | +5V | +12V | VME-Bus J2 |
| Row 3 | 32c | 31c | 3c | GND | +5V | +12V | Multibus II |
| Row 4 | 32c | 1c | 13a | GND | +5V | +12V | ECB Bus |
| Row 5 | 32c | 29a | 9a | GND | +5V | +12V | BUS7ISA |

Component side of board



5V
5V
GND
GND
12V

Alternat supply voltage also possible through the 6-pin terminal strip, RM 3,96

D

C

B

Slots for serial ports

**Jumper area**

Jumper a
Jumper b
Jumper c

Row 1    Row 5

c
b
a

32                                            1

Example for VME-Bus J2:

Jumper a
Jumper b
Jumper c

Row 1    Row 5
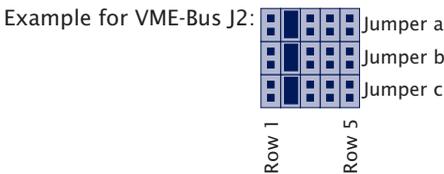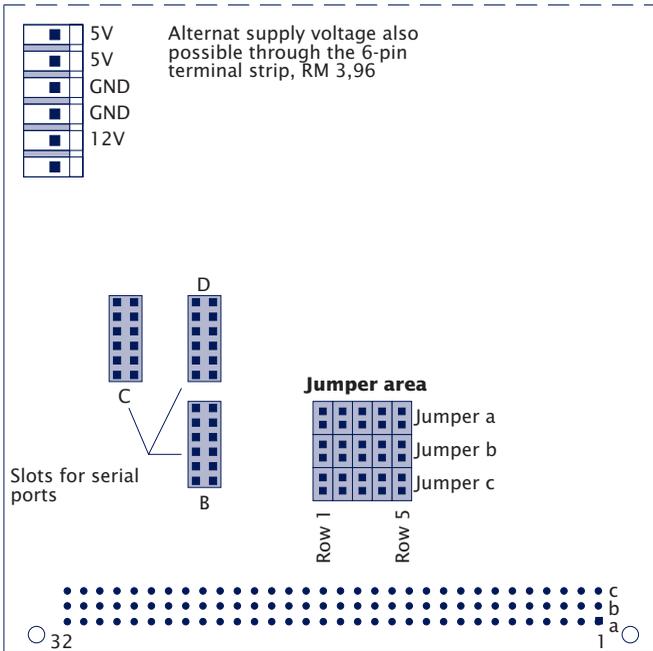
⚠️ *For standard operation of the Com-Server only the 5V supply is needed. The 12V voltage is simply brought through a fuse to Pin 11 on the module slots where it is used to supply future interface modules.*

## 4.6 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 occured) can also be read out using the Telnet configuration tool.

  **1 xflashing** = Check network connection
  The Com-Server cannot receive a link pulse from a hub. Check the cable or the hub port.
  **2 x flashing**= 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** = 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 sofstware 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

After you have finished the hardware installation and assigned the IP address, the remaining configuration is performed through the network in a Telnet session.

■ Telnet configuration under Windows

## 5.1    Telnet configuration under Windows

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. The telnet.exe program is always included with Windows 9x/NT/2000/XP, and is usually located in the Windows main directory.

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

Telnet [IP address] 1111

[adress]    = 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*.

Connect

Host Name: 172.16.232.186

Port: 1111

TermType: vt100

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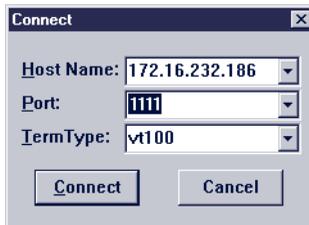
If the connection could be established, you will see the following menu in your Telnet window:

```
***************************
*  Com-Server Highspeed   *
***************************
1.  INFO  System
2.  SETUP System
3.  SETUP Port 0 (High-Speed Serial)
4.  SETUP Port 1 (High-Speed Serial)
5.  SETUP Port 2 (High-Speed Serial)
6.  SETUP Port 3 (High-Speed Serial)
7.  SAVE  Setup
```

*INFO System*, *SETUP System* and *SAVE Setup* as well as their sub-menus are the same for all Com-Server models and are described in the following section *Basic Configuration of the Com-Server*.

The menu items *SETUP Port x* are dependent of the number of serial ports on the respective Com-Server. For example, all single-port versions will only provide the *SETUP Port 0* menu. The description of these branches can be found in the section *Configuration of the Serial Ports*.

# 6  The basis configuration of the Com-Server

Here you find the explanation for all configuration possibilities which have to do with the Com-Server operating system and which are not directly related to the serial ports.

■ Navigation within the menutree

■ Menu: INFO System

■ Menu: SAVE Setup

■ Menu: SETUP System

## 6.1    Navigation within the menutree

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 *SAVE Setup*.

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

```
─INFO System ───── Cable Type              ┌──────────────────────┐
                 ├ MAC-Address            │   Always save using   │
                 ├ SOFTW Date / Rev       │    "SAVE Setup"       │
                 ├ HARDW Rev              │   to activate the new │
                 └ Run Time               │      settings !       │
                                          └──────────────────────┘
─SETUP System ──── Setup TCP/IP ─── IP-Address
                 │                ├ Subnet Mask
                 │                ├ Gateway ──────── Standard Gateway
                 │                ├ MTU (512-1024)  ├ Route 1 ─── Destination
                 │                └ BOOTP Client    ├ Route 2 ─┤ Netmask
                 ├ Telnet Password                  ├ Route 3 ─┘ Gateway
                 ├ System Password                  └ Route 4 ─
                 ├ Set Password
                 ├ System Name
                 ├ Flash Update ────── Net Update (TFTP)
                 │                └ Serial Update (Port 0)
                 ├ Factory Defaults
                 └ Reset

─SETUP Port 0    Setup for the serial port of the Com-Server.
─SETUP Port 1    This menu tree is where you configure the serial
─SETUP Port 2    port as well as the port-specific network functions.
─SETUP Port 3

└SAVE Setup      Transfer the selected parameters and save all
                 settings in non-volatile memory on the Com-Server
```

*i*

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

## 6.2 Menu: INFO System

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

↪ **Cable type**
  Shows the current Ethernet connection of the Com-Server (10BaseT or 100BaseT).

↪ **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.

## 6.3 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. 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.

## 6.4 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.

### 6.4.1  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).

↪ **Gateways**
In this menu branch the IP address of the standard gateway or router can be configured. If you have not configured fixed routes, the standard gateway is used for all network packets whose destination IP address is not in the local subnet.

↪ **Route 1-4 (Destination, Netmask, Gateway)**
In addition to the standard gateway, up to four fixed routes can be defined. Packets whose destination addresses are in the networks configured here (=destination) are always sent through the gateway assigned to this route. A fixed route is only accepted and stored by the Com-Server if the following check is true:

*Destination* AND *Netmask == Destination*

If the result is not equal to the destination, the entries are rejected.

⚠ ¡*Changes to the system parameters* IP-Address, Subnet Mask, Gateway *and* Route 1-4 *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 *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 *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.*

### 6.4.2 Menu: SETUP System → Telnet Password

Here you may specify an 8-character hexadecimal password (valid charcters = 0...9 and a...f) 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.*

### 6.4.3 Menu: SETUP System → System Password

The system password, which consists of any 31 (max.) characters, protects against any subsequent configuration and control access to the Com-Server.

- 1111: Telnet configuration menu
- 8003: Read the configuration file
- 8004: Write the configuration file
- 8888: Reset Com-Server
- 9084, 9184, 9284, 9384: Reset port status A-D
- 9094, 9194, 9294, 9394: Control port A-D
- 161 (UDP, SNMP)

On Port 1111 the system password is prompted as soon as the connection is opened. For all other associated TCP ports the password must be sent to the Com-Server no later than 2s after the TCP connection has been established.

Queries from SNMP managers are only responded to by the Com-Server if the community corresponds to the system password.

For additional information on using system passwords in conjunction with the configuration and control ports, see the section „Expanded Services of the Com-Server".

