



Users and Developers Manual

---

B-COMM WIN32 V4.4.2

---



**Users and Developers Manual**  
**B-COMM WIN32**

Bedas ® is a registered trade-mark of Kaba Benzing GmbH

Legic ® is a registered trade-mark of the Kaba group

KABA ® is a registered trade-mark of Kaba Schliesssysteme AG

© 2000 Kaba Benzing GmbH, Schwenningen

All rights reserved. The work and its parts are copyright. Any use in cases other than the ones legally permitted without Benzing's approval is forbidden and punishable. This is especially valid for duplications, translations, microfilm recordings and storage and processing on electronic systems.

---

## Table of Contents

<b>Table of Contents .....</b>	<b>5</b>
<b>1. Introduction.....</b>	<b>10</b>
1.1. Concept .....	11
1.2. System requirements.....	14
1.3. Installation .....	14
<b>2. Using the Program .....</b>	<b>15</b>
<b>3. Program Operation.....</b>	<b>29</b>
3.1. File .....	29
3.1.1. Print system messages .....	30
3.1.2. Display system messages.....	31
3.1.3. Delete system messages.....	31
3.1.4. Debug file(s).....	32
3.1.5. Exit .....	32
3.2. Functions.....	33
3.2.1. Transmit master record file .....	34
3.2.2. Autom. master record download .....	40
3.2.2.1 Father-son-procedure.....	43
3.2.3. Master record upload.....	47
3.2.4. Request bookings .....	49
3.2.5. Control buffer memory.....	51
3.2.6. Set date/time .....	53
3.2.7. Transmit display record.....	55
3.2.8. Transmit transparent data .....	57
3.2.9. Summer/Wintertime setup .....	59
3.2.10. Seconds/User language in data record.....	62
3.3. Configuration.....	64
3.3.1. Client management .....	65
3.3.2. Terminal logon .....	66
3.3.3. Update terminal program number.....	73
3.3.4. Set operating mode.....	74
3.3.5. Add new terminal type .....	76
3.3.6. Initialize communication stream.....	77
3.3.7. Start options .....	84
3.3.8. Directories.....	86
3.3.9. Printer log.....	88
3.3.10. File definitions.....	89
3.3.11. FTP Transfer configuration .....	92
3.3.12. Change password.....	93
3.3.13. Backup .....	94
3.3.14. Restore .....	96

3.3.15.	Scheduler.....	98
3.3.16.	Services .....	100
3.3.17.	Deutsch/English GB/English US .....	102
3.3.18.	Set PC date/time .....	103
3.4.	<i>System setup</i> .....	104
3.4.1.	AT Modem .....	105
3.4.2.	Create modem parameter file .....	107
3.4.3.	Device interface.....	108
3.4.4.	Software download.....	112
3.4.5.	Parameter setting .....	114
3.4.5.1	Parameter upload.....	116
3.4.5.2	Parameter download.....	118
3.4.6.	BETA setup by Telnet.....	120
3.4.7.	BETA setup by ARP and Telnet .....	122
3.5.	<i>Order</i> .....	125
3.5.1.	Display orders .....	126
3.5.2.	Delete all orders .....	129
3.5.3.	Start communication.....	130
3.5.4.	Stop communication.....	132
3.6.	<i>Info</i> .....	133
3.6.1.	About B-COMM .....	134
3.6.2.	System information.....	134
3.6.3.	Version informations .....	135
3.6.4.	Terminal status .....	136
3.6.5.	Licence registration .....	136
3.7.	<i>Help</i> .....	138
3.7.1.	Manual.....	138
3.7.2.	Tooltip help .....	138
<b>4.</b>	<b>The application SHOWDBG .....</b>	<b>141</b>
<b>5.</b>	<b>The application TERMSTAT.....</b>	<b>145</b>
5.1.	<i>Configuration</i> .....	149
5.2.	<i>Data records from TERMSTAT to BCOMMIO</i> .....	150
5.3.	<i>Data records from BCOMMIO to TERMSTAT</i> .....	151
5.4.	<i>Communication between BCOMMIO and TERMSTAT</i> .....	153
5.4.1.	Data records of TERMSTAT .....	153
5.4.2.	Data records of BCOMMIO.....	153
<b>6.</b>	<b>The B-COMM WIN32 file system .....</b>	<b>154</b>
<b>7.</b>	<b>Data Transfer Kernels BCOMMIO and BCOMMFTP .....</b>	<b>155</b>
7.1.	<i>Terminal configuration file TERMDEF.&lt;Stream&gt;</i> .....	161
7.2.	<i>Configuration file BCOMM32.INI</i> .....	164
7.3.	<i>Order file ORDER.&lt;Stream&gt;</i> .....	170
7.3.1.	Order ANALYSE.....	175
7.3.2.	Order MESSAGE.....	175

7.3.3.	Order MODEMPAR .....	176
7.3.4.	Order MRUPLD.....	176
7.3.5.	Order READCFG .....	177
7.3.6.	Order READTDF .....	177
7.3.7.	Order REDAAUTO.....	178
7.3.8.	Order REDAOFF .....	178
7.3.9.	Order REPARA.....	179
7.3.10.	Order SETMODE.....	179
7.3.11.	Order STATUS .....	180
7.3.12.	Order STOPLINE.....	180
7.3.13.	Order SWMEM.....	181
7.3.14.	Order TIMEDAT .....	182
7.3.15.	Order UPDPRGNO.....	183
7.3.16.	Order WRDATRA .....	183
7.3.17.	Order WRPARA .....	184
7.3.18.	Order WRPEAUTO .....	184
7.3.19.	Order WRPEHAAU .....	185
7.4.	System Message File SYSMESS.<Stream> .....	186
7.5.	Parameter File PARAM.DAT.....	200
7.6.	Booking file BOOKING.DAT .....	201
<b>8.</b>	<b>Translation into other Languages.....</b>	<b>202</b>
<b>9.</b>	<b>Call Interface .....</b>	<b>204</b>
9.1.	Shared Memory.....	204
9.2.	DLL functions.....	205
9.2.1.	SetCIHandle .....	205
9.2.2.	SetImmDataHandle .....	205
9.2.3.	InitCISharedMem.....	205
9.2.4.	ExitCISharedMem.....	207
9.2.5.	ReadCISharedMem .....	207
9.2.6.	WriteCISharedMem .....	209
9.2.7.	GetBCOMMIOSStatus.....	209
9.2.8.	ImmediateSTOPLINE .....	210
9.2.9.	SendImmediateData .....	210
9.3.	Program flow in a CI program .....	211
9.4.	The Active-X Control CI_OCX.....	213
9.4.1.	Properties .....	214
9.4.1.1	AutoAcknowledge .....	214
9.4.1.2	AutoDisconnect.....	214
9.4.1.3	BlinkOnError.....	214
9.4.1.4	BlinkInterval.....	215
9.4.1.5	BCOMMIOSStatus.....	215
9.4.1.6	CLError .....	215
9.4.1.7	ConnectInterval .....	215
9.4.1.8	ConnectRetries .....	215
9.4.1.9	DebugLevel .....	216
9.4.1.10	LEDShape .....	216
9.4.1.11	MessageID.....	216
9.4.1.12	OnlineBooking .....	216

---

9.4.1.13	SendInterval .....	217
9.4.1.14	SendRetries.....	217
9.4.1.15	ShowAsLED.....	217
9.4.1.16	ShowMissingTerminal.....	217
9.4.1.17	ShowTrafficlight.....	217
9.4.1.18	StartBCOMMIO.....	218
9.4.1.19	Stream .....	218
9.4.1.20	UpdateInterval.....	218
9.4.1.21	WindowSize .....	218
9.4.2.	Methods.....	219
9.4.2.1	ConnectToBCOMMIO.....	219
9.4.2.2	DisconnectFromBCOMMIO .....	219
9.4.2.3	ReadBCOMMIOStatus .....	219
9.4.2.4	SendDataToBCOMMIO .....	220
9.4.2.5	ShutdownBCOMMIO .....	220
9.4.2.6	SendImmediateDataToBCOMMIO.....	220
9.4.3.	Events.....	221
9.4.3.1	evBCOMMIODown.....	221
9.4.3.2	evBCOMMIORead .....	221
9.4.3.3	evBCOMMIOStatus.....	221
9.4.3.4	evBCOMMIUp.....	221
9.4.3.5	evDataRecordReceived .....	222
9.4.3.6	evBCOMMIOOrderStart.....	222
9.4.3.7	evBCOMMIOOrderEnd.....	222
9.4.3.8	evBCOMMIOOrderRestart .....	223
9.4.3.9	evImmediateDataAck .....	224
9.4.3.10	evImmediateDataNack .....	224
9.5.	Messages .....	226
9.5.1.	SM_WRITE .....	227
9.5.2.	SM_READ .....	227
9.5.3.	BCOMMIO_DOWN.....	228
9.5.4.	BCOMMIO_UP .....	228
9.5.5.	BCOMMIO_STATUS .....	229
9.5.6.	IMM_DATA_ACK.....	229
9.5.7.	IMM_DATA_NACK .....	230
9.5.8.	ORDER_START.....	230
9.5.9.	ORDER_END .....	231
9.5.10.	ORDER_REP_START .....	231
9.6.	DDE .....	232
9.7.	Protocol.....	232
9.8.	Commands from the CI Program to BCOMMIO .....	233
9.9.	Additional Records from BCOMMIO to the CI Program .....	236
9.10.	Flow of Communication.....	237
9.11.	Configuration of the Call Interface.....	240
<b>10.</b>	<b>Table of Figures.....</b>	<b>243</b>
<b>11.</b>	<b>Index.....</b>	<b>246</b>





---

## 1. Introduction

The B-COMM WIN32 program package serves to transfer data between Kaba Benzing terminals and a host.

The program transfers the records acquired by the terminals in files or sends it by use of a defined interface, the so called Call Interface (see chapter 9 Call Interface) to an application. These files or data records can be linked to any application, e.g. time recording or the wages package etc.

Data such as user information, the time and the date or master personnel data can also be transferred from the host to a terminal. Furthermore, the parameter settings of terminals can be managed centrally on the host and transferred to the terminals.

With the B-COMM WIN32 version, Kaba Benzing terminals can be operated in the ONLINE, OFFLINE and AUTONOMOUS modes. In doing so, data is exchanged between the terminals and the host in files or by the Call Interface. Communication orders are also sent in corresponding files.

The software package essentially contains three function modules. Installation is supported by an installation routine. The I/O module (input/output) BCOMMIO (respectively BCOMMFTP for FTP data transfer) is responsible for actual communication with Kaba Benzing terminals. The BCWIN maintenance program serves to control and operate the I/O module.

**BCWIN is not a replacement for an application written by a software partner, but is only used to configure the terminal system.**

Using the maintenance program, all functions of the package can be controlled and coordinated manually. The tasks of the control program can be assumed by any application program. Various commands, which can be stored in the order files by the application, are available for this purpose.

The following can be carried out using the program package described here:

- configuring the terminal system
- handling communication with the terminal system
- maintenance of the terminal system

---

## 1.1. Concept

The program package contains 13 modules:

- Installation program SETUP
- Maintenance program BCWIN
- Communication program BCOMMIO for LAN and serial streams (see chapter 7 Data Transfer Kernels BCOMMIO)
- Communication program BCIOSRV for LAN and serial streams to work as a service (see chapter 3.3.16 Services) under Windows NT, Windows 2000 and Windows XP
- Function library BCWIN.DLL (see chapter 9.2 DLL function) for use of the Call Interface
- Demo programs BCCITEST, BCCIWIN and BCCIDEMO for use of the Call Interface via Shared Memory (see chapter 9.1 Shared Memory)
- Demo program BCCI\_DDE for use of the Call Interface via DDE
- Starter program BCIOSTRT to start the streams from a remote PC
- Converter program DDE2SMEM for conversion between DDE and Shared Memory
- Program SHOWDBG for an easy display and analysis of the debug files, which are created by the I/O module BCOMMIO (see chapter 4 The application SHOWDBG)
- Program BCUTIL for manipulating the terminal configuration and order files
- Commandline program WRTORDER, with which orders can be inserted into an order file

---

Additionally you can order the following chargeable options:

- Communication program BCOMMFTP for FTP transfer of master record files, parameter files and transparent data to terminals of type Bedanet 938Ø (B-Net 938Ø)
- Program TERMSTAT to graphically display the state of the communication streams and the connected terminals (see chapter 5 The application TERMSTAT)
- Parameter editors for terminals of the series 929Ø, 93xx inclusive Bedanet 938Ø (B-Net 938Ø), 952Ø, 954Ø, 956Ø and 98xx
- Programming module AVISO to program 929Ø terminals
- Programming module BEDAPRO to program 956Ø terminals

---

When the communication program BCOMMIO is mentioned in this manual, then everything is always valid also for the service version BCIOSRV.

**Under the OS versions Windows NT 4.0, Windows 2000 and Windows XP the communication stream BCOMMIO could also be used as a service.**

The BCOMMIO communication program handles all communication with the terminals and stores all received data in files. The terminals can be addressed via a serial interface, a modem, Ethernet, or Token Ring. Therefore so called streams have to be created and configured (see chapter 3.3.6 Initialize communication stream). A maximum of 30 streams can be created (stream numbers 0 to 29). One stream with the number 30 is always existent and therefore has not to be created by the user. This stream is used for transmitting master records (see chapter 3.2.1 Transmit master record file and 3.2.2 Autom. master record download), transparent data (see chapter 3.2.8 Transmit transparent data) and parameter settings (see chapter 3.4.5 Parameter setting) to one or more Bedanet 9380 (B-Net 9380) terminals. Within this stream, the communication with the terminals is done by the I/O kernel BCOMMFTP, in case you bought this chargeable option

All data received by the BCOMMIO communication program can also be transferred to a call interface for evaluation. Control of the kernel and maintenance of all required files can be taken on completely by an application. All data transferred by an application to the BCOMMIO data transfer kernel is passed on to the corresponding terminals by BCOMMIO.

The source code for the Call Interface demo programs BCCIWIN, BCCITEST (Shared Memory) and BCCI\_DDE (DDE) in Microsoft Visual Basic is supplied with the program package. Also the source code for the demo program BCCIDEMO (Shared Memory) in Visual C++ is supplied. This source code can be used by software partners to develop their own applications tailored individually to the needs of the user, e.g. access control and time recording etc.

The BCOMMIO/BCOMMFTP communication program can also be controlled with the BCWIN maintenance program. All necessary files can be managed with the BCWIN maintenance program.

---

## 1.2. System requirements

The B-COMM WIN32 program package can be used on a IBM PC and compatible computers, **but not on PC's with an Alpha processor.**

The following minimum prerequisites must be met:

CPU:	Pentium / 90 MHz
Hard disk:	at least 20 MB free disk capacity
RAM:	64 MB, 128 MB recommended
Interfaces:	V.24 interface with FIFO buffer (UART 16550) and/or Ethernet or Token Ring
Operating system:	Windows ME, Windows NT 4.0 with at least Service Pack 3, Windows 2000 or Windows XP TCP/IP must additionally be installed, even if you don't want to communicate via LAN

## 1.3. Installation

At first start Windows and insert data carrier (disk resp. CD). Select the RUN menu item from the START menu. In the command line specify the drive you are using and the installation program, e.g. **a:\setup**. Follow the instructions until the installation is finished.

**Attention: for uninstalling B-COMM WIN32 the mechanisms of Windows must be used, e.g. the Add/Remove Programs item in the control panel!**

**The installation program automatically detects an earlier version of B-COMM WIN32 on the target PC. In this case, the configuration of the installed version will be left untouched and only newer files are installed.**

---

## 2. Using the Program

You must first start the program via a mouse click onto the program B-COMM WIN32 in the program group B-COMM WIN32. Then the following screen appears:

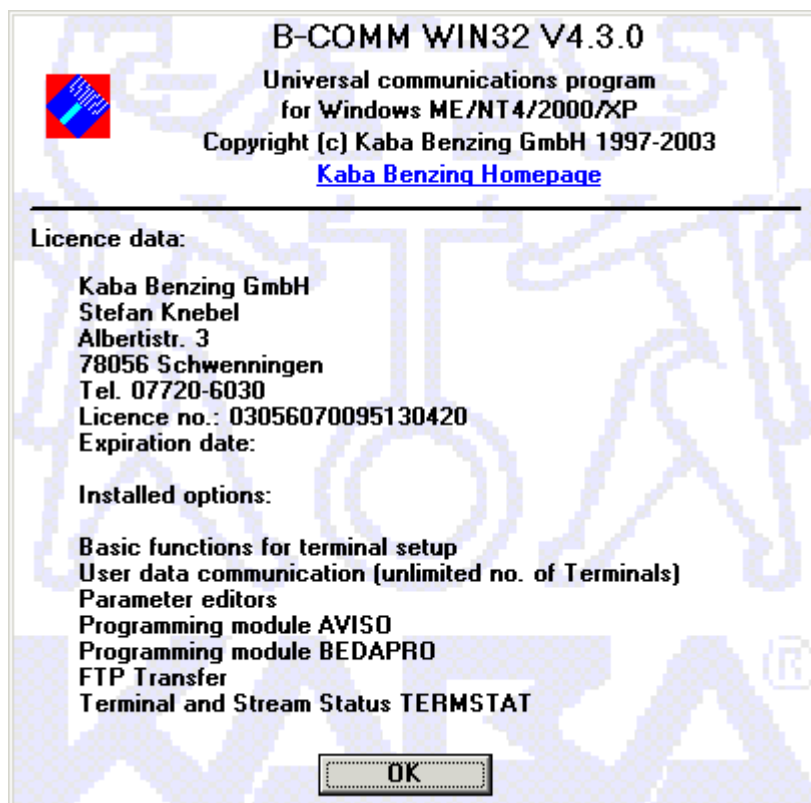
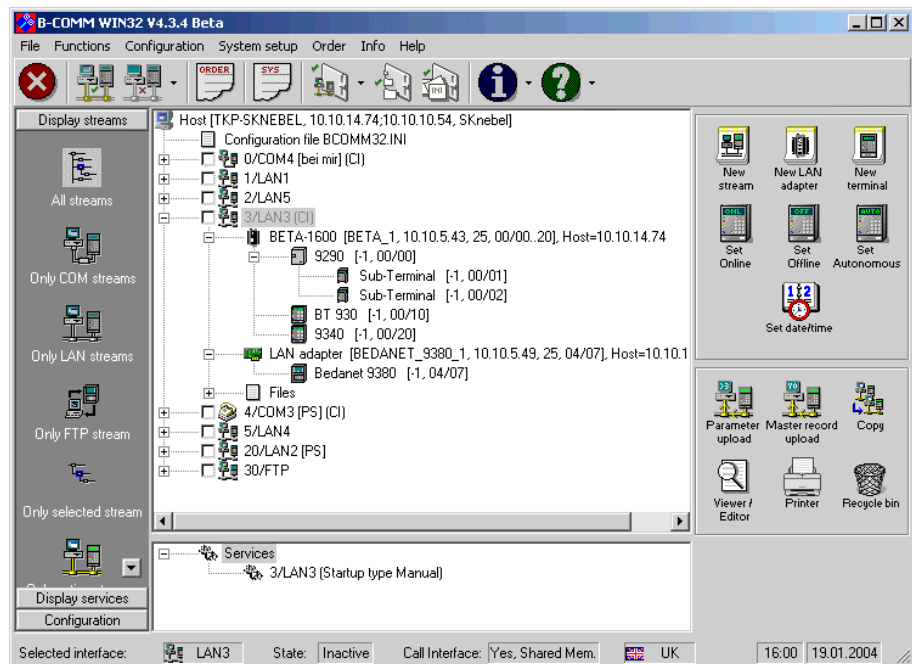


Figure 2-1: Info mask

This screen displays licence data. The following screen appears after the button **OK** is activated:



**Figure 2-2: Main window of BCWIN**

The interface selected last and the state of the corresponding data transfer kernel are displayed in the bottom of the screen, e.g. Selected interface: COM2, Status: Inactive, Call Interface: No.