⚠️ *The system password is on a higher level than the Telnet password. This means that after assigning a system password, a previously set Telnet password may become invalid, so that the system password must be used for all password prompts.*

### 6.4.4 Menu: SETUP System → System Name

The freely configurable system name consisting of max. 31 characters is used to identify the Com-Server. This name is displayed as an opening message in the client for all Telnet sessions.

### 6.4.5 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!*

### 6.4.6 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.*

### 6.4.7 Menu: SETUP System → **Reset**

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

■ Services of the Com-Server

■ Addressing in the TCP/IP network

## 7.1    Services of the Com-Server

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 x → 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 |

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), RIP, DHCP/BOOTP, SNMP
- IP, TCP (Client / Server), UDP (Client / Server)
- FTP (Client / Server), Telnet (Client / Server)
- SLIP

## 7.2 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.
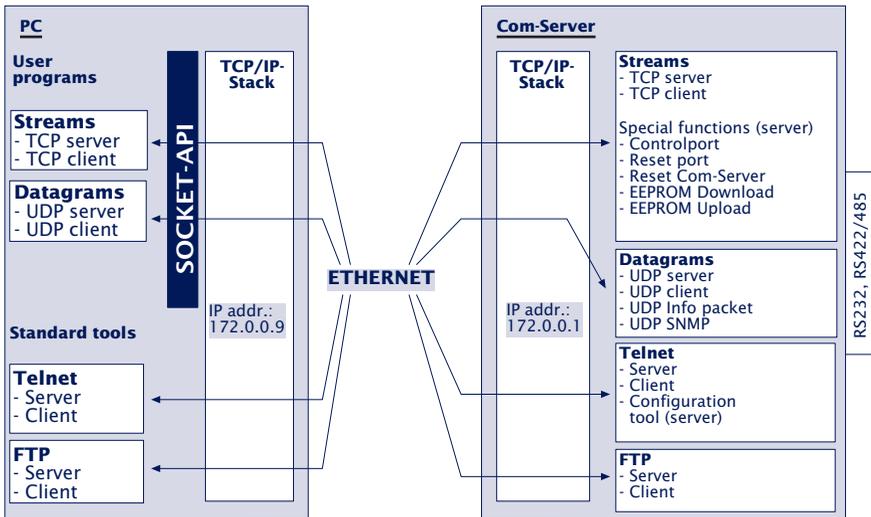


If in a particular case the factory assigned port number canot be used, you may modify it (see *Configuration of the TCP/IP modes*).

**W&T**

# 8    Configuration of the serial port

■ Overview of configuration menu

■ Configuring the serial transmission parameters

In addition to the usual settings for baud rate, data bits, parity and stop bits, the various handshake procedures and available special functions of the individual control signals are explained here.

## 8.1   The configuration menu

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*):

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

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.

```
─INFO System  ──── Cable Type
              ├ MAC-Address
              ├ SOFTW Date / Rev
              ├ HARDW Rev          ┌ IP-Address
              └ Run Time           ├ Subnet Mask
─SETUP System ──── Setup TCP/IP ── Gateway ───── Standard Gateway
              │                    ├ MTU (512-1024) ├ Route 1 ──┬ Destination
              │                    └ BOOTP Client    ├ Route 2 ─┼ Netmask
              ├ Telnet Password                      ├ Route 3 ─┴ Gateway
              ├ System Password                      └ Route 4 ─┘
              ├ Set Password
              ├ System Name
              ├ Flash Update ──┬ Net Update (TFTP)
              │                └ Serial Update (Port 0)
              ├ Factory Defaults
              └ Reset
                        The explanations for system configuration
                        can be found in the section *Base configuration
                        of the Com-Server*
```

SETUP Port 0 ┬ Port State ┬ Connection State
SETUP Port 1 ┤            ├ Error State
SETUP Port 2 ┤            └ Clear Port Mode
SETUP Port 3 ┤
(High-Speed  │
Serial )     │
             ├ UART Setup ┬ Baud ─────────┬ Standard Baudrates ┬ 230,4k ⋮
             │            │               └ Special Baud Divisor├ 172,8k ┬ 2400
             │            │                                     ├ 115,2k ├ 1200
             │            │                                     ├ 57600  ├ 600
             │            │                                     ├ 38400  ├ 300
             │            ├ Parity ───────┬ NONE                ├ 19200  ├ 150
             │            │               ├ EVEN                ├ 9600   ├ 110
             │            │               └ ODD                 ├ 7200   ├ 75
             │            │                                     ⋮ 4800   └ 50
             │            ├ Data Bits ────┬ 8
             │            │               └ 7
             │            ├ Stopbit ──────┬ 1
             │            │               └ 2
             │            ├ Handshake ────┬ None        ┬ Pin: RTS
             │            │               ├ Hardware    ├ Pin: DTR
             │            │               ├ Software    ├ Pin: CTS
             │            │               └ Special     ├ Pin: DSR
             │            │                             ├ XON/XOFF
             │            └ FIFO S/R: ────┬ FIFOs OFF   └ XON/XOFF (Filter)
             │                            └ FIFOs ON    disable
             │                                          8 / 8
             │                                          16 / 16
             │                                          32 / 56
             │                                          56 / 60
             └ TCP/IP Mode ┬ Local Port (TCP/UDP)
                           │
                           ├ TCP Client ──┬ Server Port        ┬ Inaktiv. Timeout
                           │              ├ Server IP           ├ Connect. Timeout
                           │              └ Special Options     ├ Disconnect Char
                           │                                    ├ Dispatch String 1
                           │                                    ├ Dispatch String 2
                           │                                    ├ Client: "C"+Addr
                           │                                    └ Response Mode
                           ├ UDP Client ──┬ Server Port
                           │              ├ Server IP           ┬ Client: "C"+Addr
                           │              └ Special Options     ├ Dispatch String1
                           │                                    ├ Dispatch String2
                           ├ Serial Socket ┬ Serial Protocol    ├ Serial Protocol
                           │  Interface    ├ Serial Coding      ├ Serial Coding
                           │               └ Protocol Char      ├ Write "C"+Addr
                           │                                    └ Disconnect Char
                           ├ Telnet Client ┬ Server Port (23)
                           │               ├ Server IP          ┬ Disconnect Char
                           │               └ Special Options    ├ Inaktiv. Timeout
                           │                                    └ Serial 0D->0D00
                           ├ FTP Client ──┬ Server Port (21)
                           │              ├ Server IP           ┬ Auto FTP
                           │              └ Special Options     ├ FTP Client Login
                           │                                    ├ Inaktiv. Timeout
                           ├ Box to Box ──┬ Slave Port          ├ Connect. Timeout
                           │  (TCP)       ├ Slave IP-Address     └ Protocol Char
                           │              └ Special Options     ┬ Dispatch String 1
                           │                                    └ Dispatch String 2
                           ├ IP Bus Mode ─┬ Slave: Master IP
                           │              └ Master: Subnet IP
                           ├ SLIP Router ─┬ Net Address
                           │              └ SLIP-Net Routing
                           └ System ──────┬ Network Delay
                              Options      ├ Flush Buffer
                                           └ Telnet Echo

Always save using
**SAVE Setup**
to activate the new
settings!

SAVE Setup       *Transfer the selected parameters and save all settings
                 in non-volatile memory on the Com-Server*

## 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 x → 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 and the UART-FIFOs.

⚠️ *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 Baud rate, Data bits, stop bits, parity

The desired transmission parameters can be selected by entering the corresponding designation in the respective branch of the menu tree. Invoking Save Setup in the master menu saves the changes you made in the non-volatile memory of the Com-Server and simultaneously activates them.