All configured streams are displayed as a tree view below the toolbar. In that tree view, the configured streams, the network adapters and terminals connected to a stream and the files related to that stream as well as the configuration file BCOMM32.INI are displayed as a node.

In a second tree view you can see those streams, which are configured as services (see also chapter 3.3.16 Services). This view is only for informational purposes and has no functionality. It is only visible under Windows NT4, Windows 2000 and Windows XP, because only these operating systems support services.

By use of the icons in the so called menu bar (on the left of the main screen) you can select, which streams or services you wish to be displayed.



---


Here some short explanations to the possible tree view nodes:

- Here the PC is displayed, on which B-COMM WIN32 is installed. Within the brackets you can see the name of the PC, its IP address and the name of the user, who is currently logged on. Double-clicking on this node expands or collapses all subnodes.

 Host [TKP-SKNEBEL, 10.10.10.54, SKnebel]

- This is the configuration file BCOMM32.INI (see chapter 7.2 Configuration file BCOMM32.INI), which is located either in the Windows directory or in the installation directory of B-COMM WIN32.

 Configuration file BCOMM32.INI

-  1/LAN1 [Main building] (CI)

This is the display of a stream. Behind the symbol of the stream, you can see the stream's number and its interface, e.g. stream 1, interface LAN1. If a description of the stream is given (see chapter 3.3.6 Initialize communication stream) , it is displayed in square brackets. When the stream is configured to work with Call Interface (see chapter 9 Call Interface), this is also indicated. By single-clicking the little checkbox left of the stream symbol the stream can be started or stopped.. An active stream will be displayed as follows:


☒  1/LAN1 [Main building] (CI)

When the stream is started as a service (see chapter 3.3.16 Services), then the text color is blue instead of green.


- When the terminals are connected to a modem, the following symbol is displayed for the modem:

 4/COM3  
 Modem [366]

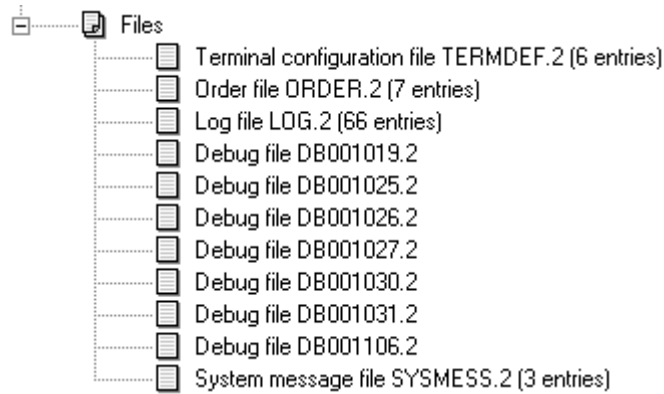
- A terminal is displayed with its symbol, behind which the line number, the group and device address (GID/DID) and, in case entered by a user, the location is displayed in brackets.

 9290 [-1, 00/00, Testplatz]

- Behind the symbol of a network adapter (BETA/BETOR) you can see in brackets its IP address, the DNS name as well as the BETA system ID and the address range of the connected terminal(s) with the following syntax: either <GID>/<DID from>..<DID to> (BETA/BETOR 1600) or <GID>/<DID> (BETA/BETOR 100, integrated network adapter). After that, the host IP of the BETA is displayed.

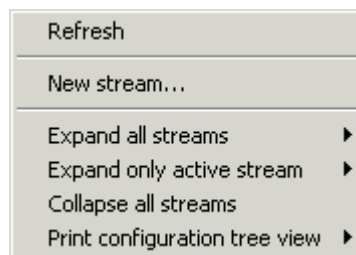
 BETA-1600 [BETA\_1, 10.10.5.43, 25, 00/00..20], Host=10.10.14.74

- Additionally the files concerning the stream are displayed as follows:



Each node has its own context menu, that could be called by a click with the right mouse button on the node.

Clicking on the topmost node of the tree view, which represents the host PC, on which B-COMM WIN32 is installed, the following context menu appears:



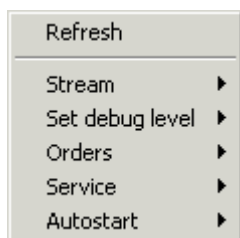
**Figure 2-3: Context menu of the PC symbol**

The following menu items are available:

REFRESH:	all configuration files are re-read and the tree view is updated
NEW STREAM:	the configuration dialog for streams is called
EXPAND ALL STREAMS:	all nodes of the tree view are expanded so that all nodes are visible
EXPAND ONLY ACTIVE STREAM:	only the nodes of the active streams are expanded, all other nodes are collapsed
COLLAPSE ALL STREAMS:	all nodes are collapsed so that only the nodes of the configured streams are visible
PRINT CONFIGURATION TREE VIEW:	the whole configuration tree view is printed on the default printer

---

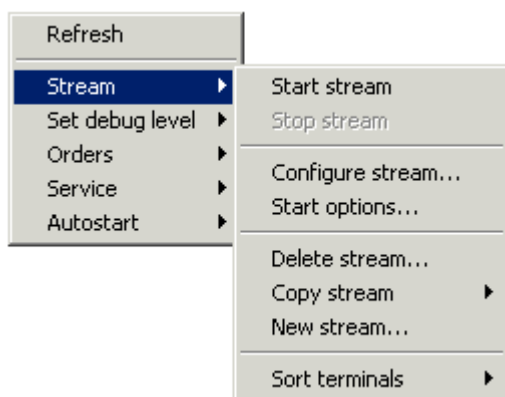
Right-clicking on the icon of a stream brings up a popup menu:



**Figure 2-4: Context menu of a stream symbol**

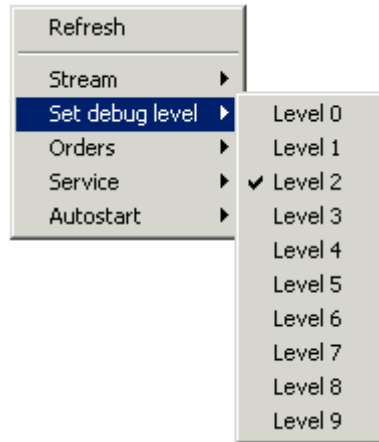
The following menu items are displayed, depending on the state of the system and the operating system used:

REFRESH	the selected stream is refreshed, which means that all info about the stream is reread from the appropriate files.
---------	--

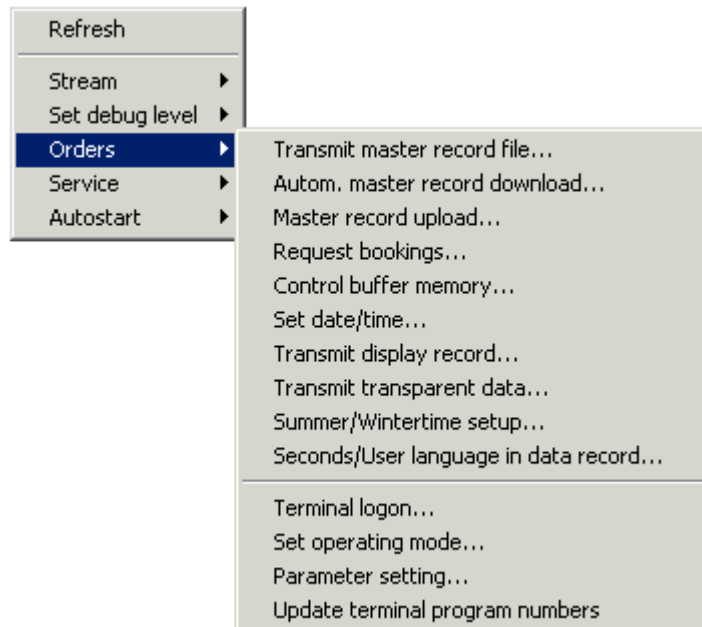


START STREAM	the I/O kernel BCOMMIO of the stream is started
STOP STREAM	the I/O kernel BCOMMIO of the stream is shut down
CONFIGURE STREAM...	the configuration dialog of the stream is called (see chapter 3.3.6 Initialize communication stream)
START OPTIONS...	the configuration dialog for the start options is called
DELETE STREAM...	the stream could be deleted by use of it's configuration dialog
Copy stream	a copy of the stream is created. The new stream number and interface are automatically set to the next possible values. By the sub menu you can select whether to copy the terminals too or not.

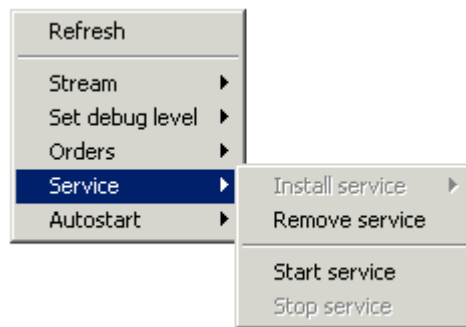
- NEW STREAM... the configuration dialog is called and you can configure a new stream
- SORT TERMINALS you can sort the terminals connected to the stream by GID/DID or – in case of a COM stream with modem – by the call number



- Set DEBUG LEVEL with this menu item you can set the debug level to values between 0 and 9 (see chapter 3.3.7 Start options)

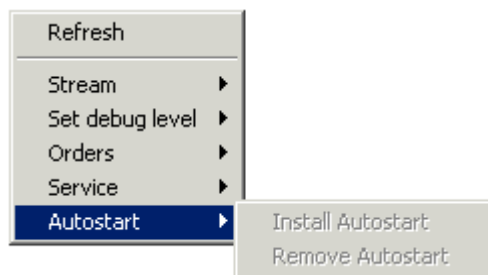


- ORDERS with this menu item and its submenu items you can call the same dialogs as with the main menu item FUNCTIONS



INSTALL SERVICE  
REMOVE SERVICE  
START SERVICE  
STOP SERVICE

these four menu items are available only under Windows NT, Windows 2000 or Windows XP. You can enter the stream into the service manager or remove it from the service manager. Also an installed service can be started or stopped.  
(see chapter 3.3.16 Services)



INSTALL AUTOSTART  
REMOVE AUTOSTART

with these two menu items the stream can be started automatically when Windows is starting or this feature can be removed

**Tip:** When a stream is installed as Autostart, BCWIN makes an entry in the Registry of Windows. You can find that entry under the following key:

**HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run**

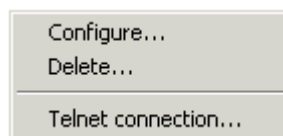
Under this key an entry like **Line<x>** is made, where **<x>** is the number of the stream.

**Attention:** when uninstalling B-COMM WIN32 the Autostart entries will **not** be removed automatically. You should do that manually or by use of B-COMM WIN32 before uninstalling it.

Clicking on the little checkbox of the node starts or stops the corresponding I/O kernel BCOMMIO.

---

By a mouse click with the right mouse button onto the node of a LAN adapter the following context menu appears:

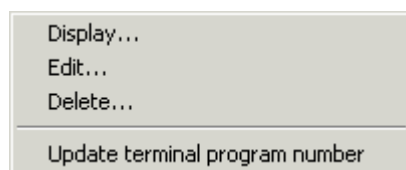


**Figure 2-5: Context menu of a LAN adapter symbol**

The following menu items are available:

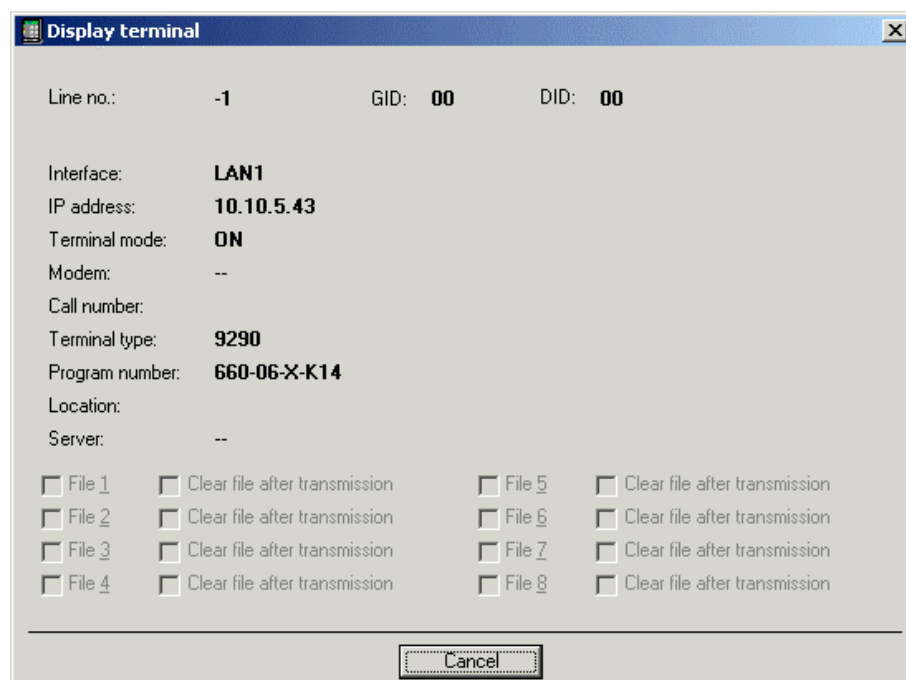
CONFIGURE:    the dialog for the configuration of a stream is called  
DELETE:        the dialog for the configuration of a stream is called

When a terminal is right-clicked you will get a context menu with the following items:



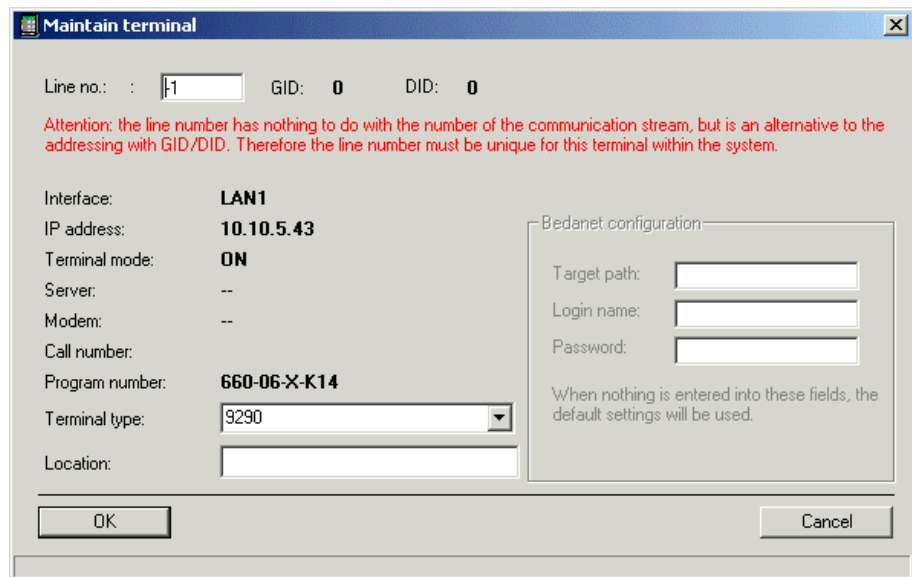
**Figure 2-6: Context menu of a terminal symbol**

DISPLAY: a dialog appears, in which all properties of the terminal are displayed:



**Figure 2-7: Terminal info**

EDIT: a dialog appears, in which you can edit some properties of the terminal, e.g. the Location:

The image shows a Windows-style dialog box titled "Maintain terminal". At the top, it has fields for "Line no.:" (containing "1"), "GiD: 0", and "DiD: 0". Below these is a red text warning: "Attention: the line number has nothing to do with the number of the communication stream, but is an alternative to the addressing with GiD/DiD. Therefore the line number must be unique for this terminal within the system." The main area contains several fields: "Interface:" (LAN1), "IP address:" (10.10.5.43), "Terminal mode:" (ON), "Server:" (--), "Modem:" (--), "Call number:" (empty), "Program number:" (660-06-X-K14), "Terminal type:" (a dropdown menu showing 9290), and "Location:" (an empty text box). To the right of these fields is a sub-dialog titled "Bedarinet configuration" with fields for "Target path:", "Login name:", and "Password:". Below these fields is a note: "When nothing is entered into these fields, the default settings will be used." At the bottom of the main dialog are "OK" and "Cancel" buttons.

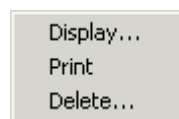
**Figure 2-8: Maintain a terminal**

DELETE: you can delete the terminal from the configuration after a confirmation

UPDATE TERMINAL PROGRAM NUMBER:  
the program number of the terminal is requested and inserted into the Terminal configuration file

TERMDEF.<Stream>

When you click on a file, you can select between the following menu items, depending on the type of file:



**Figure 2-9: Context menu of a file symbol**

DISPLAY: depending on the file, a special program or dialog is called to display the file in a special format

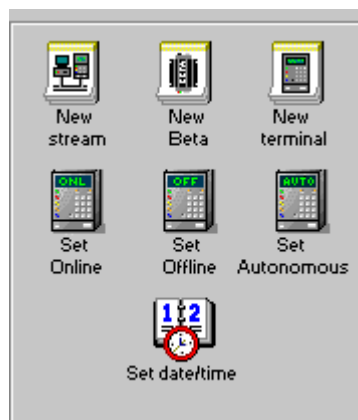
PRINT: parts of the file or the whole file is printed to the default system printer

DELETE: the file is deleted after a confirmation







---


On the right of the tree view you can see a few icons, with which you can execute or initiate some functions of B-COMM by use of drag and drop:



**Figure 2-10: Drag and Drop symbols 1**

You can drag the icons in the upper frame onto a node of the tree view to initiate the function:






-  Dragging this icon onto the PC node of the tree view creates a new stream. Therefore the configuration dialog is called (see chapter 3.3.6 Initialize communication stream)
-  When you drag this icon onto the node of a stream in the tree view, a new LAN adapter (Beta) is entered into the configuration. First a dialog appears in which you have to enter the IP address of the new adapter and then the configuration dialog is called again (see chapter 3.3.6 Initialize communication stream)
-  You can drag this icon either onto the node of a stream or onto the node of a LAN adapter. Then the dialog for log-on of terminals is called (see chapter 3.3.2 Terminal logon)
-  These icons are used to switch the operating mode of the terminal(s) to ONLINE, OFFLINE or AUTONOMOUS by dragging them either onto a terminal node in the tree view (only this terminal's mode is set) or onto a stream node (all terminals connected to the stream are set to the desired operating mode).