**Freely selectable baud rate (Special Baud Divisor)**
In the submenu Special Baud Divisor you can enter any desired divisor for generating the baud rate. This makes it possible to work even with non-standard baud rates. The baud rate and divisor are calculated using the following formulas:

Models 58631, 58431

$$\text{Divisor} = \frac{11{,}0592 * 10^6}{16 * \text{Baudrate}} \qquad \text{Baudrate} = \frac{11{,}0592 * 10^6}{16 * \text{Divisor}}$$

Models 58031, 58034, 58231, 58331, 58334

$$\text{Divisor} = \frac{7{,}3728 * 10^6}{16 * \text{Baudrate}} \qquad \text{Baudrate} = \frac{7{,}3728 * 10^6}{16 * \text{Divisor}}$$

The currently configured baud rate is displayed when selecting the submenu *SETUP Port x (serial) → UART Setup.*

By allowing any divisor, very high baud rates are possible. Please note in this connection that the upper limit for a bundled data stream is 230.4 kbps.

⚠️ *The indicated value refers to a handshake-controlled half-duplex transmission with no parity and a non-loaded network.*

### 8.2.2 The handshake modes

(SETUP Port x → 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 signal: 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 contorl signal:      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 contorl signal: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 ifnluenced. 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 und 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 Send*" 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 r serial)**
If this option is turned on and XON or XOFF characters occur in the actuqal 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 turnd 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 r 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], [Parity], [Data bits], [Stop bits], [Handshake], [FIFO]*

Possible values for the handshake variables are:

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

### 8.2.3 FIFO Send/Rec

The submenu allows the FIFO memory of the serial receive module to turn on and off. Models 58031 and 58034 also make it possible to set various trigger thresholds.

Especially when transmitting large quantities of data at high baud rates (above 57.6 Kbaud), it is recommended that you turn on the FIFOs in order to reduce the system load. If for example when working with serial protocols you require as short a delay in data transmission as possible, the FIFOs should be turned off.

# 9 Configuring the TCP/IP modes

■ Data transfer per TCP and UDP sockets

■ The standard protocols: FTP and Telnet as Client or Server

■ Tunneling of serial data streams – Box-to-box mode

■ The COM Port Redirector – Virtual COM ports under Windows

■ The Com-Server as SLIP-Router

## 9.1    The protocol stack of the Com-Server

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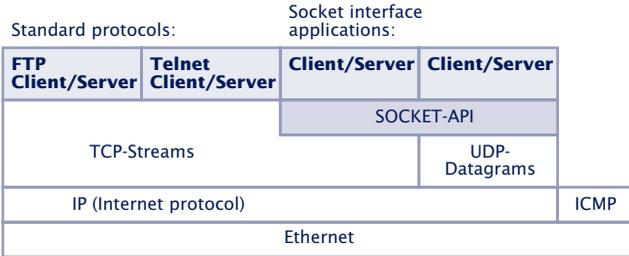
Each mode is explained individually in the following sections. The server mode does not have to be configured separately; it is activated when a client of a TCP/IP station directs a connection request to the server in the Com-Server.

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

## 9.2 Data transfer per TCP/IP and UDP/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 | TCP |
|---|---|---|
| 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 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, the following port numbers are pre-configured at the factory:

· Port A = 8000 (all models with only one serial port)
· Port B = 8100
· Port C = 8200
· Port D = 8300

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 metnod 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.*

### 9.2.2 TCP server mode

Default port numbers:
· Port A = 8000 (all models with only one serial port)
· Port B = 8100
· Port C = 8200
· Port D = 8300

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 prot 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 S*ETUP Port 0 → Port State → Connection State. *If a connection is active, the entry „*In Use Port number <IP address>*" appears there!*

### 9.2.3 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 portitself 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 x → TCP/IP Mode → Port Number.*

**The menu: SETUP Port x → 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: 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 → 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. InactivityTimeout = 0 deactivates the connection closing after a timeout.

*1 Tick: 1 second*
*Format: decimal*

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

*1 Tick: 1 second*
*Format: decimal*

↪ **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 (=deactivated).

*Format: decimal*

↪ **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.

```
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).

```
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** (**c**onnected)
   The connection was successfully opened. There is a TCP connection to the server.
**N** (**n**ot connected)
   The connection was not successfully opened. There is no response from the TCP server.
**D** (**d**isconnected)
   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** (**i**nvoked 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 x → 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  x → TCP/IP Mode → TCP Client to zero, or use the SETUP Port x → Port State → Clear Port Mode menu. The entry FREE must now appear in the SETUP Port x → Port State → Connection State menu.

**Application:**

**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 alternating connections between several Com-Server ports possible. In this case, addressing must be generated by the serial terminal devices.

**Example 1:** 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: deactive

**Com-Server B**
- Server Port: 0000
- Server IP: 0.0.0.0
- Client "C"+Addr: deactive



serial connection
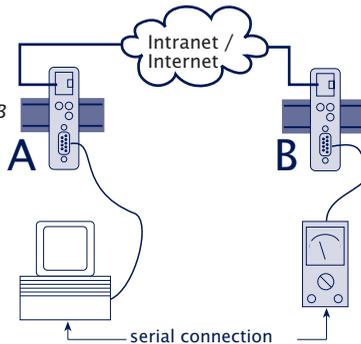
**Example 2:** A control program polls a measuring device; the measuring devicve can actively send data to the control program.

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

**Com-Server A**
- Server Port: 8000
- Server IP: *IP address Com-Server B*
- Client "C"+Addr: deactive

**Com-Server B**
- Server Port: 8000
- Server IP: *IP address Com-Server A*
- Client "C"+Addr: deactive



serial connection
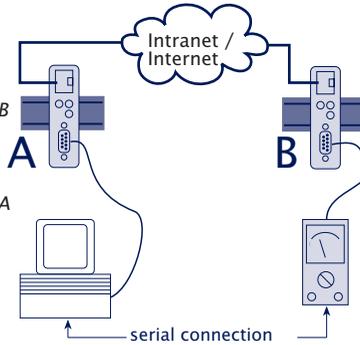
**Example 3:** A control program polls multiple measuring devices; the measuring devices can (e.g. in case of error) independently send data to the control program.

**Configuration menu:**
→ **TCP/IP Mod** → **TCP-Client**

**Com-Server A**
- Server Port: 8000
- Server IP: *IP address Com-Server B*
- Client "C"+Addr: active

**Com-Server B, C, ...**
- Server Port: 0000
- Server IP: 0.0.0.0
- Client "C"+Addr: deactive
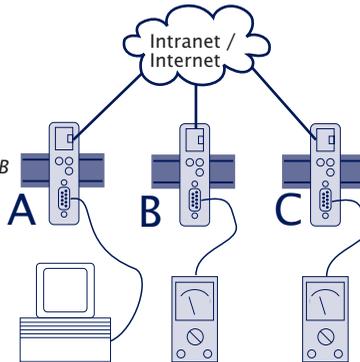


The control program addresses the measuring devices through the serial port in the following format: *C[IP address],[port_number]<CR>*

Example:                          *C172.10.230.10,8000*

**9.2.4 UDP client mode (Menu: UDP Client)**

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).

This wireless functionality of UDP must be used to configure the Com-Server port in any case (as a Client or as a Server) for exchanging data via UDP datagrams. There are two ways to specify the network destination for received serial data. First, directly through the menu items ...UDP Client -> Server Port and ...Server IP. Alternately you can activate the option Client:->C->+Addr., whereby the destination parameters IP address and port number are expected in serial format before the actual user data.

If the Com-Server is configured as a UDP client, all UDP datagrams addressed to the configured port number are accepted.

**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 port specified in the menu *Setup Port 0 → TCP/IP Mode → Local Port.* The following port numbers are pre-configured at the factory:

· Port A       = 8000 (all models with only one serial port)
· Port B       = 8100
· Port C       = 8200
· Port D       = 8300

**The menu: SETUP Port x → 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.

```
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).

```
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.

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

**Deactivating UDP client mode**
In the menu *SETUP Port x → TCP/IP Mode → UDP Client* set
the entry for *Server Port* to zero, or use the menu *SETUP Port*
*x →Port State → Clear Port Mode*. In *SETUP Port x → Port*
*State → Connection State* the entry *FREE* must be visible.