- 
- 
 When this icon is dragged onto the node of a stream in the tree view, the system date and time is sent to all connected terminals (see chapter Set date/time). When you drag the icon onto the node of a terminal, date and time are only sent to that terminal. Dragging this icon onto the PC node brings up the dialog for setting the PC's date and time (see chapter 3.3.18 Set PC date/time).



**Figure 2-11: Drag and Drop symbols 2**

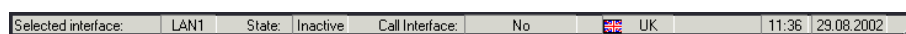
You can drag nodes from the tree view onto one of the above icons to execute a special function of B-COMM (the icons can't be dragged onto the tree view):

- 
 You can drag the node of a stream or of a terminal onto this icon to start a parameter upload of all terminals or one particular terminal (see chapter 3.4.5 Parameter setting)
- 
 You can drag the node of a stream or of a terminal onto this icon to start a master record upload of all terminals or one particular terminal (see chapter 3.2.3 Master record upload)
- 
 When you drag the node of a stream onto this icon, a copy of the stream is created. The stream number and interface of the new stream are automatically set to the next possible values. When terminals are logged on the stream, these terminals are copied too. After the copy you can decide whether to call immediately the configuration dialog for the new stream (see chapter 3.3.6 Initialize communication stream).
- 
 You can drag nearly every node of the tree view onto this icon. Depending on the dragged node, a special program or dialog is called to view details of that node (e.g. the program SHOWDBG is started when the node of a debug file was dragged, or the Windows editor is called when the node of the configuration file BCOMM32.INI was dragged onto the icon).
- 
 When you drag the PC node onto this icon, a graphical printout of the complete configuration is created. When you drag a file onto the icon, that file is printed.

- 
-  When a node of the tree view is dragged onto this icon, the corresponding file, LAN adapter, terminal or stream is deleted from the configuration.

---

Int the status bar you can initiate various actions by clicking onto one of the following fields:



**Figure 2-12: The status bar**

When you click onto the display of the selected interface (**COM2** in the picture above), the next configured stream in the tree view is selected and the status bar is updated.

When you click onto the display of the status of the interface (**Inactive** in the picture above), the appropriate stream is started or stopped.

When you click onto the display of the language (**UK** in the picture above), the next possible language is set. After the first installation of B-COMM you can switch between English and German. (see also chapter 8 Translation into other Languages)

When you click onto time or date display, the dialog for setting the PC's date and time is called (see chapter 3.3.18 Set PC date/time)

### 3. Program Operation

As with common Windows programs, the program is operated with the mouse or the keyboard. Please refer to your Windows documentation for further information on Windows and how to operate.

#### 3.1. File

When you select the FILE menu, you can choose between the PRINT LOG FILE..., DISPLAY LOG FILE..., DELETE LOG FILE..., PRINT SYSTEM MESSAGES..., DISPLAY SYSTEM MESSAGES..., DELETE SYSTEM MESSAGES..., DEBUG FILE(S)... and EXIT menu items.

**The menu items for the Log files are only available for streams 0 to 29, because the FTP stream 30 doesn't write a Log file.**

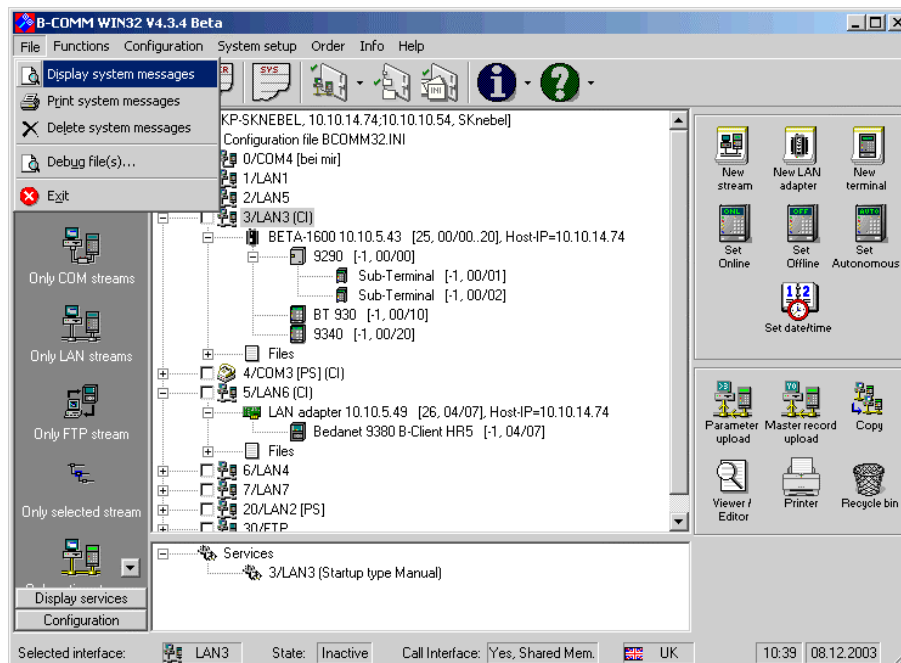


Figure 3-1: Menu File

---

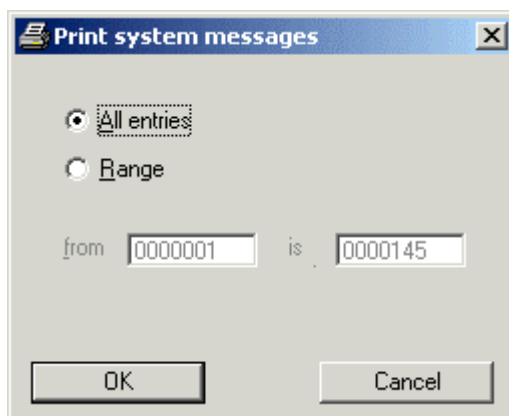
### 3.1.1. Print system messages

System messages (S, X and O records) of the terminals are entered in the system message file (see chapter 7.4 System Message File SYSMESS.<Stream>). Further error messages originating from BCOMMIO (e.g. interface not ready etc.) are also stored in this file. In doing so, a separate system message file is created for each stream.

**The system message files of the streams 0 to 30 are binary ring files. The maximum size of the files could be specified by the user (see chapter 3.3.10 File definitions). The files could not be viewed or edited with an ASCII editor.**

The messages contain the date and time when the error occurred. Each error message is assigned an error code, the record received by the terminal and a text.

By selecting the PRINT SYSTEM MESSAGES... menu item, you can output the system message file on the system printer. In doing so, the system message file assigned to the last selected interface is used. You can either print all entries of the file or just an area of the file. Therefore the following dialog is displayed:

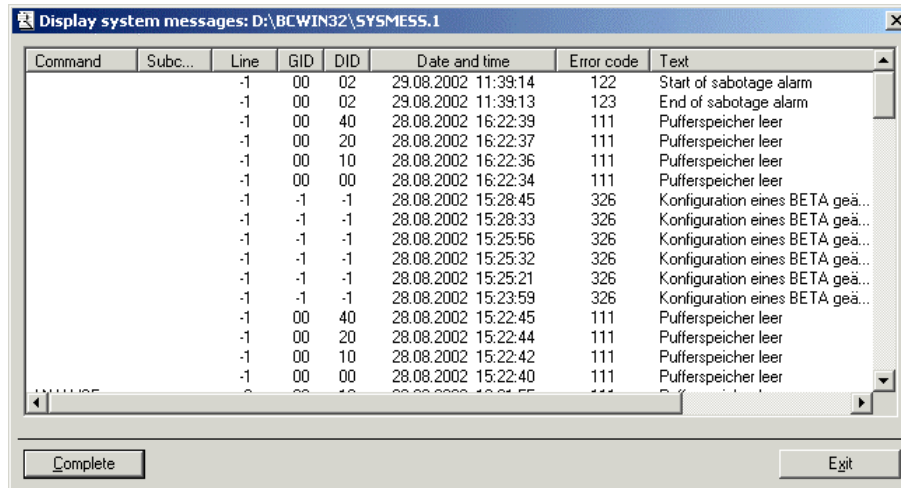


**Figure 3-2: Print system messages**

Activating the button **OK** starts the printout.

### 3.1.2. Display system messages

By selecting the DISPLAY SYSTEM MESSAGES... menu item, you can display the system message file. In doing so, the system message file assigned to the last selected interface is used.



The screenshot shows a window titled "Display system messages: D:\BCWIN32\SYSMESS.1". It contains a table with the following columns: Command, Subc..., Line, GID, DID, Date and time, Error code, and Text. The table lists various system messages, including sabotage alarms and buffer status reports. At the bottom of the window, there are two buttons: "Complete" and "Exit".

Command	Subc...	Line	GID	DID	Date and time	Error code	Text
		-1	00	02	29.08.2002 11:39:14	122	Start of sabotage alarm
		-1	00	02	29.08.2002 11:39:13	123	End of sabotage alarm
		-1	00	40	28.08.2002 16:22:39	111	Pufferspeicher leer
		-1	00	20	28.08.2002 16:22:37	111	Pufferspeicher leer
		-1	00	10	28.08.2002 16:22:36	111	Pufferspeicher leer
		-1	00	00	28.08.2002 16:22:34	111	Pufferspeicher leer
		-1	-1	-1	28.08.2002 15:28:45	326	Konfiguration eines BETA geä...
		-1	-1	-1	28.08.2002 15:28:33	326	Konfiguration eines BETA geä...
		-1	-1	-1	28.08.2002 15:25:56	326	Konfiguration eines BETA geä...
		-1	-1	-1	28.08.2002 15:25:32	326	Konfiguration eines BETA geä...
		-1	-1	-1	28.08.2002 15:25:21	326	Konfiguration eines BETA geä...
		-1	-1	-1	28.08.2002 15:23:59	326	Konfiguration eines BETA geä...
		-1	00	40	28.08.2002 15:22:45	111	Pufferspeicher leer
		-1	00	20	28.08.2002 15:22:44	111	Pufferspeicher leer
		-1	00	10	28.08.2002 15:22:42	111	Pufferspeicher leer
		-1	00	00	28.08.2002 15:22:40	111	Pufferspeicher leer

Figure 3-3: Display system messages

At first only the most recent 100 system messages are displayed for speed reasons. By pressing the button **Complete**, you can display the whole file. This can last a few minutes when displaying a large system message file.

You can exit the DISPLAY SYSTEM MESSAGES... menu item by selecting the **Exit** button.

### 3.1.3. Delete system messages

You can delete the system message file (see chapter 7.4 System Message File SYSMESS.<Stream>) with this menu item. To do this, you must also activate the **Yes** button. In doing so, the system message file that is assigned to the last selected interface is deleted. You can cancel this menu item by selecting the **No** button.

---

#### **3.1.4. Debug file(s)**

With this menu item you can launch the external program SHOWDBG (see chapter 4 The application SHOWDBG) for displaying and/or analyzing the debug files created by BCOMMIO/BCOMMFTP.

#### **3.1.5. Exit**

BCWIN is ended.



## 3.2. Functions

After selecting the menu FUNCTIONS you can choose between the menu items TRANSMIT MASTER RECORD FILE..., AUTOM. MASTER RECORD DOWNLOAD..., MASTER RECORD UPLOAD..., REQUEST BOOKINGS..., CONTROL BUFFER MEMORY..., SET DATE/TIME..., TRANSMIT DISPLAY RECORD..., TRANSMIT TRANSPARENT DATA... and SOMMER/WINTERTIME SETUP....

**For the FTP stream 30 only the menu items TRANSMIT MASTER RECORD FILE..., AUTOM. MASTER RECORD DOWNLOAD... and TRANSMIT TRANSPARENT DATA... are available.**

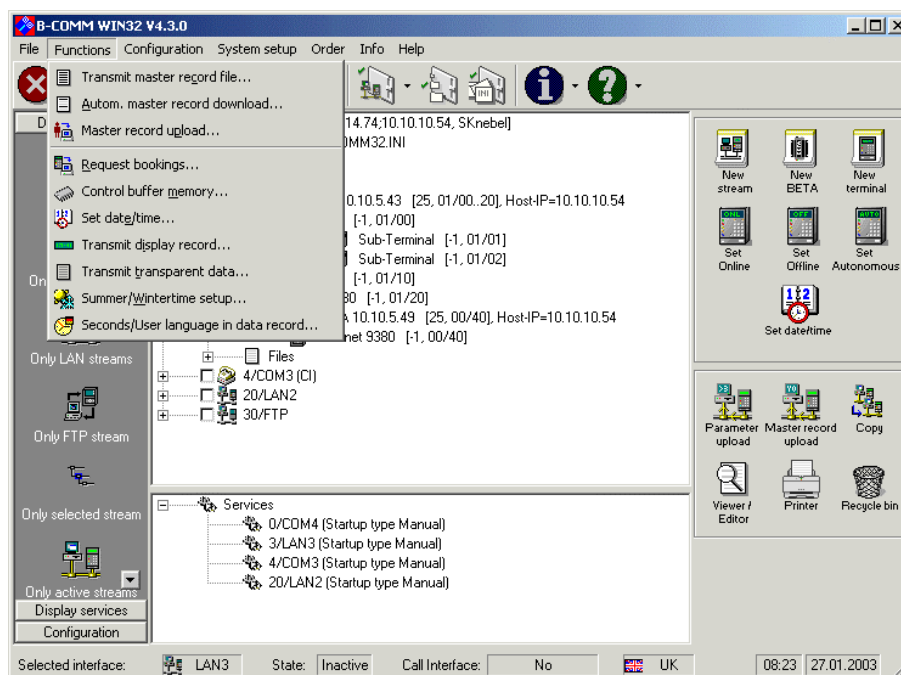


Figure 3-4: Menu Functions

---

### 3.2.1. Transmit master record file

An existing master record file can be sent to the terminals. The master record file must be stored as an **ASCII file**. **Every master record in this file must contain the addressing information**. It is allowed to use wildcards  $\Delta$  ( $7F_{\text{hex}}$ ) in the addressing information in the master records.

#### Example 1:

**@@ Y000000001000200FFFF00**

The master record used in example 1 can only be sent to the terminal with the group/device address 00/00.

#### Example 2:

**$\Delta\Delta$  Y000000001000200FFFF00**

The master record used in example 2 can be sent to all existing terminals.

On terminal log-on, see chapter 3.3.2 Terminal logon, a line no. which can be used as an address alternatively to the GID/DID can be assigned to each terminal. Instead of the group and device identification, the line no. can also be used in the address of master records. To transmit a master record to all existing terminals, line no. 000 must be entered.

#### Example 3:

**004 Y000000001000200FFFF00**

The master record used in example 3 can only be transmitted to the terminal with line no. 004.

#### Example 4:

**000 Y000000001000200FFFF00**

The master record used in example 4 can be transmitted to all terminals existing.

The following table shows the relationship between logical device addressing and the corresponding transferred ASCII characters.

Logical group/device identification	ASCII character (hex)	Logical device identification	ASCII character (hex)
ØØ	@ (4Ø)	3Ø	^ (5E)
Ø1	A (41)	31	_ (5F)
Ø2	B (42)	32	` (6Ø)
Ø3	C (43)	33	a (61)
Ø4	D (44)	34	b (62)
Ø5	E (45)	35	c (63)
Ø6	F (46)	36	d (64)
Ø7	G (47)	37	e (65)
Ø8	H (48)	38	f (66)
Ø9	I (49)	39	g (67)
1Ø	J (4A)	4Ø	h (68)
11	K (4B)	41	i (69)
12	L (4C)	42	j (6A)
13	M (4D)	43	k (6B)
14	N (4E)	44	l (6C)
15	O (4F)	45	m (6D)
16	P (5Ø)	46	n (6E)
17	Q (51)	47	o (6F)
18	R (52)	48	p (7Ø)
19	S (53)	49	q (71)
2Ø	T (54)	5Ø	r (72)
21	U (55)	51	s (73)
22	V (56)	52	t (74)
23	W (57)	53	u (75)
24	X (58)	54	v (76)
25	Y (59)	55	w (77)
26	Z (5A)	56	x (78)
27	[ (5B)	57	y (79)
28	\ (5C)	58	z (7A)
29	] (5D)	59	{ (7B)

The master record file is sent **in blocks of 1Ø master records each**. **Booking is still possible at the terminals.** After a block has been sent, any pending data is fetched from the terminal. Existing master records remain stored in the terminal.

The following screen is displayed when you select the TRANSMIT MASTER RECORD FILE... menu item:

L-No.	GID	DID	Type	Location
<input type="checkbox"/> 0	*	*	All listed terminals	
<input type="checkbox"/> -1	01	*	All listed terminals of group 01	
<input type="checkbox"/> -1	01	60	9840	
<input checked="" type="checkbox"/> -1	01	00	9290	
<input type="checkbox"/> -1	01	10	9340	
<input type="checkbox"/> -1	01	20	BT 930	

Click on the checkbox to select/deselect a terminal

Master record file:

Data file:

☐ Start time-controlled (scheduler has to be started separately by "Configuration->Scheduler...")

OK Display... Maintain... Cancel

List of the available terminals, select one or more terminals

**Figure 3-5: Transmit master record file**

When the FTP stream 30 is the active stream, in the terminal list all Bedanet 9380 (B-Net 9380) terminals of all configured streams are displayed. In the terminal list you have to select the desired terminals by activating the check box at the beginning of each line of the list. To the selected terminal only those master records are sent, which contain the appropriate address.

Enter the file that you wish to send in the **Master record file** text box. The master record file must be in the path that you specified in the **Path for master records** box during installation (see chapter 3.3.8 Directories).

All data transferred from the terminal during execution of this function is stored in the file that you have specified in the **Data file** text box.

Shall the master record file transmit be time-controlled, activate the **Start time-controlled** check box (see chapter 3.3.15 Scheduler). This check box is not available when the download should be done by BCOMMFTP.

#### Available buttons:

##### **OK**

Pressing the button **OK** stores the order in the (time-controlled) order file. This button is disabled until at least one terminal is selected.

##### **Display**

With the button **Display** you can view more detailed information about the selected terminal.

##### **Maintain**

By pressing the **Maintain** button, you can edit the master record file after the message: „**Master record creation for test purposes...format of devices 9220/40/60, 9340/60 and 9520/40**“ and after entering the password. Only master record files containing up to 5000 master records can be edited. The following screen appears for this purpose:

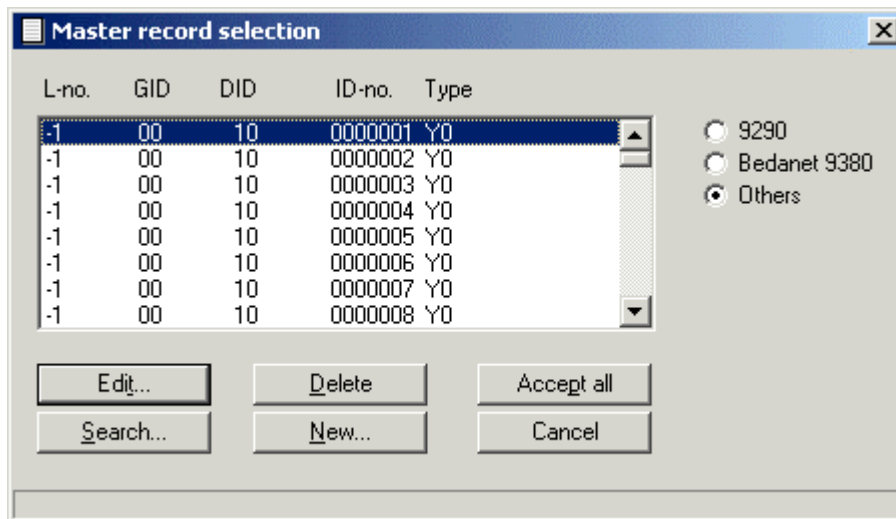


Figure 3-6: Select master record

You can highlight a specific master record with the cursor keys or by clicking it with the mouse. BCWIN tries to detect the type of the highlighted terminal and sets one of the option fields **9290**, **Bedanet (B-Net) 9380** or **Others** accordingly.

### Available buttons:

#### **Edit**

By clicking the **Edit** button, you can edit the highlighted master record.

#### **Search**

By clicking the **Search** button, you can search for a specific master record in the master record file. After activating the **Search** button, you must enter the ID number. A master record that has been found is highlighted when displayed on the screen.

#### **Delete**

By clicking the **Delete** button, you can delete the highlighted master record.

#### **New**

You can create a new master record with the **New** button. To do this, the following screen appears:

**Master record maintenance**

Line no. . . . -1      GID: 0      DID: 10

ID no. from: 0000001      Version: 0      P/N: FFFF

ID no. to: 0000001      Profile: 00

☒ **Change status**

☐ Employee present      ☐ Official absence authorisation

☐ Not checking 1st booking plausibility      ☐ Correction authorisation

☐ Not checking plausibility      ☐ Block master record

☐ **Short master record**      Mailboxtext no. 0

Display info 1 11111111      Display Info 5      Display Info 8

Display info 2 22222222      Display Info 6      Display Info 9

Display info 3 33333333      Display Info 7      Display Info 10

Display info 4 44444444      (Allowed characters for display info: 0-9 + - , . : )

Personal information

User defined text

Room no.

OK      Terminal...      Cancel

**Figure 3-7: Maintain master record**

Be sure that you did set one of the options fields **9290, Bedanet (B-Net) 9380 or Others** of the dialog **Master record selection**, so that you're able to create a correct master record for the desired terminal.

---

When you click the **OK** button, the newly created master record is transferred to the file after you have entered the data in the mask. You can list all logged on terminals by clicking the **Terminal** button. You can highlight a specific terminal with the cursor keys or by clicking it with the mouse. When you activate the **Display** button, the log-on data of the highlighted terminal is displayed. By clicking the **Accept** button, you transfer the highlighted terminal to the previous screen.

**Accept**

When you click the **Accept** button, the modified master record file is accepted.

**Cancel**

You can cancel the TRANSMIT MASTER RECORD FILE... menu item by selecting the **Cancel** button.

---

### 3.2.2. Autom. master record download

You can send an existing master record file by selecting the AUTOM. MASTER RECORD DOWNLOAD... menu item. Master records that are in the terminal, but which are not in the file, are deleted in the terminal (father-son procedure; see separate documentation entitled „Principles of data communication“ for further information), i.e. only those master records that are in the master record file are now located in the terminal. The master record file is sent **in blocks of 10 master records each. Booking is still possible at the terminals.** After a block has been sent, any pending data is fetched from the terminal.

The master record file must be stored as an **ASCII file**. Each master record in this file **must** contain the addressing information. The use of wildcards  $\Delta$  (7F<sub>hex</sub>) in the addressing information of the master records is permitted.

#### Example 1:

`@@ Y000000001000200FFFF00`

The master record used in example 1 can only be sent to the terminal with the group/device address 00/00.

#### Example 2:

`$\Delta\Delta$  Y000000001000200FFFF00`

The master record used in example 2 can be sent to all existing terminals.

On terminal log-on (see chapter 3.3.2 Terminal logon) a line no., which can be used as address alternatively to the GID/DID, can be assigned to each terminal. Instead of the group and device identification, the line no. can also be used in the address of master records. To transmit a master record to all existing terminals, line no. 000 must be entered.

#### Example 3:

`004 Y000000001000200FFFF00`

The master record used in example 3 can only be transmitted to the terminal with line no. 004.

#### Example 4:

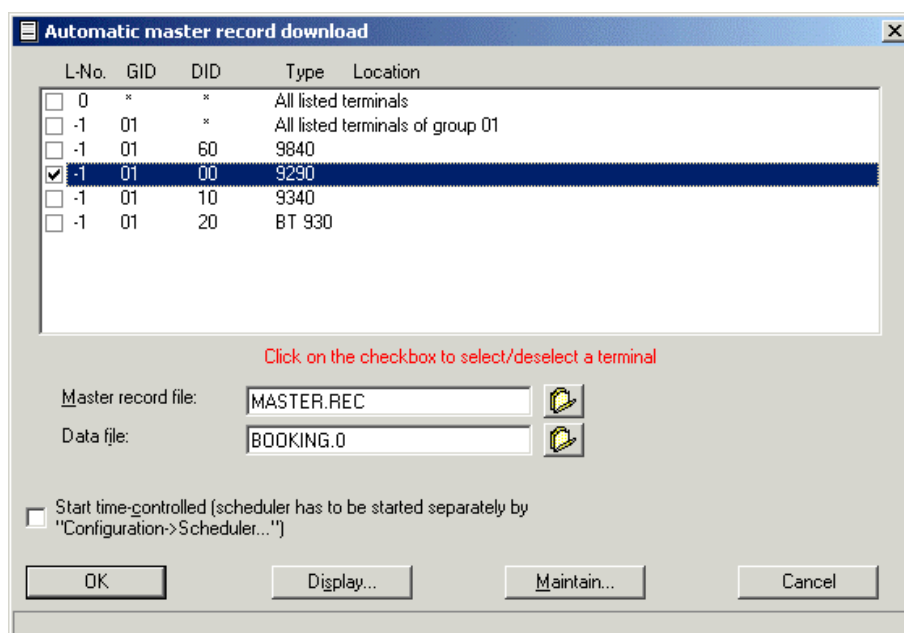
`000 Y000000001000200FFFF00`

The master record used in example 4 can be transmitted to all terminals existing.



The relationship between logical device addressing and the corresponding transferred ASCII characters is seen in chapter 3.2.1 Transmit master record file.

When you select the AUTOM. MASTER RECORD DOWNLOAD... menu item, the following screen is displayed:



**Figure 3-8: Automatic master record download**

When the FTP stream 3Ø is the active stream, in the terminal list all Bedanet 938Ø (B-Net 938Ø) terminals of all configured streams are displayed. In the terminal list you have to select the desired terminals by activating the check box at the beginning of each line of the list. To the selected terminal only those master records are sended, which contain the appropriate address.

Enter the file that you wish to send in the **Master record file** text box. The master record file must be in the path that you specified in the **Path for master records** box during installation (see chapter 3.3.8 Directories).

All data transferred from the terminal during execution of this function is stored in the file that you have specified in the **Data file** text box.

Shall the master record file transmit be time-controlled, activate the **Start time-controlled** check box (see chapter 3.3.15 Scheduler). This check box is not available, when the download should be done by BCOMMFTP.

---

**Available buttons:**

**OK**

Pressing the button **OK** stores the order in the (time-controlled) order file. This button is disabled until at least one terminal is selected.

**Display**

With the button **Display** you can view more detailed information about the selected terminal.

**Maintain**

By pressing the **Maintain** button, you can edit the master record file as described in chapter 3.2.1 Transmit master record file.

**Cancel**

You can cancel the AUTOM. MASTER RECORD DOWNLOAD... menu item by selecting the **Cancel** button.

---

### 3.2.2.1 Father-son-procedure

The father-son-procedure for the download of master records is approved in practical use and requires two files which must fulfill the following functions:

- **Father file**  
This file is filled with the master records to be loaded by an application program. It contains all data necessary to address the data records such as line number, group address and device address, in other words the **nominal state** of the terminal's master records.  
The father file **does not contain delete records**, only record type **YØ** exists. Even if several terminal lines and groups exist, it is recommended to use only one file for all terminals.
- **Son file**  
**Only the communication program** manages and controls the son file. This file shows the present state of the distribution and addressing of the terminal's master records, i.e. the terminal's **real state**.

These two files supply all the information necessary to manage the terminal's master records. Functions necessary herefore:

- Write new master records into the terminals
- Update existing master records
- Delete „old“ master records

---

How to realize these functions in the father-son-procedure:

The communication program reads the order file and the terminal configuration file from which it receives the addresses of the terminals considered for the download (GID/DID). After that, the first master record for the first terminal, e.g. terminal 01/02, in the father file is read. In the son file, it will be checked for all master records stored in the father file for this terminal if this master record already exists. If master records not existing in the father file are found in the son file, they must be deleted in the terminal for which a delete record Y1 is generated.

After these checks all Y0 and Y1 records and, finally, a Y9 record (end of master record download) are transmitted to the terminal.

The son file is then updated and the procedure continues with the next terminal of the group until all terminals are processed. After the download, the father file turns into the son file.

Example for the father-son-procedure:

in this example, three terminals to which master records must be transmitted are addressed in group 00. The master files are compared terminal by terminal.

Father file:

@0Y01000100000000FFFF00	101	102	103	104
@0Y010001010**00FFFF00	102	103	104	105
@0Y01000102000000FFFF00	103	104	105	106
@0Y010001030**00FFFF00	104	105	106	107
@0Y0100010402*00FFFF00	105	106	107	108
A0Y0100011700*00FFFF00	118	119	120	121
A0Y0100011800*00FFFF00	119	120	121	122
A0Y0100011900*00FFFF00	120	121	122	123
B0Y010001100**00FFFF00	111	112	113	114
B0Y01000111000000FFFF00	112	113	114	115
B0Y010001120**00FFFF00	113	114	115	116
B0Y010001130**00FFFF00	114	115	116	117

```
@@0Y01000101...
@@0Y01000103...
@@0Y01000104...
@A0Y01000117...
@A0Y01000118...
@A0Y01000119...
@A0Y01000125...
@A0Y01000126...
@B0Y01000109...
@B0Y01000110...
@B0Y01000112...
@B0Y01000113...
@B0Y010001345...
```

Badges number 101, 103 and 104 are already contained in the son file and will be updated. Badges number 100 and 102 are new. All master records are transmitted directly from the father file to the terminal.

@@0Y01000100000000FFFF00	101	102	103	104
@@0Y010001010*00FFFF00	102	103	104	105
@@0Y01000102000000FFFF00	103	104	105	106
@@0Y010001030*00FFFF00	104	105	106	107
@@0Y0100010402*00FFFF00	105	106	107	108
@@0Y9				

Badges number 117, 118 and 119 are contained in both files and loaded to the terminal unchanged. Badges number 125 and 126 are only contained in the son file and must be deleted in the terminal. The communication program generates delete record for these two badges.

@AØYØ100011700*ØØFFFFØØ	118	119	12Ø	121
@AØYØ100011800*ØØFFFFØØ	119	12Ø	121	122
@AØYØ100011900*ØØFFFFØØ	12Ø	121	122	123
@AØY11000125				
@AØY11000126				
@AØY9				

---

Effect for terminal Ø2:

Badges number 11Ø, 112 and 113 are contained in both files and loaded to the terminal unchanged. Badge number 111 is new and will also be loaded unchanged. Badges number 1Ø9 and 345 are only contained in the son file and must be deleted in the terminal. The communication program generates delete records for these two badges.

```
@BØY11ØØØ1Ø9
@BØYØ1ØØØ11ØØ**ØØFFFFØØ      111      112      113      114
@BØYØ1ØØØ111ØØØØØFFFFØØ      112      113      114      115
@BØYØ1ØØØ112Ø**ØØFFFFØØ      113      114      115      116
@BØYØ1ØØØ113Ø**ØØFFFFØØ      114      115      116      117
@BØY11ØØØ345
@BØY9
```

After the download of the master records, the son file is updated to the state of the father file and then contains the following entries:

```
@ØYØ1ØØØ1ØØ...
@ØYØ1ØØØ1Ø1...
@ØYØ1ØØØ1Ø2...
@ØYØ1ØØØ1Ø3...
@ØYØ1ØØØ1Ø4...
@ØYØ1ØØØ117...
@ØYØ1ØØØ118...
@ØYØ1ØØØ119...
@BØYØ1ØØØ11Ø...
@BØYØ1ØØØ111...
@BØYØ1ØØØ112...
@BØYØ1ØØØ113...
```

### 3.2.3. Master record upload

By selecting the menu item MASTER RECORD UPLOAD..., the master records of a terminal could be read out and stored in a file. When you select the MASTER RECORD UPLOAD... menu item, the following screen is displayed:

L-No.	GID	DID	Type	Location
<input checked="" type="checkbox"/> 0	*	*	All listed terminals	
<input checked="" type="checkbox"/> -1	01	*	All listed terminals of group 01	
<input checked="" type="checkbox"/> -1	01	60	9840	
<input checked="" type="checkbox"/> -1	01	00	9290	
<input checked="" type="checkbox"/> -1	01	10	9340	
<input checked="" type="checkbox"/> -1	01	20	BT 930	

Click on the checkbox to select/deselect a terminal

ID no.:

Master record file:

Data file:

☒ Wait until order finished      Timeout  sec.

☐ Terminal Server Client

☐ Start time-controlled (scheduler has to be started separately by "Configuration->Scheduler...")

OK      Display...      Cancel

Select if order should be started time-controlled

Figure 3-9: Master record upload

In the terminal list you have to select the desired terminals by activating the check box at the beginning of each line of the list.

In the drop-down combo box **ID no** you have to enter the 7-digit ID number of the master record to request from the terminal. When **xxxxxxx** is entered here, all master records are requested, when **9999999** is entered, only the pseudo master record is requested..

Enter the name of the target file into the text box **Master record file**. After the transfer, this file resides in the path that you specified in the **Path for master records** text box during installation (see chapter 3.3.8 Directories).

---

If more than one terminal is selected for upload,, you cannot enter a filename into this field. In this case for each terminal a separate file is created. The filename contains the GID/DID and the stream number of the terminal, e.g. MAS0001.2 for a master record file of terminal 00/01 on communication stream 2. If a file already exists it is overwritten without prompting.

All data transferred from the terminal during execution of this function is stored in the file that you have specified in the **Data file** text box.

When the dialog should remain visible until the upload has finished, select the **Wait until order finished** check box. The maximum wait time can be entered in the text box **Timeout**.

Shall the master record file transmit be time-controlled, activate the **Start time-controlled** check box (see chapter 3.3.15 Scheduler). This check box is not available, when the download should be done by BCOMMFTP.

**Available Buttons:**

**OK**

Pressing the button **OK** stores the order in the (time-controlled) order file. This button is disabled until at least one terminal is selected.

**Display**

With the button **Display** you can view more detailed information about the selected terminal.

**Cancel**

You can cancel the MASTER RECORD UPLOAD... menu item by selecting the **Cancel** button.



### 3.2.4. Request bookings

When communication is started, data is fetched from terminals in the Online mode. Data is also fetched from terminals in the Offline mode after communication started provided they are not connected to a modem.

When you select the REQUEST BOOKINGS... menu item, data can be fetched from terminals in the autonomous mode. Data can also be fetched from terminals in the Offline mode provided they are connected to a modem and data can be fetched from terminals if they are connected to a Bedas 98xx server.

When you select the REQUEST BOOKINGS... menu item, the following screen is displayed:

Request bookings

Line no. ....: -1    GID: 1\_    DID: 10

☐ Autonomous  
☒ Offline  
☐ Bedas 98xx

<input checked="" type="checkbox"/> File 1	<input checked="" type="checkbox"/> Clear file after transmission
<input type="checkbox"/> File 2	<input type="checkbox"/> " " " "
<input type="checkbox"/> File 3	<input type="checkbox"/> " " " "
<input type="checkbox"/> File 4	<input type="checkbox"/> " " " "
<input checked="" type="checkbox"/> File 5	<input checked="" type="checkbox"/> " " " "
<input type="checkbox"/> File 6	<input type="checkbox"/> " " " "
<input type="checkbox"/> File 7	<input type="checkbox"/> " " " "
<input type="checkbox"/> File 8	<input type="checkbox"/> " " " "

Data file: BOOKING.3

☐ Start time-controlled (scheduler has to be started separately by "Configuration->Scheduler...")

OK    Terminal...    Cancel

Name of the booking file

Figure 3-10: Request bookings

In this screen, complete the **GID** and **DID** text boxes or the **Line no.** text box to address the terminal (see chapter 3.3.2 Terminal login). Wildcards (\*) can be used in the **GID** and **DID** text boxes. If you want to address all terminals, enter a Ø in the **Line no.** text box.

---

You can use the cursor keys or you can click with the mouse to define whether data is to be fetched from terminals in the Autonomous or the Offline mode or from terminals that are connected to a Bedas 98xx server.

When fetching data from terminals that are connected to a Bedas 98xx server, in the **File 1** to **File 8** check boxes you can specify the memory file from which data is to be read and which data is to be deleted in the server after sending.

All data fetched from the terminals is stored in the file that you specify in the **Data file** text box.

Shall the data fetch be time-controlled, activate the **Start time-controlled** check box (see chapter 3.3.15 Scheduler).

**Example 1 of a booking file:**

```
A@0B1192012209490>>>2:35012489
A@0B219201220949000000005000033
```

The data has been fetched from the terminal with the group/device address 01/00.

Refer to 3.2.1 Transmit master record file for details of the relationship between the logical device address and the corresponding transmitted ASCII character.

If a line number has been assigned to a terminal on terminal log-on (see chapter 3.3.2 Terminal logon), the line number is used in the booking file.

**Example 2 of a booking file:**

```
0040B1192012209490>>>2:35012489
0040B219201220949000000005000033
```

The booking data has been fetched from the terminal with the line number 004.

**Available buttons:**

**OK**

By pressing the **OK** button, you can store the order in the (time-controlled) order file.

**Terminal**

When you press the **Terminal** button, all logged on terminals are listed. You can highlight a specific terminal with the cursor keys or by clicking it with the mouse. If you activate the **Display** button, the log-on data of the highlighted terminal is displayed. If you activate the **Accept** button, the highlighted terminal is transferred to the previous screen.

**<Cancel>**

You can cancel the REQUEST BOOKINGS... menu item by selecting the **Cancel** button.