## 9.3 Serial Socket Interface

The Serial Socket Interface enables defined communication between Com-Servers and the device connnected to their serial port. Comparable to the TCP/IP socket interface in Windows of 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 netework 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 *(http://www.wut.de).*

## 9.4 Data transfer per Telnet

Telnet is a terminal emulation protocol. Data transmission is character-oriented. Telnet llows 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

Default port numbers:
· Port A     = 23 or 6000
· Port B     = 6100
· Port C     = 6200
· Port D     = 6300

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

Telnet invocation:

```
telnet [IP address]  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 x → TCP/IP Mode → System Options*.

### 9.4.2 Telnet client mode

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.

**The Menu: SETUP Port x → 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. InactivityTimeout = 0 deactivates the connection closing after a timeout.

    *1 Tick: 1 second*
    *Format: decimal*

↪ **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. Disconnect Char = 0 deactivates this mode

*Format: decimal*

↪ **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 x → 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 x → TCP/IP Mode → Telnet Client* to zero, or use the *SETUP Port x → Port State → Clear Port Mode* menu. The entry *FREE* must now appear in the *SETUP Port x → 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

Default port numbers:
- Port A     = 21 or 7000
- Port B     = 7100
- Port C     = 7200
- Port D     = 7300

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 address] 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:

↪ PUT *[local file] [remote file]*
  Sends the file *local file* to the RS232. Enter any desired character for [*remote file]*, since no remote file exists.

↪ GET *[remote file] [local file]*
  Gets characters from the RS232 in local file. 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.

↪ ASCII
Send in ASCII mode

↪ IMAGE
Send in binary mode

↪ QUIT
Ends the FTP session

⚠ *The entry* FREE *must be located in the S*ETUP Port 0 Þ Port State → Connection State *menu before a connection can be opened*

### 9.5.2 FTP client mode

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.

**The Menu: SETUP Port x → 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 *FTP-Client* now appears in the *SETUP Port x → 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. Inactivity timeout = 0 means *infinite.* In this case the parameter *Protocol Char* m u s t be configured.

   *1 Tick: 1 second*
   *Format: decimal*

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

*1 Tick: 1 second*
*Format: decimal*

↪ **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. Disconnect Char = 0 deactivates this mode. In this case the parameter „*Inactivity Timeout*" must be configured.

*Format: decimal*

|  |  | **Open connection** | **Close connection** |
|---|---|---|---|
| LIST<br>RETR | Directory<br>File | 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 |
| APPE<br>STOR | File<br>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 *Inactivity Timeout* or when the *Protocol Char* was received. |

**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" to activate the serial protocol of the FTP-Client

↪ **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 *File* |
| APPE | File | Saves all serially received data to the file end of *File* |
| RETR | File | Outputs *File* to the serial port |
| DELE | File | Deletes *File* |
| 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 capitalization of the commands. A maximum of 128 characters are available for the complete login string!*

In serial generation of the login string, the individual commands are separated from each other by a linefeed (0x0a). When entering via keyboard, the character can be generated by pressing CTRL J. This results in the 3 following formats for the login string:

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

login<lf>
password<lf>
DELE dir/remote file<lf>
<Protocol Char>

RESET<lf>
<Protocol Char>
```

**Example 1:** 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.

## Example 2:

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:

| Com-Server | Serial device |
|---|---|
| `egon<lf>happy<lf>TYPE A<lf>RETR /etc/hosts<lf><Ctrl C>` | ← |
| `OK<Ctrl C>` → | |
| `[content of file]` → | |
| `221 Goodbye<Ctrl C>` → | |
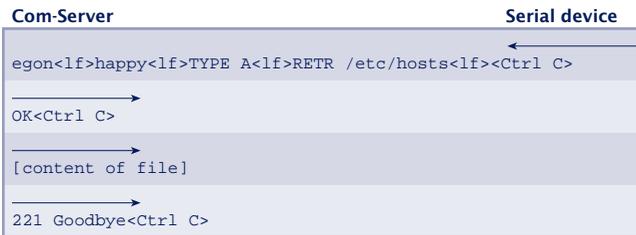
## Example 3:

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:

| Com-Server | Serial device |
|---|---|
| `egon<lf>happy<lf>TYPE I<lf>APPE /usr/egon/config<lf><Ctrl C>` | ← |
| `OK<Ctrl C>` → | |
| `[binary Data]` | ← |
| `221 Goodbye<Ctrl C>` → | |
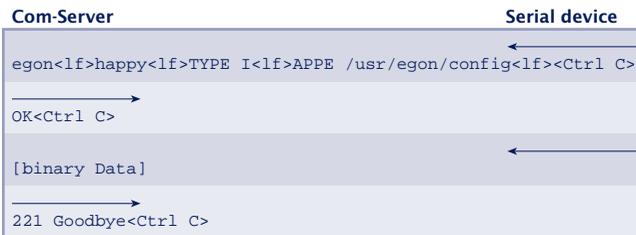
**Example 4:**

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:

**Com-Server**                                                    **Serial device**

```
egon<lf>happy<lf>TYPE A<lf>RETR /usr/privat<lf><Ctrl C>

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 x → TCP/IP Mode → FTP Client* to zero, or use the menu *SETUP Port x → Port State → Clear Port Mode.* You must see the entry *FREE* on the *SETUP Port x → Port State → Connection State* menu.

## 9.6 Box-to-Box mode

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).

On the network side, the Com-Servers in a box-to-box connection only exchange data if there are also serial user data present. There is no acknowledgement transmission beyond the TCP protocol

**The Menu: SETUP Port x → 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 (see examples at the end of this section).*

↪ **Slave Port** (set only for master port)
   Port of the serial Com-Server Slave port. The following port numbers are pre-configured at the factory:

   • Port A = 8000 (all models with only one serial port)
   • Port B = 8100
   • Port C = 8200
   • Port D = 8300

   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 S*ETUP Port x → 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.

**Sampleconfiguration for „Box-to-Box" mode**

**Box-to-Box Master**
IP address: 172.16.231.8
Port number of Port A: 8000

SETUP Port 0
↳ TCP/IP Mode
   ↳ Box to Box
      → Slave Port
        **8000**
      → Slave IP-Address
        **172.016.231.005**
      ↳ Special Options
        → Dispatch Str.1
          **0D0A**
        ↳ Dispatch Str.2
          **0000**

**Box-to-Box Slave**
IP address: 172.16.231.5
Port number of Port A: 8000

SETUP Port 0
↳ TCP/IP Mode
   ↳ Box to Box
      → Slave Port
        **0000**
      → Slave IP-Address
        **000.000.000.000**
      ↳ Special Options
        → Dispatch Str.1
          **3132**
        ↳ Dispatch Str.2
          **0D0A**



Intranet / Internet

A          B
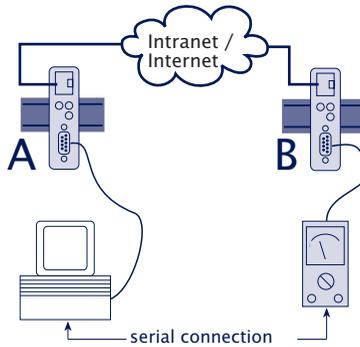
serial connection

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 commection 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 x → TCP/IP Mode → Box to Box* menu to zero („-"), or use the menu *SETUP Port x → Port State → Clear Port* Mode.

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

## 9.7 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.

ETHERNET, Subnet Mask: 255.255.255.0



| **Master** | **Slave** | **Slave** | **Slave** |
|---|---|---|---|
| | Measuring device | Measuring device | |

**IP address:**

| Com-Server A: | Com-Server B | Com-Server C | Com-Server D |
|---|---|---|---|
| 172.16.231.1 | 172.16.231.36 | 172.16.231.89 | 172.16.231.14 |

**Menü: SETUP Port 0 → TCP/IP Mode → IP Bus Mode**

| → Master: Subnet IP | → Slave: Master IP | → Slave: Master IP | → Slave: Master IP |
|---|---|---|---|
| 172.16.231.0 | 172.16.231.1 | 172.16.231.1 | 172.16.231.1 |
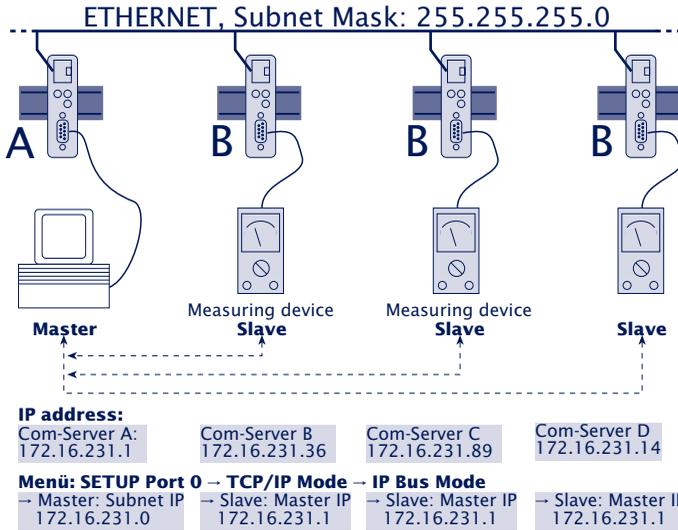
The Master sends to all Slaves and receives from all Slaves. Slaves are not able to exchange data between each other however.

⚠ *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.*

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

⚠️ *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 serttings you must first close the Telnet connection using „q" and then reopen it.*

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

## 9.8 The Windows COM port redirector

With the COM Port Redirector available for Windows 9x, NT, 2000 and XP, the virtual COM ports are installed in the respective system. These behave in a serial application exactly like normal local ports, but are actually located on Com-Servers located in the network.

Use of the COM Port Redirector in connection with Com-Servers is free and not subject to any licensing requirements.

**Where to I obtain a COM Port Redirector?**
The latest version of the COM Port Redirector as well as additional tools, application descriptions and FAQs can always be downloaded from our Web site.

http://www.wut.de

The simplest way to navigate from there is by using the menu tree on the left side of the page. Follow the *Products & Downloads → Com-Servers* path to get to the menu branch containing a direct link to the COM Port Redirector.

**Com-Server settings**
From the point of view of the Com-Server only the 3 network-specific parameters IP Address, Subnet Mask and Gateway Address need to be configured for operating with the COM Port Redirector. It is not necessary to set the serial transmission parameters such as baud rate. These are determined by the serial application and sent by the COM Port Redirector to the Com-Server over the network.

**Installing and configuring the COM Port Redirector**
Installation and configuration of the COM Port Redirector is described in the respective online help.

## 9.9 SLIP-Router mode

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.

**The Menu: SETUP Port x → 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 serttings 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!

**Packet frame:** | Ethernet | IP datagram | ¦ | SLIP | IP datagram |
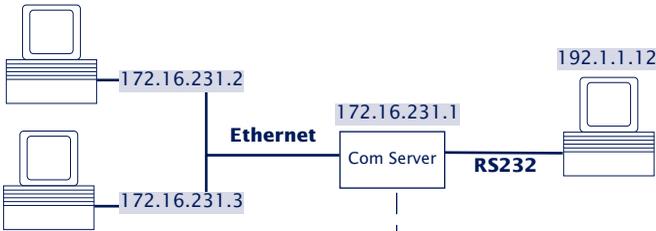
..→ **SLIP Router → Net Address = 192.1.1.0 / → SLIP-Net Routing = 1**

ping 172.16.231.1

ping 192.1.1.12

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

ping 172.16.231.1

Destination IP:         = 172.16.231.1          = 192.1.1.12

Source IP:              = 172.16.231.1          = 192.1.1.12
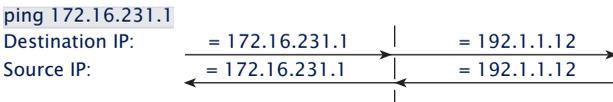
ping 192.1.1.2

This IP address is not routed!

**Deactivating SLIP router mode**

Set the parameter *Net Address* to zero (0.0.0.0) in the *SET-UP Port x → TCP/IP Mode → SLIP Router,* or use the menu *SET-UP Port x → 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 der Konfigurationsdaten**
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 | TYPEDEF | BYTE | TYPEDEF | |

**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 | unsigned int (16 bit) | RD+WR |
| 5 | ... → SLIP Router → Net-Address | long (32 bit) | RD+WR |
| 6 | ... → 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 sent in net order, with the high byte first and the low byte last.

**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 |
|------|-----|----------|------------|----------|-------------|---|
| TYPE | LEN | PARAM_NO | IP address | PARAM_NO | Subnet Mask | |
| 1 | 10 | 1 | hex: AC 10 EF 1 (172.16.231.1) | 2 | hex: FF FF FF FF (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 |
|------|-----|----------|----------|---|
| **TYPE** | **LEN** | **PARAM_NO** | **PARAM_NO** | |
| 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 | 7 | 8 | 12 |
|------|-----|----------|-----|----------|-------------|---|
| TYPE | LEN | PARAM_NO | MTU | PARAM_NO | MAC-Adresse | |
| 3 | 10 | 4 | hex: 02 00 (512) | 16 | hex: 00 C0 3D 00 30 DB (00-C0-3D-00-30-DB) | |

## 9.10 Menu ... → TCP/IP Mode → System Options

In this menu branch you can set specific system parameters.

↪ **Network Delay [10ms-Ticks]**
   **Default: 0000**

This value specifies the minimum delay time of the Com-Server after serial data have arrived. After this time has expired, it packs the data into a network packet and sends them to the respective communications partner. The default setting 0 means the attempt is made to send the data as quickly as possible. The resulting high time transparency does however bring with it the drawback of a higher number of network packets.

If the serial transmission takes place in predictable block sizes, this value can be adjusted to optimize the network load. There is also the advantage that the serial blocks are transmitted within a network packets, so that they arrive at the receiver bundled.

**Example:**
You are using serial data blocks of 25 bytes each and transmission parameters of 9600 baud, 8 data bits, no parity and 1 stop bit. Each block thus has a length of approx. 26ms (1/9600 * 10 bits * 25 bytes). If the network delay is set here to a value of 3 (=30ms), each block is sent to the receiver with a network packet.

⚠️ *The situation described here applies only to a non-overloaded and error-free network. If there are interruptions in the data flow on the network side, any serially accumulated data are also sent in larger network packets.*

↪ **Telnet Echo**
**Default: active**

When opening a Telnet session, negotiation between the stations takes place on the network side to determine who will generate the echo of the characters sent by the client. Either the client application generates a local echo, or the Telnet server generates a remote echo by immediately returning all received data. The Telnet echo option on the Com-Server is defined as follows:

**Telnet Echo = active**
The Com-Server negotiates a remote echo with the client application, and the local echo is turned off. In this case the device serially connected to the Com-Server must generate the echo.

**Telnet Echo = disabled**
The Com-Server tells the client application when opening the connection that it will not generate an echo, which results there in activation of the local echo. The local echo may have to be manually activated on the Telnet client.

↪ **Flush Buffer**
**Default: active**

If this switch is "active", the serial buffer is cleared each time a new connection is opened; any residual data which cannot be transmitted (e.g. handshake stop of the serial terminal device) are thus lost. But if you connect a terminal device to the Com-Server which for example requires a longer handshake stop for program reasons, you can prevent clearing of the buffer by deactivating the switch. In this way multiple connections can be opened one after the other on the network side and the data are accumulated in the buffer until they are sent.

**W&T**

# 10  Status and error messages

■ Status messages of the menu *Port State*

■ Resetting a fixed mode: *Clear Port Mode*

**10.1 Menu Setup Port x → 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 occured. In addition, Clear Port Mode can be used to clear a fixed set mode.

↪ **Connection State**

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

| | |
|---|---|
| Connection State<br><br>FREE | The port is in standard mode and has no connection |

| | |
|---|---|
| Connection State<br><br>In Use: Port 2000 (172.016.231.001) | The port is in server mode and has a connection to the process having the port no. 2000 on the TCP/IP station having IP address 172.16.231.1<br>→ **Client address** |

| | |
|---|---|
| Connection State<br><br>TCP Client<br>Locked: Port 2000 (172.016.231.001) | The port is configured as a TCP and has connection to the server process having port no. 2000 on IP station having IP address 172.16.231.1<br>→ **Configured TCP/IP mode**<br>→ **Servers 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 ).

Disconnect: The Com-Server wants to open a connection but it is **refused** by the server

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

↪ **Error State**

This list shows the errors which occured on the port. If more than five errors have occured 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 ius the time when the error occured in hours and minutes since the last time the Com-Server was reset. You can clear the eror 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 con-figured 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 non-volatile memory.*

# 11 Expanded services of the Com-Server

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.

■ The control port and its data structure

■ Status requests and configuration of the serial port

■ Resetting the Com-Server

■ Uploading/downloading of the configuration data

■ Inventory taking per UDP

■ SNMP management

## 11.1 The control port

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.

Control port numbers:

· Port A    = 9094
· Port B    = 9194
· Port C    = 9294
· Port D    = 9394

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

The data- andcontrol ports are independent of each other. They may be individually opened and closed whenever desired.

**Use of the system password**
If a system password has been configured (see section „Basic Configuration of the Com-Server"), this must be null-terminated (=[password] + 0x00) and sent to the Com-Server via the Control Port within 2s after a connection has been successfully opened. If the Com-Server does not receive a correct or even any password within this time, it sends the message *PASSWD?* followed by a null byte (0x00) to the client and closes the TCP connection.

If no system password is configured you can, as described below, begin with the exchange of info structures as soon as the TCP connection has been established.

### 11.1.1 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 contro port, the reply is still an info structure. The declaration of the structures is made in C language.

*Note for all structures:*

A ***word*** corresponds to a 16-bit integer

A ***char*** corresponds to a byte (8bit)

Hex convention: ***0x*** preceding the value

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.

## Structure 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
                     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;
```