### 3.2.5. Control buffer memory

This function is only active on the terminals on which „Buffer Backup“ is set. The buffer memory can be addressed with this function. The following screen is displayed when you select the CONTROL BUFFER MEMORY... menu item:

Figure 3-11: Control buffer memory

In this screen, complete the **GID** and **DID** text boxes or the **Line no.** text box for terminal addressing (see chapter 3.3.2 Terminal logon). The use of wildcards (\*) in the **GID** and **DID** text boxes is permitted. If you wish to address all terminals, enter a Ø in the **Line no.** text box.

In the next fields, you can select a control command with the cursor keys or by clicking with the mouse.

The **Get Data once more...** radio button reads the buffer memory.

Data that has been transferred can be deleted from the buffer memory by selecting the **Delete Data already got...** radio button.

Select the **Delete all data from buffer** radio button if you wish to clear the buffer memory.

If you intend to address the Bedas 98xx server, in the **File 1** to **File 8** check boxes you can specify the memory file from which the data is to be read. The **File 1** to **File 8** check boxes are not active until a Bedas 98xx server is

---

addressed in the **GID/DID** or **Line no.** text boxes with the aid of the **Terminal** button.

All data transferred from the terminal during execution of this function is stored in the file that you have specified in the **Data file** text box.

Shall the buffer memory control be time-controlled, activate the **Start time-controlled** check box (see chapter 3.3.15 Scheduler).

**Available buttons:**

**OK**

By pressing the **OK** button, you can store the order in the (time-controlled) order file.

**Terminal**

When you press the **Terminal** button, all logged on terminals are listed. You can highlight a specific terminal with the cursor keys or by clicking it with the mouse. If you activate the **Display** button, the log-on data of the highlighted terminal is displayed. If you activate the **Accept** button, the highlighted terminal is transferred to the previous screen.

**Cancel**

You can cancel the CONTROL BUFFER MEMORY... menu item by selecting the **Cancel** button.

### 3.2.6. Set date/time

You can use this function to send the date and the time to the terminals. The following screen is displayed when you select the SET DATE/TIME... menu item:

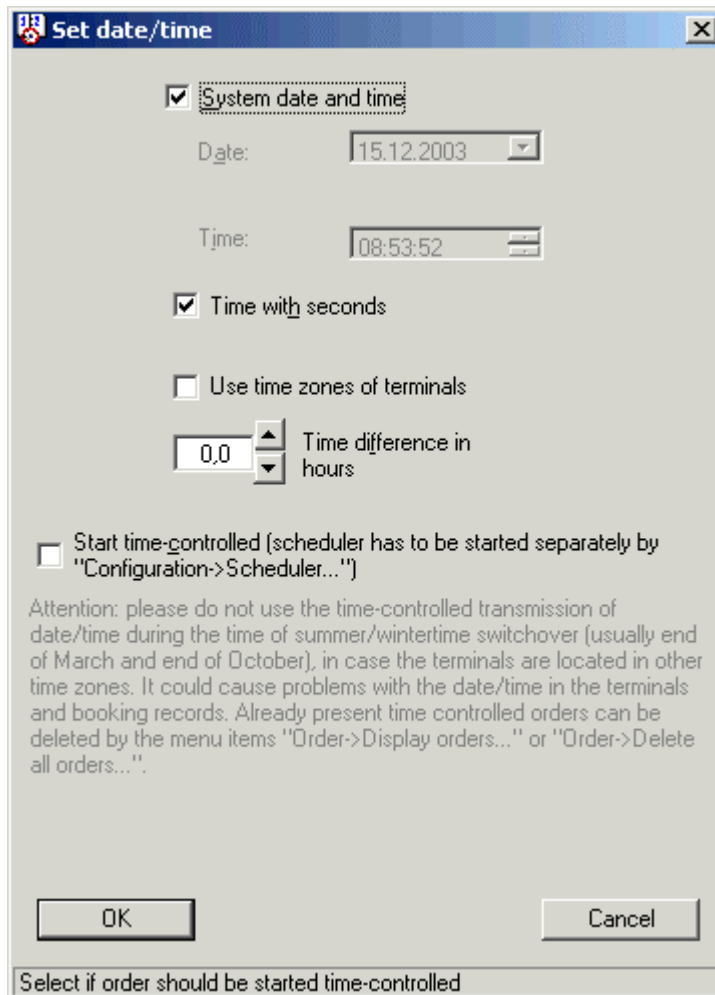


Figure 3-12: Set date/time

When the checkbox **Send time/date to each terminal with GID/DID, not with General Select** in the configuration dialog of the stream (see chapter 3.3.2 Terminal logon) is set, for each terminal a separate data record is sent, otherwise only one record with General Select is sent to all terminals.

Exception: when you decide to **Use time zones of the terminals**, there are always sent separate records for each terminal, because on one stream there could be terminals in different time zones.

When you don't use the time zones of the terminals, you can set a **time difference in hours**, which could be between -12 and + 13 hours.

---

The system date and time of the host are used when you activate the **System date and time** check box. Then the date and time in the moment of the order execution is used.

You can enter a transmit date and a transmit time if you do not activate the **System date and time** check box. The input formats depend on the settings made in CONTROL PANEL – REGIONAL SETTINGS.

If you activate the **Time with seconds** check box, seconds values are also transferred to the terminals.

Shall the set of the date/time be time-controlled, activate the **Start time-controlled** check box (see chapter 3.3.15 Scheduler).

**Available buttons:**

**OK**

By pressing the **OK** button, you can store the order in the (time-controlled) order file.

**Cancel**

You can cancel the SET DATE/TIME... menu item by selecting the **Cancel** button.

### 3.2.7. Transmit display record

You can send display records to the terminals.

The following screen is displayed when you select the TRANSMIT DISPLAY RECORD... menu item:

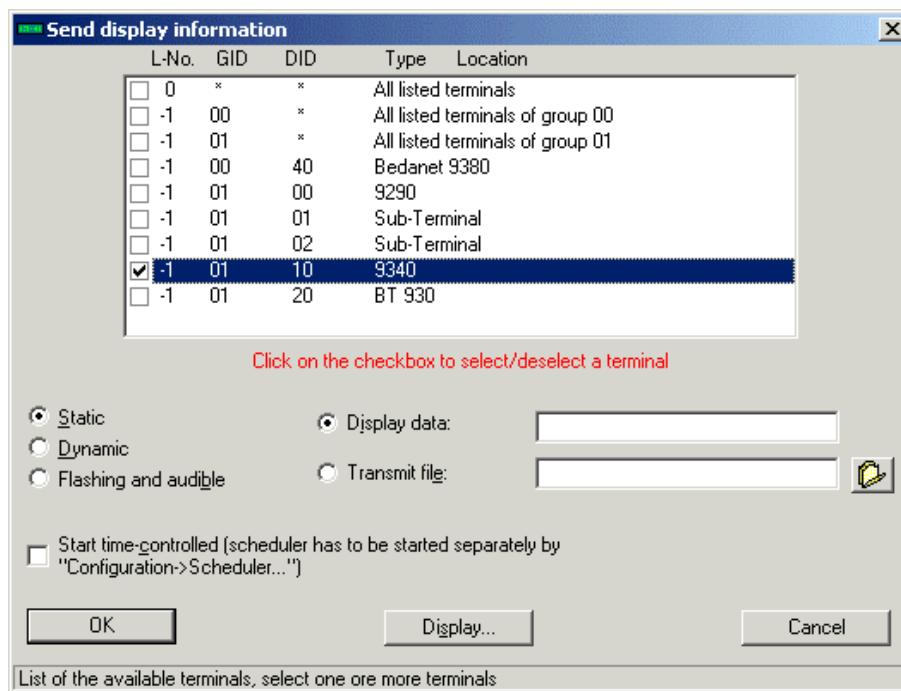


Figure 3-13: Transmit display record

In the terminal list you have to select the desired terminals by activating the check box at the beginning of each line of the list.

If the **Transmit file** radio button is activated, you can enter a filename into the appropriate text box. This file must be stored as a **ASCII file**. It is sent to the terminal in records. **Each record must be terminated with CR/LF**. Each display record in this file should begin with 3 blanks and should contain the record type and the record type modification. The terminal address is entered in the first three positions when sending the file. Addressing information in the records is ignored.

**Example of a transmit file:**

```
D2hallo
D2 hallo
D2   hallo
D2    hallo
D2     hallo
D2      hallo
D2       hallo
D2        hallo
D2         hallo
D2          hallo
```

**The file must be in the path that you specified in the “Path for user data” box during installation.**

If the **Display data** radio button is activated, you can enter any text in the appropriate text box that is to be sent to the terminal display.

You can select the mode of display (**Static**, **Dynamic** or **Flashing and audible**) by clicking with the mouse.

Shall the display record transmit be time-controlled, activate the **Start time-controlled** check box (see chapter 3.3.15 Scheduler).

**Available buttons:**

OK

By pressing the **OK** button, you can store the order in the (time-controlled) order file. This button is disabled until at least one terminal is selected.

**Display**

With the button **Display** you can view more detailed information about the selected terminal.

**Cancel**

**Cancel** You can cancel the TRANSMIT DISPLAY RECORD... menu item by selecting the **Cancel** button.



### 3.2.8. Transmit transparent data

By selecting this menu item, you can transmit any chosen Kaba Benzing records from a file to the terminals. The following screen is displayed when you select TRANSMIT TRANSPARENT DATA... menu item:

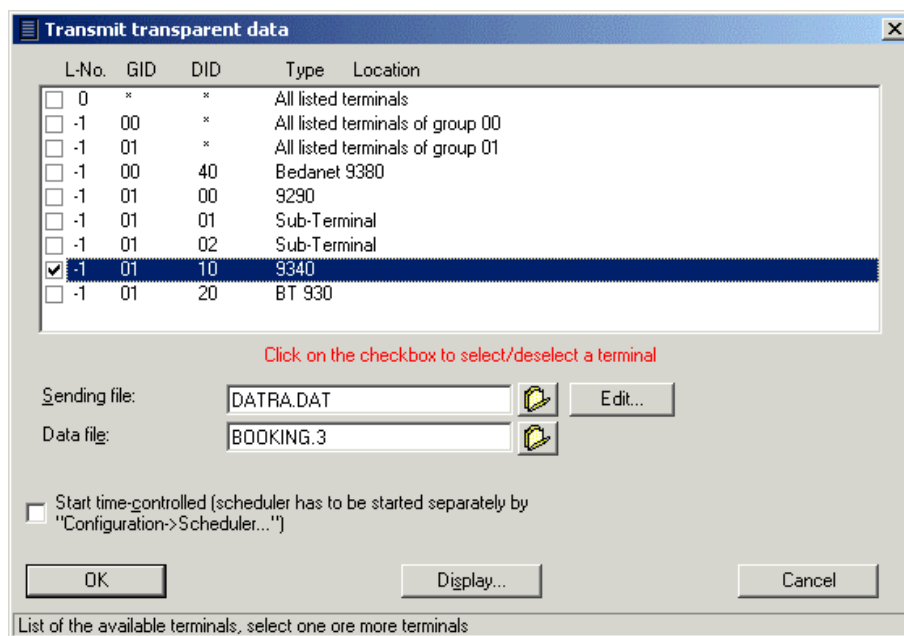


Figure 3-14: Transmit transparent data

When the FTP stream 30 is the active stream, all Bedanet 9380 (B-Net 9380) terminals of all configured streams are displayed in the terminal list. In the terminal list you have to select the desired terminals by activating the check box at the beginning of each line of the list.

In the **Sending file** text box, enter the file containing the records. This file must be stored as an **ASCII file**. **Each record must be terminated with CR/LF**. The file is sent **in blocks of 10 records each**. After one block has been send, any pending data is fetched from the terminal. The contents of the file are not checked. Each record **must** begin with 3 blanks. When the file is sent, the terminal address is entered in the first three positions. Addressing information in the records is ignored.

---

Example of a transmit file:

```
D2Master memory init
Y1*****
D2End
```

**The file must be in the path that you specified in the „Path for User Data“ box during installation.**

All data transferred from the terminal during execution of this function is stored in the file that you specify in the **Data file** text box.

Shall the transparent data transmit be time-controlled, activate the **Start time-controlled** check box (see chapter 3.3.15 Scheduler).

**Available buttons:**

**OK**

By pressing the **OK** button, you can store the order in the (time-controlled) order file. This button is disabled until at least one terminal is selected.

**Display**

With the button **Display** you can view more detailed information about the selected terminal.

**Cancel**

You can cancel the TRANSMIT TRANSPARENT DATA... menu item by selecting the **Cancel** button.

**Attention: as soon as all data records have been transmitted, the order is marked as finished. That means, that the I/O kernel BCOMMIO does not interpret the records. If there is a data record in the file, which causes a terminal to send data records, the order is marked finished before the terminal has sent these records.**

With this function, you can send a date to the terminals for summer/winter time switchover. The following screen is displayed when you select the SUMMER/WINTERTIME SETUP... menu item:

**Figure 3-15: Summer/Wintertime setup**

In the field **Date** you have to enter the date for the switchover. The input format depends on the settings made in CONTROL PANEL – REGIONAL SETTINGS. If you specify a date from the first half of the year as the changeover date, the terminal changes automatically from winter time to summer time on the day concerned. It switches from 2:00 h to 3:00 h in the morning. If you specify a date from the second half of the year, the terminal changes automatically from summer time to winter time on the date concerned. It changes over to 3:00 h at 2:00 h in the morning.

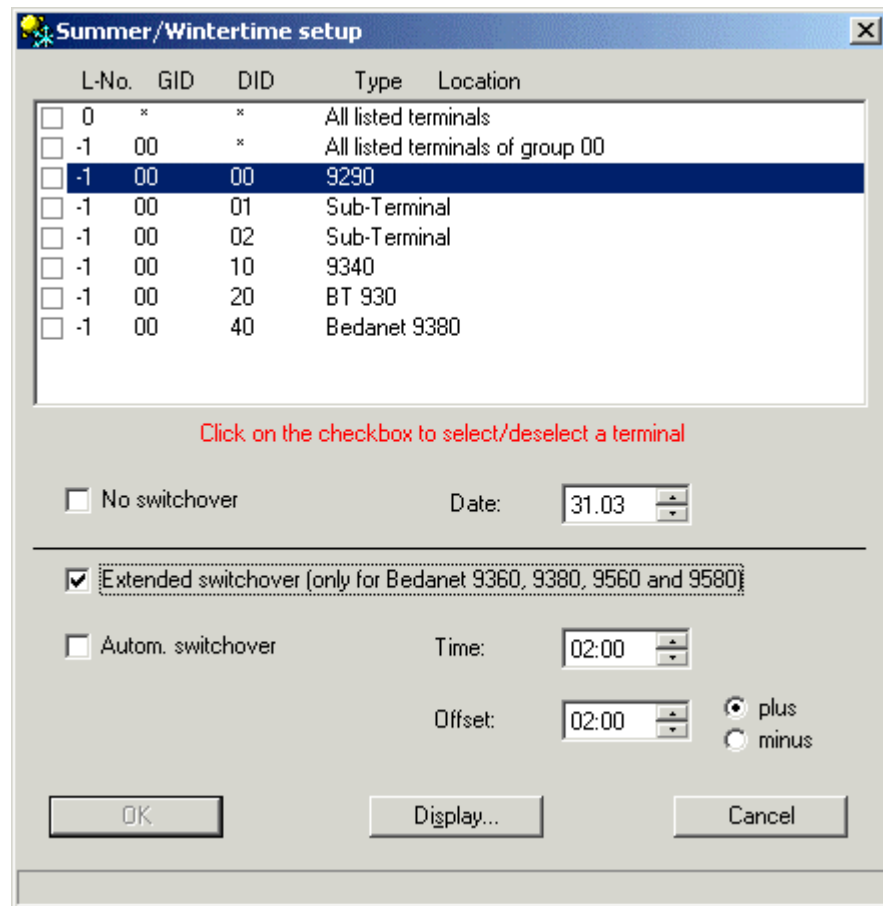
When you activate the check box **No switchover**, the Summer/Wintertime switchover in the terminal is switched off.

For terminals of type Bedanet 936Ø, 938Ø and 956Ø an extended swithover possibility exists. To use it, activate the checkbox **Extended swithover**. Now you can choose whether you want to activate the **Automatic swithover** or if you want to enter the **Time** and the **Offset** manually

When the **Automatic swithover** is activated the swithover is done by the terminal like described above.

When the automatic swithover is not activated, you can set the exact time of the swithover by entering the appropriate values in the fields **Time** and **Offset**.

An example: if you want to let the swithover take place on 31st of March, and the clock should switch from 2:ØØ h in the morning to 4:ØØ in the morning, the values should be set as in the following screenshot:



**Figure 3-16: Example for Summer/Wintertime setup**

---

**Available buttons:**

**OK**

By pressing the **OK** button, you can store the order in the order file. This button is disabled until at least one terminal is selected.

**Display**

With the button **Display** you can view more detailed information about the selected terminal.

**Cancel**

You can cancel the SUMMER/WINTERTIME SETUP... menu item by selecting the **Cancel** button.

### 3.2.10. Seconds/User language in data record

With this function you can specify whether the time in data records should be sended with seconds and whether the user language should be sended or not. The following screen is displayed when you select the SECONDS/USER LANGUAGE IN DATA RECORDS ... menu item:

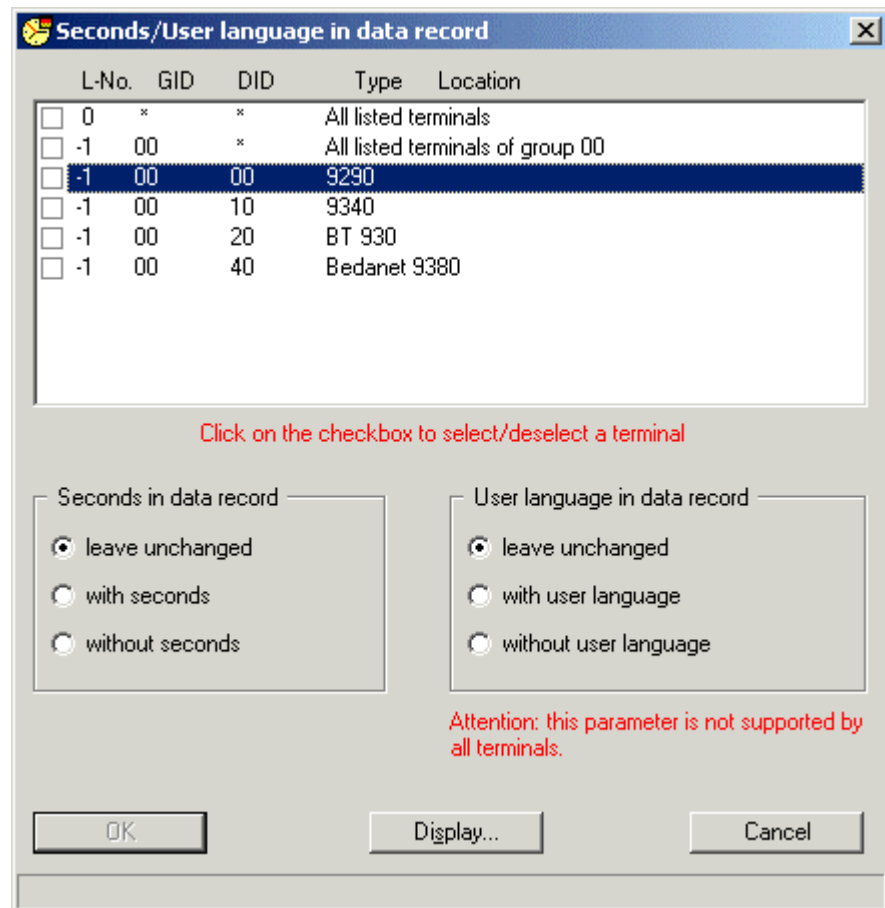


Figure 3-17: Seconds/User language in data record

In the terminal list you have to select the desired terminals by activating the check box at the beginning of each line of the list.

You can – separately for seconds and user language – leave a feature unchanged or switch the feature on or off by selecting the appropriate radio button.

**Attention: the parameter **User language in data record** is only supported by the newest terminal generation.**

---

**Available buttons:**

**OK**

By pressing the **OK** button, you can store the order in the order file. This button is disabled until at least one terminal is selected.

**Display**

With the button **Display** you can view more detailed information about the selected terminal.

**Cancel**

You can cancel the SECONDS/USER LANGUAGE IN DATA RECORD... menu item by selecting the **Cancel** button.

### 3.3. Configuration

When you select the CONFIGURATION menu, you can choose from the CLIENT MANAGEMENT..., TERMINAL LOGON..., UPDATE TERMINAL PROGRAM NUMBER..., SET OPERATING MODE..., ADD NEW TERMINAL TYPE..., INITIALIZE COMMUNICATION STREAM..., START OPTIONS..., DIRECTORIES..., PRINTER LOG..., FILE DEFINITIONS..., FTP TRANSFER CONFIGURATION..., CHANGE PASSWORD..., BACKUP..., RESTORE..., SCHEDULER..., SERVICES..., DEUTSCH/ENGLISH... and SET PC DATE/TIME... menu items.

**For the FTP stream 30 the menu items TERMINAL LOGON..., UPDATE TERMINAL PROGRAM NUMBER..., SET OPERATING MODE..., INITIALIZE COMMUNICATION STREAM... and SCHEDULER... are not available.**

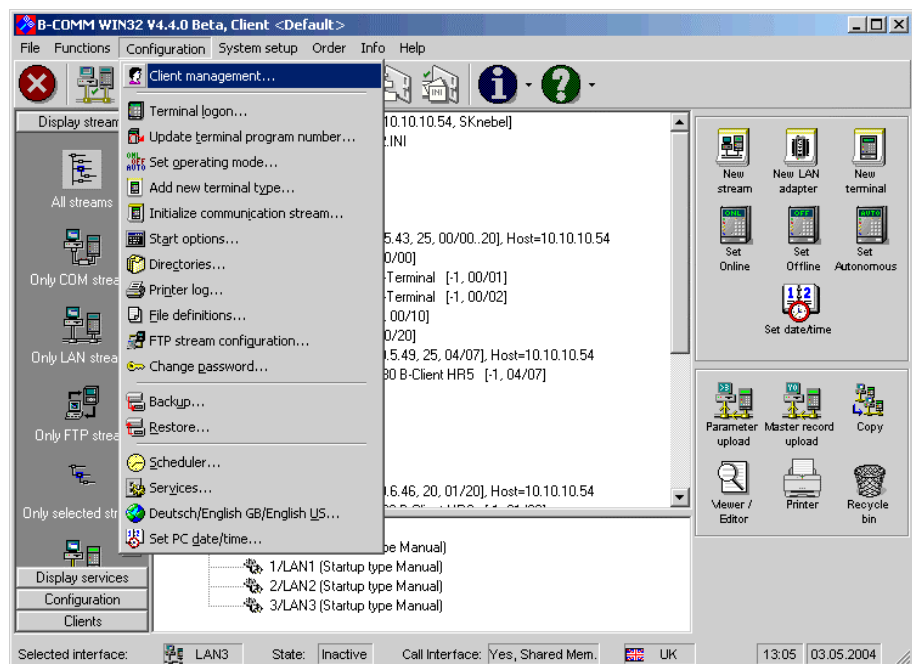


Figure 3-18: Menu Configuration



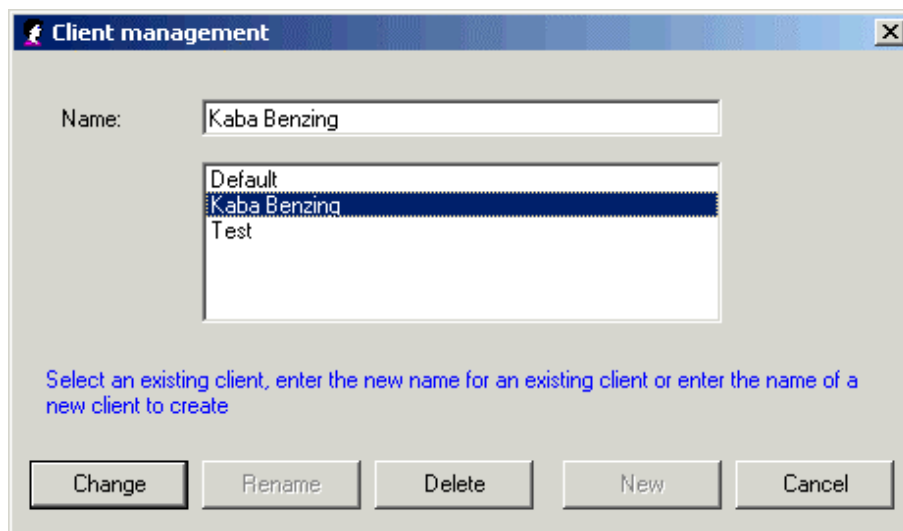
### 3.3.1. Client management

With this menu item, which is only selectable if the option "Multi-client capability" is licenced, you can create new clients, rename or delete existing clients or switch the active client.

**Attention: if this option should be used, the file BCWIN.DLL must be moved from the \Windows(WinNT)\System32 directory to the B-COMM installation directory before the first client is created! Also the configuration file BCOMM32.INI must be moved from the \Windows(WinNT) directory to the B-COMM installation directory and the entry "INIPath" in the main configuration file BCWIN.INI has to be adjusted to point to the B-COMM directory.**

A client called "Default" always exists and it can neither be deleted nor renamed. For each additional client a subdirectory below the installation directory of B-COMM WIN32 will be created, which has the name of the client. In this directory an own configuration file BCOMM32.INI for each client will be created, which only contains the configuration of that client. Also a new directory structure for master record files, parameter files etc. is created below the clients' directory.

The tree structure in the main window of B-COMM WIN32 always displays the configuration of the currently active client, the streams, network adapters, terminals and files of the other clients are not visible.



**Figure 3-19: Client management**

When a client is selected and this client is not the currently active client, you can set the selected client to active by the button **Change**.

If not the Default client is selected you can change the name of the selected client in the text field **Name** and then press the button **Rename** to change the name of the selected client.

Additionally the selected client can be deleted with the button **Delete**.

---

If a new name was entered in the text field **Name**, you can create the new client and its directory structure by pressing the button **New**.

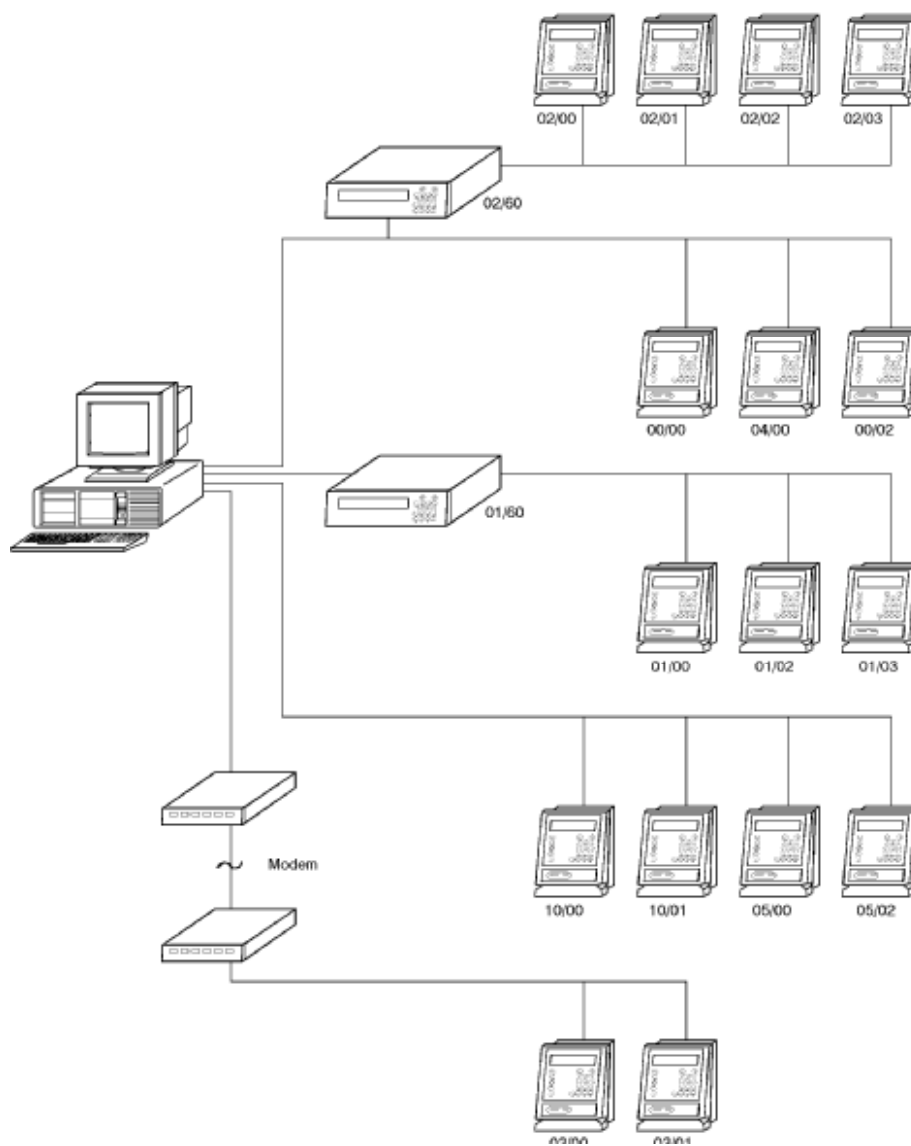
For each client the whole amount of streams (30) is available, all streams can be active simultaneously. A restriction exists concerning serial interfaces, i.e. two or more clients can't share the same serial interface. With LAN streams there is no restriction regarding the interfaces (LAN1 to LAN30), as long as all streams use different port numbers, i.e. one or more clients can have the stream 1/LAN1 active if the streams are configured for different port numbers.

### 3.3.2. Terminal logon

To enable BCOMMIO to address a terminal, this terminal must be logged on. After logon, a terminal is included in the configuration. The terminal configuration file is updated after logon (see chapter 7.1 Terminal configuration file TERMDEF.<Stream>).

BCOMMIO supports up to 60 terminals in each group. A group address from 00 to 29 can be assigned. 00 to 59 can be assigned as the device address. One group address is not limited to one communication stream. **The combination of a group and device address has to be unique, which means, it is only allowed once within the terminal system.** Up to 30 communication streams are supported. Up to 30 groups can be assigned to one communication stream. One terminal group can be addressed via a server, where the server has device address 60. Terminals that are connected to one server **must** have the same group address. **If more than one Access Manager Bedas 9290 are connected to a server, be sure that the 9290s are addressed in steps of 10, which means that the first 9290 has e.g. address 00/00, the second the address 00/10 and so on.** The same is true for Access Managers connected to a LAN adapter which has the **9290 Support** flag set.

The following figure shows these relationships:



**Figure 3-20: Example of a terminal configuration**

When you select the **TERMINAL LOGON...** menu item and enter the password, the following screen is displayed:

**Terminal logon**

GID: 00 DID: 20 LAN adapter: 10.10.5.44

**Modem**

- ☒ No modem
- ☐ AT modem

**Operating mode**

- ☐ Offline
- ☐ Autonomous
- ☒ Online
- ☐ Unchanged

**Server**

- ☒ No server
- ☐ Server
- ☐ Bedas 98xx
- ☐ via Server
- ☐ via Bedas 98xx
- ☐ Subterminal

**Automatic logon of subterminals**

- ☐ No subterminals (Bedanet 9360)
- ☐ 4 subterminals (Bedanet 9360)
- ☒ 8 subterminals (Bedas/Bedanet 9290)
- ☐ 16 subterminals (Bedanet 9290)

Call number:

Location:

Data file: BOOKING.2

☒ Logon a single terminal

☒ Set date/time

Logon Exit

**Figure 3-21: Terminal logon**

The logical address of the terminal you wish to logon must be entered in the **GID** and **DID** text boxes. When terminals on a LAN stream should be logged on, these fields are filled automatically with the first terminal address, which is given by the configuration of the selected LAN adapter (see chapter 3.3.6 Initialize communication stream).

In the combo box **LAN adapter** you can select the LAN adapter to which the terminals are connected to. This is only true if the selected stream is a LAN stream and more than one LAN adapter is configured for that stream.

---

If the terminal you wish to logon is connected to a modem, you must specify this in the **Modem** radio buttons. Additionally you must enter the phone number in the **Call number** text box. Pulse dialing or inband signaling must also be entered.

You can set the required operating mode (Online, Offline or Autonomous) of the terminal you wish to logon in the **Operating mode** radio buttons.

If you select **Unchanged**, then at first the current operating mode is requested and that mode is not changed during the logon. If there are bookings in the terminal at that time, it could happen, that a wrong operating mode is displayed (e.g. Offline instead of Online), but the mode remains unchanged.

You must activate the **Server** (for a TS986 or BITS) or **Bedas 98xx** radio buttons if you wish to logon a server. Then all terminals that are connected to the server are analyzed too. **Attention: if more than one Bedas 9290 is connected to the server, then the 9290s must be addressed in steps of 10, e.g. 00/00 for the first 9290, 00/10 for the second 9290 and so on.**

The following guidelines are to be followed for some terminals:

- for a Bedas 9290 the DID has to be an address dividable by 10 (00, 10, 20, 30, 40 or 50)
- for a Bedanet 9290 with a maximum of 16 subterminals the DID has to be dividable by 20 (00, 20 or 40). If the Bedanet 9290 should server as a replacement for a Bedas 9290, i.e. a maximum of 8 subterminals should be connected, then the DID could be dividable by 10 (00, 10, 20, 30, 40 or 50)
- for a Bedanet 9360 with subterminals the DID must be dividable by 5 (00, 05, 10, 15 etc.). A Bedanet 9360 without subterminals can have any valid DID from 00 to 59.

If a terminal should be logged on that is connected to a server and you don't wish to logon the server and all connected terminals again, select the radio button **via Server** or **via Bedas 98xx**.

In case of a LAN stream usually all terminals connected to the selected LAN adapter are logged on. If you just want to logon a single terminal (e.g. which is installed additionally), activate the check box **Logon a single terminal**. **Attention: it is not possible to logon an additional subterminal (connected to a 9290). In this case, the 9290 itself has to be logged on (automatically including all connected subterminals).**

If a control unit Bedas 9290, Bedanet 9290 or Bedanet 9360 is logged-on, all subterminals are analyzed which are connected to the access control unit (independent of the option **Logon a single terminal**). In case of a COM stream and the logon of a Bedanet 9290 or Bedanet 9360 you'll have to specify how many subterminals to logon automatically (no or 4 subterminals for a Bedanet 9360, 8 or 16 subterminals for a Bedanet 9290).

All data that is transferred from the terminal during execution of this function is stored in the file that you specify in the **Data file** text box.

---

If the check box **Set date/time** is activated the current date and time is sent to the terminal after its logon procedure, depending on the timezone that was set for the terminal.

**Available buttons:**

**Logon**

You can start the TERMINAL LOGON... menu item by selecting the **Logon** button. The "Analyzing terminal" message appears. If the terminal cannot be addressed, the „Terminal could not be logged on“ message appears.

**Exit**

You can exit the TERMINAL LOGON... menu item by selecting the **Exit** button.

When you logon a Bedas 98xx server, you must also specify the memory file out of which data is read and which data is to be deleted in the server after sending. The following screen is displayed for this purpose:

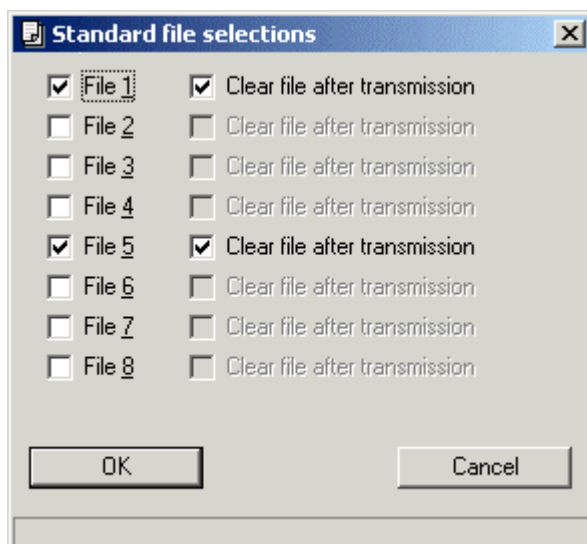


Figure 3-22: File selection

The following screen is displayed if it was possible to address the terminal to be logged on:

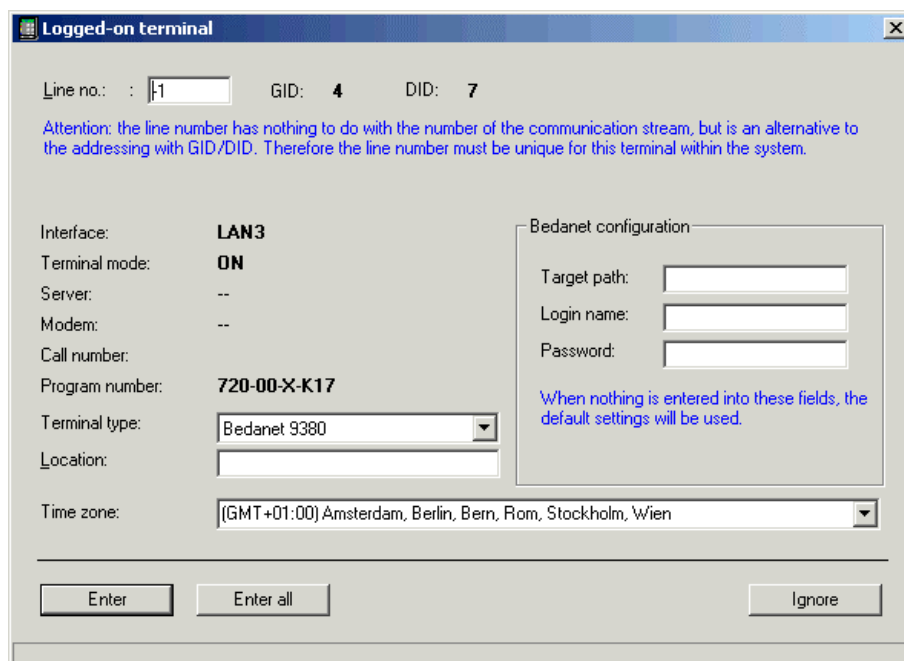


Figure 3-23: Logged-on terminal

---

By clicking the button **Enter**, the terminal information is written into the terminal configuration file.