**Structure 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 number of bytes which are in the send and receive buffers 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 break_mode :1; //1 = set_break was set-         - HighByte
                                    //0 = clear_break was set       |
                word dummy       :1; //not used                     |
                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*

**Sending a break signal**

The *set_break/clear_break* flags can be used to activate/deactivate break mode and send a break signal to the serially connected device. Activated break mode is signaled by *break_mode*=1. When creating the break signal, a distinction between the various Com-Server models needs to be made as follows:

↪ **58631, 58431**

With these models, break mode is first set using set_break. Then with each following serial character to the *data connection* the Com-Server sends a baud rate-dependent BREAK signal (approx. 10 bit times). Setting *clear_break* deactivates break mode and reestablishes normal transmission mode.

↪ **58031, 58034, 58231**

After setting *set_break*, break mode is activated and a baud rate-dependent break signal is generated. The level of the data output remains inverted until break mode is deactivated again by setting *clear_break*.

**Structure 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
        {
          struct_baud_fifo
          {
          char baud       :5;  //Baud rate for channel
                               // 11 = 230400  6 = 2400
                               // 14 = 153600  7 = 1200
                               // 15 = 115200  8 = 600
                               // 0  = 57600    9 = 300
                               // 1  = 38400   10 = 150
                               // 2  = 19200   16 = 110
                               // 3  = 9600    12 = 75
                               // 5  = 4800    13 = 50
                               // Baudrates are coded within the first 5 bits
          char fifo_aktiv :1;  // 0 = FIFO disabled, 1 = FIFO enabled
          char fifo       :2;  // Send/Receive FIFO trigger
                               // 0 = 8 / 8
                               // 1 = 16 / 16
                               // 2 = 32 / 56
                               // 3 = 56 / 60
                               // models 58631 and 58431 have a fix FIFO
                               // trigger so they ignore these bits
          };

          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 must 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).

## 11.2 Reset Com-Server-Port

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.*

TCP port numbers:

- Port A    = 9084
- Port B    = 9184
- Port C    = 9284
- Port D    = 9384

**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 station172.16.231.100 and deletes its parameters.

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

**Use of the system password**
If a system password has been configured (see section „Basic Configuration of the Com-Server"), this must be null-terminated (=[password] + 0x00) and sent to the Com-Server via the Control Port within 2s after a connection has been successfully opened. If the Com-Server does not receive a correct or even any password within this time, it sends the

message PASSWD? followed by a null byte (0x00) to the client and closes the TCP connection.

If no system password is configured, then as the example shows the Com-Server immediately closes the TCP connection as soon as it has been established and performs a port reset.

*⚠️ 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 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.*

## 11.3 Software reset of the Com-Server

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.

### Use of the system password

If a system password has been configured (see section „Basic Configuration of the Com-Server"), this must be null-terminated (=[password] + 0x00) and sent to the Com-Server via the Control Port within 2s after a connection has been successfully opened. If the Com-Server does not receive a correct or even any password within this time, it sends the message PASSWD? followed by a null byte (0x00) to the client and closes the TCP connection.

If no system password is configured, then as the example shows the Com-Server immediately closes the TCP connection as soon as it has been established and performs a port 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!*

## 11.4 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 non-volatile memory and send it to another Com-Server. Especially when installing a large number of Com-Servers having the same settings this methjod eliminates having to open a Telnet configuration connection for each individual unit.

### Use of the system password

If a system password has been configured (see section „Basic Configuration of the Com-Server"), this must be null-terminated (=[password] + 0x00) and sent to the Com-Server via the Control Port within 2s after a connection has been successfully opened. If the Com-Server does not receive a correct or even any password within this time, it sends the message PASSWD? followed by a null byte (0x00) to the client and closes the TCP connection.

If no system password is configured, the Com-Server expects or sends the configuration data directly after the TCP connection has been established.

### 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 1024-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 1024 bytes for a new configuration. After the client closes the TCP connection, the Com-Server saves the data in its non-volatile memory 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 non-volatile memory 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 non-volatile memory 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.*

**11.4.1 Uploading/downloading 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*

## 11.5 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 8513. 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. 8513 is used.

⚠️ *For reasons of downward compatibility, the Com-Ser vers continue to support the previous UDP infoport 8512. In new applications however you should use only Port 8513 as described here.*

### 11.5.1 The infopacket

Each infopacket consists of 3-6 data structures. *BOX_VERSION* contains information on the respective model of the Com-Server and its firmware status. The structure *BOX_DESCRIPT* provides the network-relevant data such as MAC- and IP-address. The 3.-6. structure *PORT_DESCRIPT* provides information about the configured mode and the current connection status of the individual ports. The total length of the InfoPacket is calculated as follows: 1*0 + 22 + (port_no x 10) bytes.*

```
#pragma pack(1)

typedef struct _BOX_VERSION           // ( 10 byte )
   {     unsigned int version;        // 0x10: 1.0 (Version of this structure)
         unsigned int sw_rev;         // z.B. 1.24 (0x1501)
         unsigned int hw_rev;         // C2_EURO         = 2.0 (0x0002)
                                      // C3_PC104        = 3.0 (0x0003)
                                      // C4_MINI         = 4.0 (0x0004)
                                      // C5_100BT        = 5.0 (0x0005)
                                      // C6_INDUSTRY     = 6.0 (0x0006)
                                      // C8_LOW_VOLTAGE  = 8.0 (0x0008)
         unsigned int reserved[2];
   }BOX_VERSION;

typedef struct _BOX_DESCRIPT          // ( 22 byte)
   {     unsigned char mac_addr[6];   // Com-Server MAC address
         unsigned long IP_number;     // Com-Server IP address
         unsigned long gateway;       // Gateway
         unsigned long subnet_mask;   // Subnet Mask
         unsigned int  MTU;           // MTU
         unsigned int  port_anz;      // Number of Com-Server ports
   }BOX_DESCRIPT;                      // (port_anz * 10 byte)

typedef struct _PORT_DESCRIPT         // ( 10 byte )
   { union
         { unsigned int wState;
           struct _new_type
             { unsigned char port_type; // 0x01 = serial port (serial CPU port)
                                        // 0x02 = serial port (UART port)
                                        // 0x03 = reserved
                                        // 0x04 = Digital E/A
                                        // 0x05 = Analog E/A (under construction)
                                        // 0x06 = W&T Dual Port RAM
              unsigned char state;      // 0=free, 1=connect, 2=waits
              };
            };
         unsigned int  mode;         // 0x0001 = TCP client
                                     // 0x0002 = TELNET client
                                     // 0x0003 = FTP client
                                     // 0x0004 = Box2Box client (active)
                                     // 0x0005 = UDP Send/ReceiveMode
                                     // 0x0006 = MULTI PORT (DPRAM, SERIAL PROTOCOL)
                                     // 0x0007 = SNMP agent
                                     // 0x0008 = Box2Box server (passive)
                                     // 0x0010 = SLIP router
                                     // 0x0020 = PPP router (under construction)
                                     // 0x0030 = Box2Box bus system: Slave Box
                                     // 0x0040 = Box2Box bus system: Master Box
         unsigned long remote_IP;   // if state == CONNECT, else 0
         unsigned int  remote_port; // if state == CONNECT, else 0
   }PORT_DESCRIPT;
```

```
typedef struct _WT_INTERN3        // ** all parameters in Hostorder / Low Byte First **
  {       BOX_VERSION    bv;      // Port = UDP_BOX_INFO_8513
          BOX_DESCRIPT  bd;
          PORT_DESCRIPT pd[ACT_PROCESS];
  } WT_INTERN3;
```

```
#pragma pack()
```