When you select the **Enter all** button, all found terminals are written into the terminal configuration file at once, pressing **Ignore** ignores the terminal and it's not entered into the terminal configuration file.

By use of the **Terminal type** selection box you can change the terminal type if there exist more than one terminal with the same program number.

With the **Time zone** selection box you can assign a time zone to the terminal. This time zone is automatically taken into account if you send date/time to the terminal. That allows you to operate multiple terminals at one stream which are located in different time zones.

When logging on a terminal, each terminal can be assigned a so-called line number (**Line No.**), which can be used for addressing information as an alternative to GID/DID. A line number between 1 and 999, by which it can be addressed, can be assigned to one terminal. A line number between 1 and 999 must only be assigned once. If both GID/DID and the line number are used in addressing information, the line number has the higher priority. You must enter the required line number in the corresponding text box. GID, DID, the Port, the Terminal mode, Server, Modem, Call number, Terminal type and the Program number are also displayed. The **Location** text box merely serves to facilitate orientation and need not necessarily be filled with information. The terminal data is included in the configuration when you select the **OK** button.

In the text boxes **Target path**, **Login name** and **Password** those data could be entered, that are used by the I/O kernel BCOMMFTP to login into a Bedanet 938Ø (B-Net 938Ø) terminal. If you don't make any entries into these text boxes, BCOMMFTP uses default settings (see chapter 3.3.11 FTP Transfer configuration). The text boxes are only available if a Bedanet 938Ø (B-Net 938Ø) was logged on.



### 3.3.3. Update terminal program number

The program number of logged on terminals can be updated and written into the Terminal configuration file TERMDEF.<Stream>. This is especially interesting for terminals, whose programs can be updated via Internet. The following screen is displayed when you select the UPDATE TERMINAL PROGRAM NUMBER... menu item:

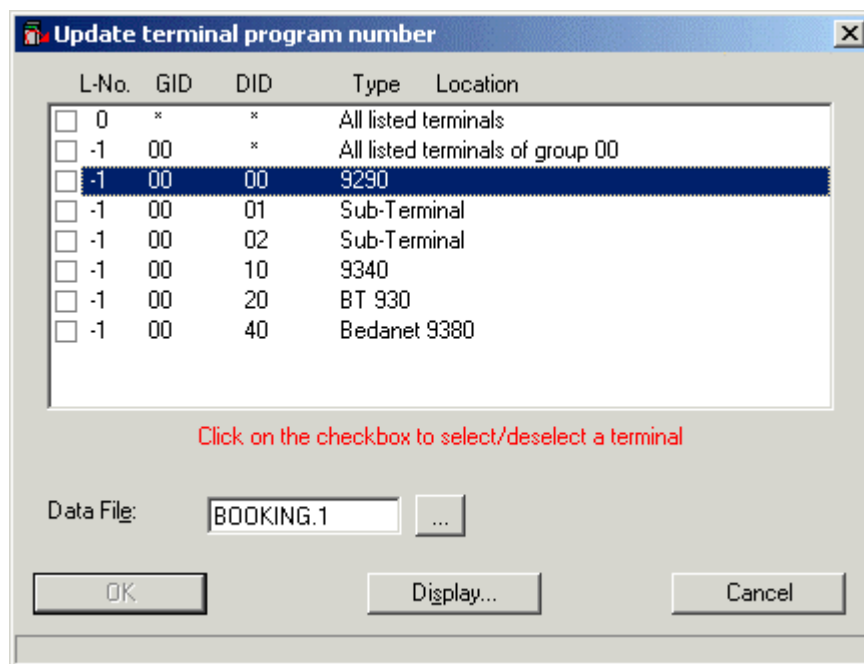


Figure 3-24: Update terminal program number

In the terminal list you have to select the desired terminals by activating the check box at the beginning of each line of the list.

All data transferred from the terminal during execution of this function is stored in the file that you have specified in the **Data file** text box.

#### Available buttons:

##### **OK**

By pressing the **OK** button, you can store the order in the order file. This button is disabled until at least one terminal is selected.

##### **Display**

With the button **Display** you can view more detailed information about the selected terminal.

##### **Cancel**

You can cancel the UPDATE TERMINAL PROGRAM NUMBER... menu item by selecting the **Cancel** button.

### 3.3.4. Set operating mode

Logged on terminals can be set to a required operating mode. Once you have set a terminal to a required operating mode, the change is stored in the terminal configuration file. The following screen is displayed when you select the SET OPERATION MODE... menu item:

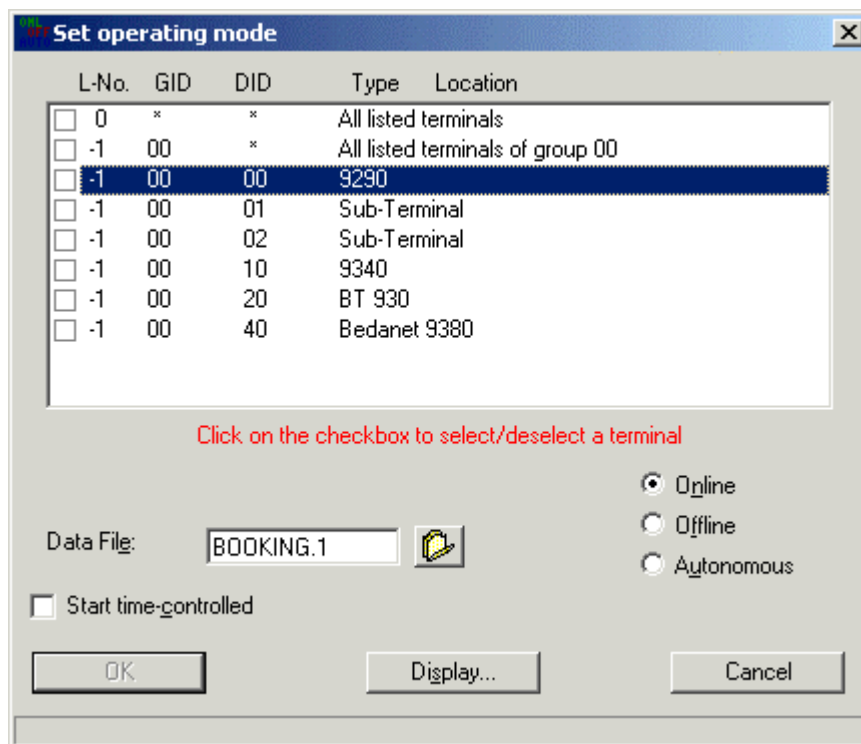


Figure 3-25: Set operating mode

In the terminal list you have to select the desired terminals by activating the check box at the beginning of each line of the list.

You can select the operating mode with the cursor keys or by clicking with the mouse.

All data transferred from the terminal during execution of this function is stored in the file that you have specified in the **Data file** text box.

Shall the operation mode set be time-controlled, activate the **Start time-controlled** check box (see chapter 3.3.15 Scheduler).

#### Available buttons:

##### **OK**

By pressing the **OK** button, you can store the order in the (time-controlled) order file. This button is disabled until at least one terminal is selected.

---

**Display**

With the button **Display** you can view more detailed information about the selected terminal.

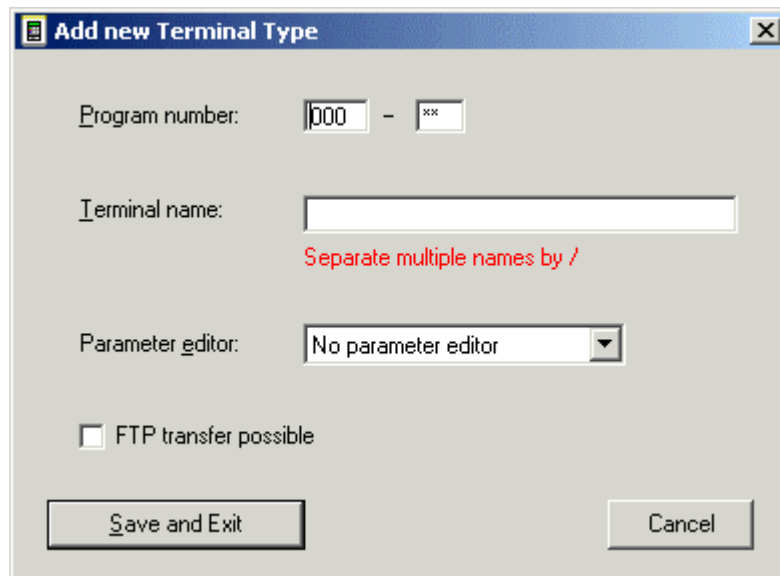
**Cancel**

You can cancel the SET OPERATION MODE... menu item by selecting the **Cancel** button.

---

### 3.3.5. Add new terminal type

When Kaba Benzing releases a new terminal, you can insert that terminal into B-COMM WIN32 by use of this menu item. The following screen is displayed when you select the ADD NEW TERMINAL TYPE ... menu item:



**Figure 3-26: Add new terminal type**

Enter the 3-digit program number in the **Program number** field.

In the text field **Terminal name** you can enter a name for the new terminal. If more than one terminal type is possible for the given program number, please separate the names by /.

By use of the combo box **Parameter editor** you can select the appropriate parameter editor to edit the parameter records of that terminal.

If FTP transfer is possible for the new terminal, click the appropriate checkbox **FTP transfer possible**. Only then the terminal is entered in the file FTP.TXT and will appear under FTP stream 30.

All entries you make in this dialog are automatically entered into the appropriate configuration files. These files are: TERMTYP.TXT, BCOMM32.INI and FTP.TXT.

---

### 3.3.6. Initialize communication stream

You can define, configure or delete the streams. When no communication stream has been defined until now, the following screen appears first:

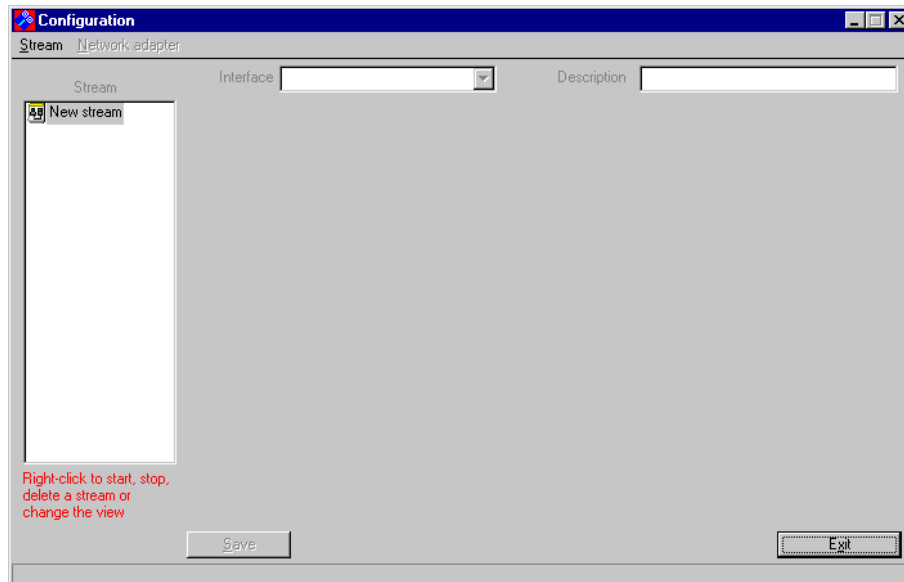


Figure 3-27: New stream

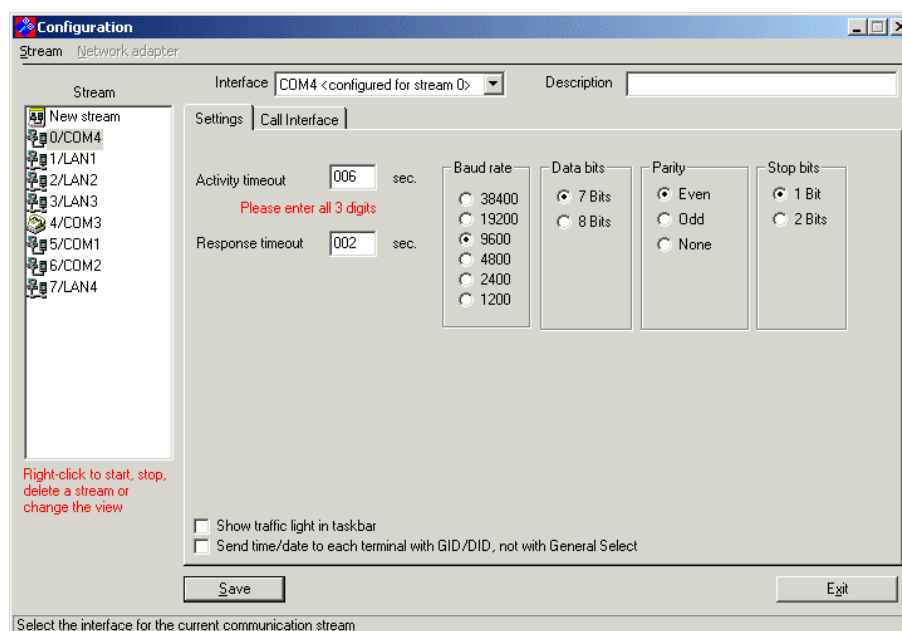
First a new communication stream must be created via a doubleclick on the **New stream** icon or by use of the STREAM – NEW STREAM... menu. To do so, the following screen appears:



Figure 3-28: Add communication stream

In this screen a communication stream (0 to 29) must be entered.

After entering the stream and selecting either **Serial (COMx)** or **Network (LANx)** the new stream can be created by pressing the **OK** button. After that (when a serial stream was created) the following screen mask appears:








**Figure 3-29: Serial stream**

An interface must be selected in the **Interface** combo box (COM1 to COM8 resp. LAN1 to LAN3Ø). The following entries are possible:

- **COM<x>/LAN<x>**: the interface COM<x>/LAN<x> is physically available, it is not assigned to a stream and currently not occupied by any program.
- **COM<x>/LAN<x> <in use by stream <y>**: the interface COM<x>/LAN<x> is physically available, it is assigned to stream <y> and the stream is currently active. In this case no other interface can be assigned to the stream.
- **COM<x> <in use>**: the interface COM<x> is physically available but currently in use by another program than B-COMM
- **COM<x>/LAN<x> <configured for stream <y>**: the interface COM<x>/LAN<x> is physically available, it is assigned to stream <y> and the stream is currently not active
- **COM<x> <not available>**: there is no interface COM<x> in the PC

In the field **Description** you can enter an optional description of the stream for a better identification.

Additionally you can specify whether a little **traffic light** symbol should be displayed on the task bar. This traffic light shows the state of BCOMMIO:

-  during initialization
-  initialization finished and ready, no data transfer
-  data transfer
-  no data transfer, min. one terminal not available
-  data transfer, min. one terminal not available

### Setting a serial interface (COM1 to COM8)

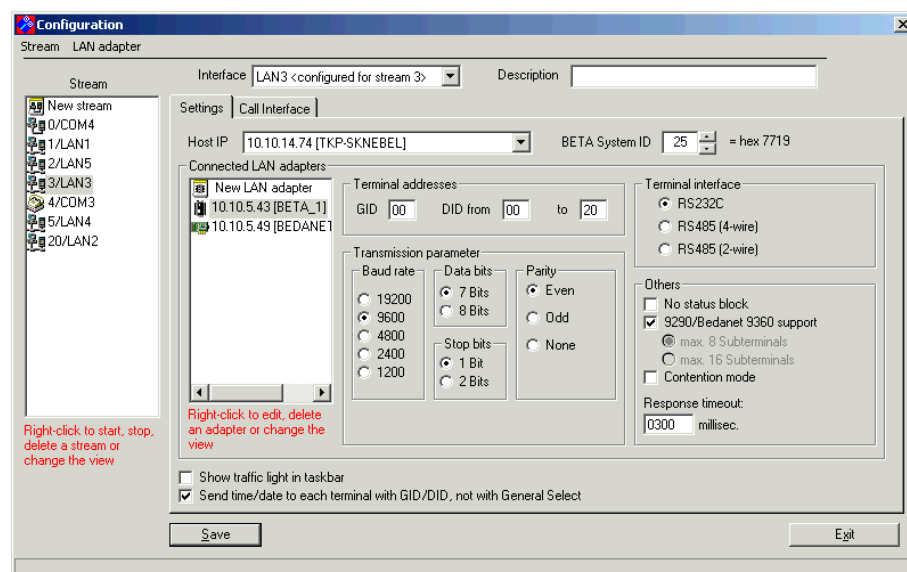
The upper screen displays the tab **Settings** for setting a serial interface.

As a default the following values are set:

**Activity Timeout** 6 seconds, **Response Timeout** 2 seconds, **Baud rate** 9600, **7 Data bits**, even Parity, **1 Stop bit**.

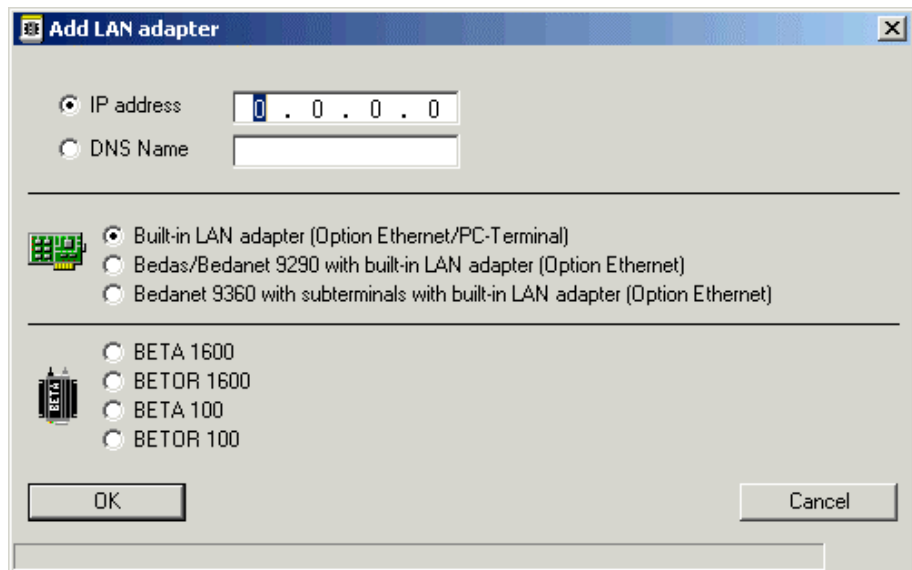
### Setting a network interface (LAN1 to LAN30)

The following screen displays the tab **Settings** for setting a network interface.