⚠️ *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 broadcase if necessary.*

## 11.6 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.

**Where do I obtain the MIB file?**
The latest version of the private MIB as well as other tools, application descriptions are always available at

http://www.wut.de

The simplest way to navigate from there is by using the menu tree on the left side of the page. Follow the *Products & Downloads → Com-Server → Tools* path to get to the menu branch containing a direct download link for the private MIB.

⚠ *If a system password was configured in the Com-Ser ver, queries from SNMP managers are only answered if the community contained there corresponds to the system password.*

# 12 Firmware update of the Com-Server

Since the operating software is under constant develpoment, this device also allows you to perform a firmware update. The available upload variations are described here.

■ Where do I get the current firmware?

■ Network firmware update unter Windows

■ Network firmware update under UNIX

## 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*

The simplest way to navigate from there is by using the menu tree on the left side of the page. Follow the

*Products & Downloads → Com-Server → Group → Model number → Firmware*

path to get to the menu branch containing a direct download link for the firmware.

If you do not know the model number, you can find it on the sticker located on the narrow side of the enclosure, likewise the Ethernet address.

Type no.            Description, e.g. TB-number
                    for special versions

58xxx   [description]
EN=00c03d004a05
OK xxxxxx
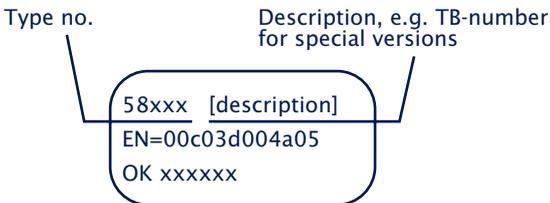
⚠ *In particular if the sticker indicates an TB number as the part number, it is possible that the Com-Server has a special, customer-specific firmware or configuration. This would be overwritten by uploading the standard firmware. Please contact the responsible administrator in such cases before the update.*

## 12.2 Network firmware update under Windows

Required is a PC running Windows 9x/NT/2000/XP 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.

It is not necessary to do any special preparation in the Com-Server for a firmware update. You only need to close all data, control and configuration connections.

The update tool is for the most part self-explanatory. Should you have any questions, however, please refer to the associated documentation or use Online Help.

⚠ *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.*

## 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 6 files that are send 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 six 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.
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.

**Example: SCO UNIX**

Enter the following commands after the corresponding prompt:

```
# tftp
tftp> connect [ip_number|host_name]
tftp> binary
tftp> put Cxr1_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
#
```

⚠️ *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.*

# Appendix

■ TCP/IP activation under Windows

■ Overview of the port numbers

■ Example: Serial assignment of the IP address under Windows

■ Web applications HTTP,SMTP, POP3 ...

■ Technical Data

## 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 ruters, 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.

## 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.

## Overview of the port numbers

The port numbers factory set in tzhe Com-Server:

| Port number | Application |
|---|---|
| 6000, 23 (TCP) | Telnet server port A |
| 6100 (TCP) | Telnet server port B |
| 6200 (TCP) | Telnet server port C |
| 6300 (TCP) | Telnet server port D |
| 7000, 21 (TCP) | FTP server port A |
| 7100 (TCP) | FTP server port B |
| 7200 (TCP) | FTP server port C |
| 7300 (TCP) | FTP server port D |
| 8000 (TCP) | Socket server port A (can be configured in the setup) |
| 8100 (TCP) | Socket server port B (can be configured in the setup) |
| 8200 (TCP) | Socket server port C (can be configured in the setup) |
| 8300 (TCP) | Socket server port D (can be configured in the setup) |
| 1111 (TCP) | TELNET configuration port |
| 9094 (TCP) | Control port Port A |
| 9194 (TCP) | Control port Port B |
| 9294 (TCP) | Control port Port C |
| 9394 (TCP) | Control port Port D |
| 9084 (TCP) | Reset port status port A |
| 9184 (TCP) | Reset port status port B |
| 9284 (TCP) | Reset port status port C |
| 9384 (TCP) | Reset port status port D |
| 8888 (TCP) | Reset Com-Server |
| 8003 (TCP) | Read the configuration data |
| 8004 (TCP) | Write the configuration data |
| 8512 (UDP) | Inventory (replaced through port 8513) |
| 8513 (UDP) | Inventory |
| 161 (UDP) | SNMP |

## 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:

```
COM1 Properties                        [?][X]
 Port Settings

     Bits per second: [9600        ▼]

          Data bits: [8           ▼]

             Parity: [None        ▼]

          Stop bits: [1           ▼]

       Flow control: [None        ▼]

     [Advanced...]        [Restore Defaults]

        [  OK  ]  [ Cancel ]  [ Apply ]
```
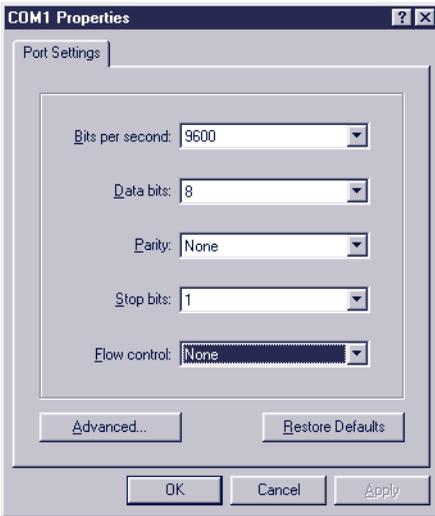
4. After you have confirmed the settings with OK, you will find yourself in the actual terminal window. All keyboard entries are output thorugh the selected COM port. Now turn the Com-Server on and off to reset it and while it is starting up hold down the „x" key (lower case, no Caps Lock!). After about 2-3 seconds the prompt *IP no.:+<ENTER>:* will appear in Hyperterminal.
5. 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.*

## Web application 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 prptocol 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 send transparent to the serially connected device for further processing.

**Example HTTP**
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.*

## WuTility - Invetory and management tool

In Windows environments the WuTility Tool can also be used for inventory and management of Com-Server installations. Pressing a button lists all the Com-Servers located in the local network together with their key data. Following are the most interesting functions which can be run directly from the list:

- ✓ Assigning the IP address, even if the currently set address does not fit in the local network
- ✓ Copying of entire configuration blocks from one Com-Server to any other
- ✓ Creating and archiving configurations
- ✓ Firmware updates

**Where do I obtain this tool?**
The latest version is always published on our Web pages at the following address:

*http://www.wut.de*

The simplest way to navigate from there is by using the menu tree on the left side of the page. Follow the

*Products & Downloads -> Com-Server -> Tools*

path to get to the menu branch containing a direct download link for the tool.

## Hardware reset to factory defaults

In addition to resetting the Com-Server to its factory defaults using the Telnet configuration (Port 1111), this can also be done via hardware. For this purpose all models have two adjacent jumpers on the board. For normal operation these jumpers must be out. To set the factory defaults, proceed as follows:

- Power off the Com-Server and open the enclosure
- Insert *both* jumpers and reconnect the supply voltage. An internal self-test will be performed during which messages will be issued on serial port A. The Fail message in the lines „Port A:" and „TP Test:" can be ignored.
- The self-test will be finished in approx. 20 s, at which point the factory defaults are active.
- Turn off the Com-Server, open both jumpers and close up the housing again.

⚠ *Resetting the non-volatile memory results in a loss of all the settings which are different from the factory defaults, including the IP address. The setting profile of the factory defaults can be replaced by a customer-specific profile. In this cases the customer settings are activated after resetting.*

## Technical Data

### Com-Server Highspeed Industry 58631

| | |
|---|---|
| **Supply voltage ...** | |
| ... DC | 12V - 40V (+/-10%) |
| ... AC | 9V - 24V (+/-10%) |
| **Current draw** | typ. 85mA @24V |
| **Permissible ambient temperatur ...** | |
| ... storage | -40 ... +70°C |
| ... operating, non-cascaded | 0 ... +60°C |
| ... operating, cascaded | 0 ... +50°C |
| **Permissible relative humidity** | 0 - 95% (not condensing) |
| **Network** | 10/100BaseT, RJ45 for STP cables |
| **Galvanic isolation** | min. 500Vrms |
| **Dimensions** | 105 x 75 x 22mm |
| **Weight** | ca. 150g |
| **Serial ports** | 1 x RS232/422/485 configurable, DB9/Male |
| **Baud rates** | 50 to 230.400 kBit/s |
| **Data formats** | 7, 8 data bit, 1, 2 stop bit, NO, EVEN, ODD parity |
| **Flow control** | Hardware-Handshake or Xon/Xoff protocol |

## Com-Server Highspeed Office 58031, 58034

| | |
|---|---|
| **Supply voltage** | 100-250V /50-60Hz with cold-device cable |
| **Current draw** | approx. 3,5W |
| **Permissible ambient temperature ...** ... storage ... operating | -20 ... +60°C 0  ... +50°C |
| **Permissible relative humidity** | 0 - 95% (not condensing) |
| **Network** | 10/100BaseT, RJ45 for STP cables |
| **Galvanic isolation** | min. 500V to Network |
| **Dimensions** | 212x168x40mm |
| **Weight** | approx. 1000g |
| **Serial ports** ... 58031 ... 58034 | 1 x RS232/422/485 configurable 4 x RS232/422/485 configurable all DB9/Male |
| **Baud rates** | 50 to 230.400 kBit/s |
| **Data formats** | 7, 8 data bit, 1, 2 stop bit, NO, EVEN, ODD parity |
| **Flow control** | Hardware-Handshake or Xon/Xoff protocol |

## Com-Server Highspeed 19" 58331, 58334

| | |
|---|---|
| **Supply voltage** | DC 5V +/-5% |
| **Current draw ...**<br>58331<br>58334 | <br>typ. 320mA @5V<br>typ. 360mA @5V |
| **Permissible ambient temperature ...**<br>... Lagerung<br>... Betrieb | <br>-40 ... +70°C<br>0  ... +60°C |
| **Permissible ambient humidity** | 0 - 95% (not condensing) |
| **Network** | 10/100BaseT, RJ45 for STP cables |
| **Galvanic isolation** | min. 500V to network |
| **Dimensions ...**<br>Front panel 58331<br>Front panel 58334<br>Board 58331 & 58334 | <br>approx. 130x40mm (3HEx8TE)<br>approx. 130x60mm (3HEx12TE)<br>160x100mm |
| **Weight ...**<br>58331<br>58334 | <br>approx. 180g<br>approx. 240g |
| **Serial ports ...**<br>58331<br><br>58334 | <br>1 x RS232/422/485 configurable<br>with DB9/male<br>4 x RS232/422/485 configurable<br>with DB9/male |
| **Baud rates** | 50 to 230.400 kBit/s |
| **Data formats** | 7, 8 data bit, 1, 2 stop bit,<br>NO, EVEN, ODD parity |
| **Handshake** | Hardware-Handshake or<br>Xon/Xoff protocol |

## Com-Server Highspeed OEM 58431

| | |
|---|---|
| **Supply voltage** | DC 5V +/-5% |
| **Current draw** | typ. 250mA, max. 325mA @5V |
| **Permissible ambient temperature ...**<br>... storage<br>... operating | <br>-40 ... +70°C<br>0   ... +60°C |
| **Permissible relative humudity** | 0 - 95% (not condensing) |
| **Network** | 10/100BaseT, RJ45 for STP cables |
| **Galvanic isolation** | min. 500Vrms |
| **Dimensions** | 95,8 x 71,1mm |
| **Weight** | ca. 200g |
| **Serial ports** | 1 x TTL, 2mm 12-pin adapter<br>(optional 10pol./RM 2,54mm) |
| **Baud rates** | 50 to 230.400 kBit/s |
| **Data formats** | 7, 8 Data bit, 1, 2 stop bit,<br>NO, EVEN, ODD parity |
| **Flow control** | Hardware-Handshake or<br>Xon/Xoff protocol |

## Com-Server Highspeed Compact 58231

| | |
|---|---|
| **Supply voltage ...** | DC 5V +/-5% |
| **Current draw** | typ. 270mA, max. 345mA @5V |
| **Permissible ambient temperatur ...**<br>... storage<br>... operating | <br>-40 ... +70°C<br>0 ... +50°C |
| **Permissible relative humidity** | 0 - 95% (not condensing) |
| **Network** | 10/100BaseT, RJ45 for STP cables |
| **Galvanic isolation** | min. 500Vrms to network |
| **Dimensions** | 115 x 90 x 34mm |
| **Weight** | approx. 300g |
| **Serial ports** | 1 x RS232/422/485 configurable,<br>DB9/Male |
| **Baud rates** | 50 to 230.400 kBit/s |
| **Data formats** | 7, 8 data bit, 1, 2 stop bit,<br>NO, EVEN, ODD parity |
| **Flow control** | Hardware-Handshake or<br>Xon/Xoff protocol |

## Declaration of conformity

W&T
www.wut.de

### Declaration of conformity
### according to paragraph 10.1 of directive 89/336/EWG

Wiesemann & Theis GmbH hereby confirms that the products

| | |
|---|---|
| **Com-Server Highspeed Industry** | **Type 58631** |
| **Com-Server Highspeed Office, 1-Port** | **Type 58031** |
| **Com-Server Highspeed Office, 4-Port** | **Type 58034** |
| **Com-Server Highspeed Compact, 1-Port** | **Type 58231** |

fulfill the requirements of the directives / regulations specified below:

1.  Emission according to

    1.1. EN 55022-B (1997)
    1.2. EN 61000-3-2 (1996)
    1.3. EN 61000-3-3 (1996)

2.  Noise Immunity according to EN 61000-6-2 (1999):

    2.1. EN 61000-4-2     ESD
    2.2. EN 61000-4-3     Radiated Immunity
    2.3. EN 61000-4-4     Burst
    2.4. EN 61000-4-5     Surge
    2.5. EN 61000-4-6     Conducted Immunity
    2.6. EN 61000-4-8     H-Field
    2.7. EN 61000-4-11    Supply Voltage Dips and Interruptions

3.  Product-specific Low-Voltage Directive for communications technology

    3.1. EN 60950  (1997)

Wuppertal, 12/09/2003

Klaus Meyer

Dipl.-Ing. Rüdiger Theis, Managing Director

Interfaces
für
Netzwerke,

serielle
Schnitt-
stellen

und
Drucker-
schnitt-
stellen

# Index

## Symbole