**Figure 3-30: Network stream**

By double-clicking the **New adapter** icon a network adapter can be entered. To do so, in the following screen the IP address or the DNS name as well as the adapter type must be entered.



**Figure 3-31: Add network adapter**

In the drop-down combo box **Host IP** the IP address of the local computer must be entered. BCWIN detects automatically the IP addresses of all network cards which are built in the computer.

In the text box **BETA System ID** the last two digits of the UDP port number must be entered in decimal values. The port number must correspond with the port numbers which shall be configured in the LAN adapters.

In the **GID** and **DID** text boxes the corresponding address must be entered. It must be observed, that the addresses entered in the text boxes **DID from** and **to** must exist continuously.

With the radio buttons for **Baud rate**, **Data bits**, **Parity** and **Stop bits** the corresponding transmission parameters must be entered.

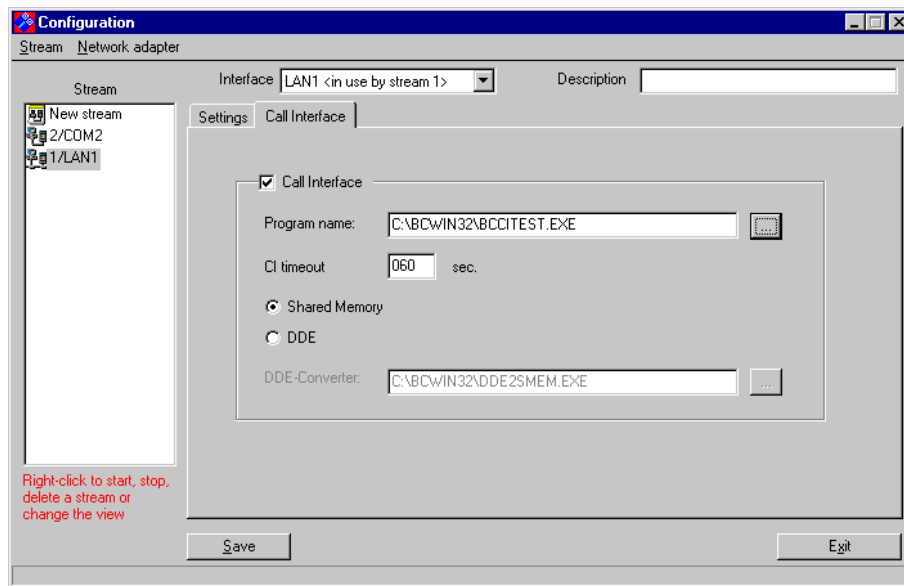
With the radio buttons **Terminal interface** the interface for the connection between the LAN adapters and the terminals must be selected.

With the check boxes **Others** you have to select whether **Contention mode**, **No status block** or **9290 support** is to be used. The **No status block** is necessary for networking via ISDN dial connections.

In the text box **Response time-out** the response timeout is entered in milliseconds.

In the register tab **Call Interface** it must be entered whether the data transmission kernel BCOMMIO shall start a so-called Call Interface. A Call Interface is a program which is started by the data transmission kernel BCOMMIO. **All** data records are transmitted to the Call Interface. The Call Interface is also transmitting data records to the data transmission kernel. The kernel transmits the data records to the corresponding terminals. After the tab **Call Interface** is selected and the check box **Call Interface** is activated the following screen appears:





**Figure 3-32: Call Interface**

The box text **Program name** defines which program is started as a Call Interface. E.g. the demo program for usage of the call interface can be entered C:\BCWIN32\BCCIW.IN.EXE. If no program is entered the data transmission kernel is also waiting for a response. See chapter 7 Data Transfer Kernels BCOMMIO. The text box **CI Timeout** defines how long the data transmission kernel is waiting for a response. If no response is received during the CI timeout BCOMMIO will shut down.

You must also select whether **Shared Memory** or **DDE** is used for the connection to the Call Interface (see chapter 9 Call Interface). If DDE is used for the connection to the Call Interface in the text box **DDE converter** the corresponding converter program must be entered, e.g. C:\BCWIN32\DDE2SMEM.EXE.

#### **Available Buttons:**

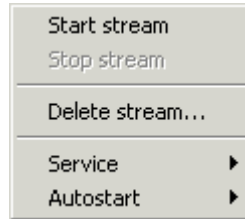
##### **Save**

With the button **Save** the settings are written into the configuration files. The message „Configuration saved“ follows.

##### **Exit**

You can exit the menu item INITIALIZE COMMUNICATION STREAM... by clicking the **Exit** button.

By clicking the icon of a stream with the right mouse button, the following popup menu appears:



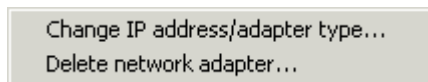
**Figure 3-33: Context menu of a stream**

START STREAM	With the menu item START STREAM the I/O kernel BCOMMIO for the selected communication stream is started.
STOP STREAM	With the menu item STOP STREAM the I/O kernel BCOMMIO for the selected communication stream is stopped.
DELETE STREAM...	With the menu item DELETE STREAM... the selected communication stream is removed from the configuration. <b>Attention: Doing this, all files related to the communication stream (terminal configuration, orders etc.) will be deleted also! Additionally entries of the stream as a service or in Autostart are removed also.</b>
INSTALL SERVICE REMOVE SERVICE START SERVICE STOP SERVICE	these four menu items are available only under Windows NT or Windows 2000. You can enter the stream into the service manager or remove it from the service manager. Also an installed service can be started or stopped.(see chapter 3.3.16 Services)
INSTALL AUTOSTART REMOVE AUTOSTART	with these two menu items the stream can be started automatically when Windows is starting or this feature can be removed.

All these menu items are also available in the STREAM main menu.

---

By clicking the icon of a network adapter with the right mouse button, the following popup menu appears:



**Figure 3-34: Context menu of a network adapter**

CHANGE IP ADDRESS/  
ADAPTER TYPE...

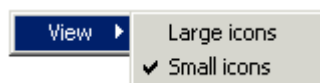
With the menu item CHANGE IP ADDRESS/ADAPTER TYPE... you can change the IP address and/or the type of the selected network adapter.

DELETE NETWORK ADAPTER...

With the menu item DELETE NETWORK ADAPTER... you can remove the selected network adapter from the configuration of the stream.

These menu items are also available in the NETWORK ADAPTER main menu.

Clicking into the free area within the list of streams or within the list of network adapters with the right mouse button brings up another popup menu, with which you can toggle the view of the list between large and small icons:



**Figure 3-35: Large/Small icons**

### 3.3.7. Start options

With this menu item you could set various start options for the communication streams. The following screen is displayed after selecting the menu item START OPTIONS..., when currently a LAN or COM stream is selected in the main window:

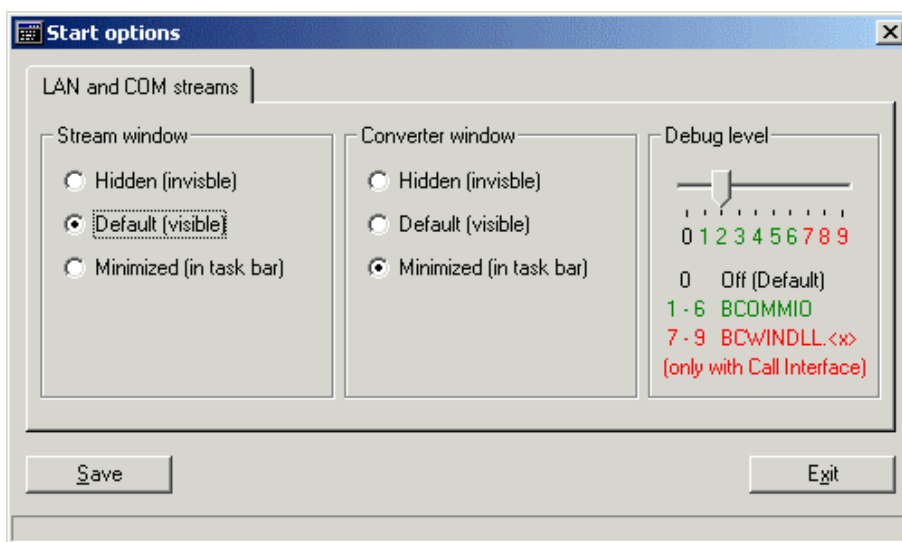


Figure 3-36: Start options for LAN and COM streams

For BCOMMIO resp. converter windows several start options can be entered. With the **Debug level** slider 0 (no display) to 9 (display details) within the BCOMMIO window can be entered. In detail, the debug level values have the following meaning:

**level 1:** only the data traffic between BCOMMIO and the connected terminals is shown

**level 2:** additionally the data traffic between BCOMMIO and the Call Interface is shown

**level 3:** additionally the UDP datagrams between BCOMMIO and the TERMSTAT (see chapter 5 The application TERMSTAT) program are shown

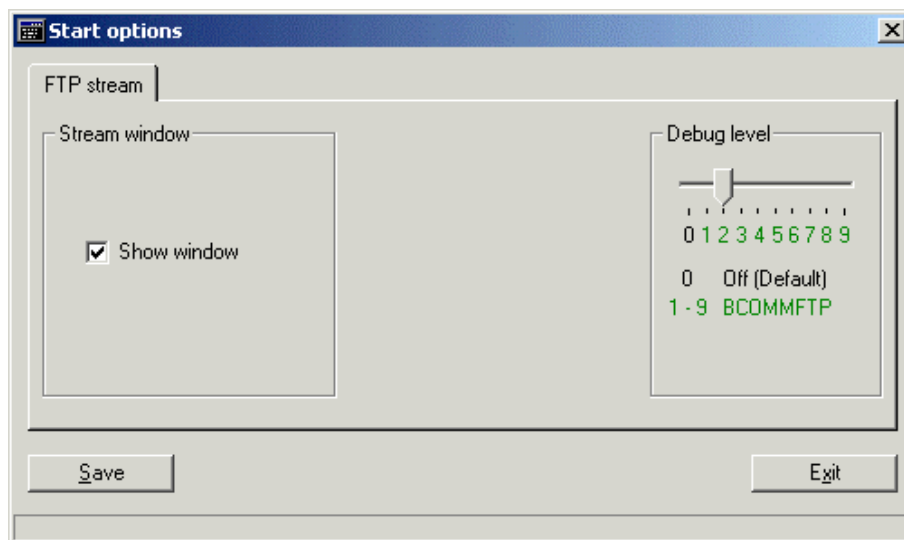
**level 4:** no additional information

**level 5:** additionally the data records between BCOMMIO and a network adapter are shown

**level 6:** if the switch debug\_poll in the BCOMM32.INI file (see chapter 7.2 Configuration file BCOMM32.INI) is set to Y, then the polling in a serial communication is shown

**level 7 to 9:** these levels should only be set when Kaba Benzing told so. When one of these levels is set, very detailed recordings in the file BCWIN.DLL are made, so the debug file grows rapidly.

When in the main window the FT stream is currently selected, the following dialog is displayed when selecting this menu item:



**Figure 3-37: Start options for FTP stream**

By use of **Show window** you decide whether BCOMMFTP has a visible window on the desktop or not.

In this dialog you can also set the **Debug level** for the FTP stream.

#### **Available Buttons:**

After the **Save** button is activated the settings are stored in the configuration file.

**For LAN and COM streams:** when the settings for the stream window and/or the debug level were changed, the new values are automatically sent to all currently active streams and will therefore take effect immediately.

#### **Exit**

You can exit the START OPTIONS... menu item by selecting the **Exit** button.

### 3.3.8. Directories

This menu allows to change the directories. The following screen is displayed when you select the DIRECTORIES... menu item:

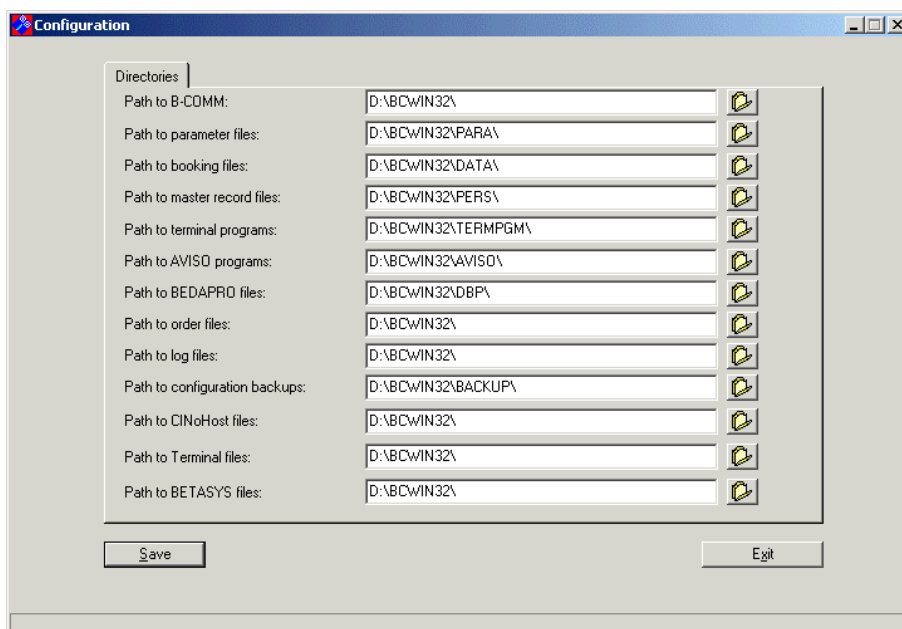


Figure 3-38: Directories

The directory **Path to B-COMM** refers to the path of the work and configuration files for B-COMM (except the configuration file BCOMM32.INI (see chapter 7.2 Configuration file BCOMM32.INI), which resides in the Windows directory).

In the **Path to parameter files** all parameters are stored which are received from a terminal via upload. The parameters which shall be transmitted to the terminal are also stored in this path.

In the **Path to booking files** all data records are stored which are received from a terminal. The files which shall be transmitted via the command TRANSMIT TRANSPARENT DATA... (see chapter 3.2.8 Transmit transparent data) or TRANSMIT DISPLAY RECORD... (see chapter 3.2.7 Transmit display record) must also be in this directory.

In the **Path to master record files** all master record files which shall be transmitted with the B-COMM commands TRANSMIT MASTER RECORD FILE... (see chapter 3.2.1 Transmit master record file) and AUTOM. MASTER RECORD DOWNLOAD... (see chapter 3.2.2 Autom. master record download) must be available. In this directory, the file created by the MASTER RECORD UPLOAD... (see chapter 3.2.3 Master record upload) command, will be stored.

In the **Path to terminal programs** all terminal programs which shall be transmitted with a SOFTWARE DOWNLOAD... (see chapter 3.4.4

---

Software download) must be available.

In the **Path to AVISO programs** all files have to be stored, which should be sent to or received from a Bedas 929Ø with the program option AVISO.

In the **Path to BEDAPRO files** all files have to be stored, which should be sent to or received from a Bedas 956Ø with the program option BEDAPRO.

In the **Path to order files** the (time-controlled) order files have to be stored, which should be executed by BCOMMIO/BCOMMFTP.

Into the **Path to log files**, BCOMMIO and BCOMMFTP write their log, debug and system message files.

In the **Path to configuration backups** the configuration files are stored when a backup of the configuration is made or when a saved configuration should be restored.

In the **Path to CInoHost files** BCOMMIO writes temporary files during its operation. In network installations, where B-COMM WIN32 is installed on a server in the network but BCOMMIO is running on a workstation, this path should be set to a local path of the workstation to avoid heavy network traffic.

#### **Available buttons:**

##### **Save**

After the **Save** button is activated the settings are stored in the configuration file.

##### **Exit**

You can exit the DIRECTORIES... menu item by selecting the **Exit** button.

### 3.3.9. Printer log

BCWIN offers the possibility of logging data communications with the terminals. The following screen is displayed when you select the PRINTER LOG... menu item and activate the **Printer protocol** check box:

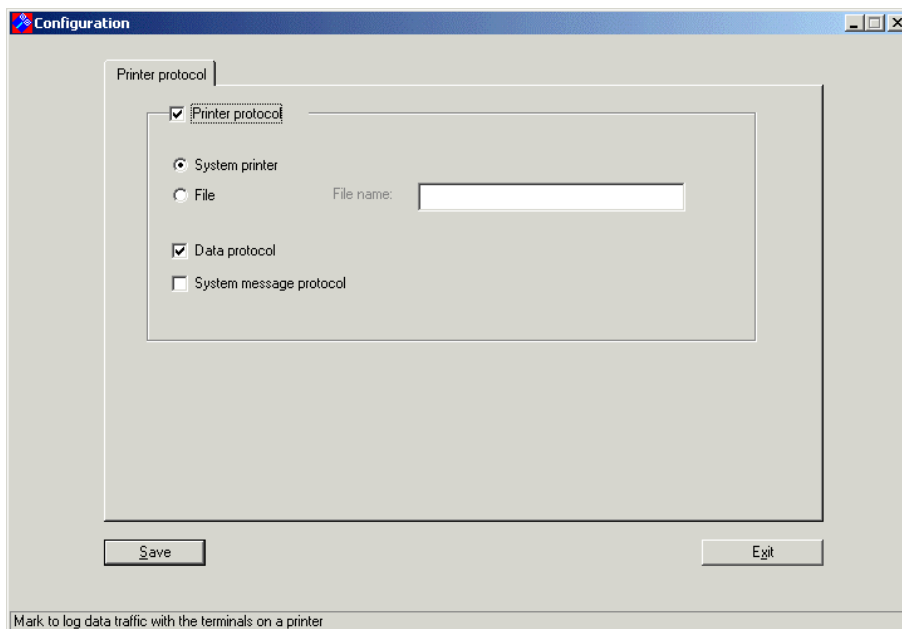


Figure 3-39: Printer log

In this screen must be entered whether the standard **system printer** shall be used. Each printer, also a network printer, can be configured. If the system printer is used, printing starts after the communication is stopped.

Data communications can also be logged in a **file**. Enter the name of the file in the **File name** text box. The file is stored as an **ASCII file**. If this file already exists, the data is appended to it.

With the **Data protocol** check box, you can specify whether you wish logging of data communications with the terminals. When the **System message protocol** check box is activated, logging of occurring system messages is also made.

#### Available buttons:

##### **Save**

After the **Save** button is activated the settings are stored in the configuration file.

##### **Exit**

You can exit the PRINTER LOG... menu item by selecting the **Exit** button.



### 3.3.10. File definitions

By this menu item you can define which files are displayed as a default in several functions. It can also be defined whether a log file is created and the booking file is mirrored. The following screen is displayed when you select the FILE DEFINITIONS... menu item:

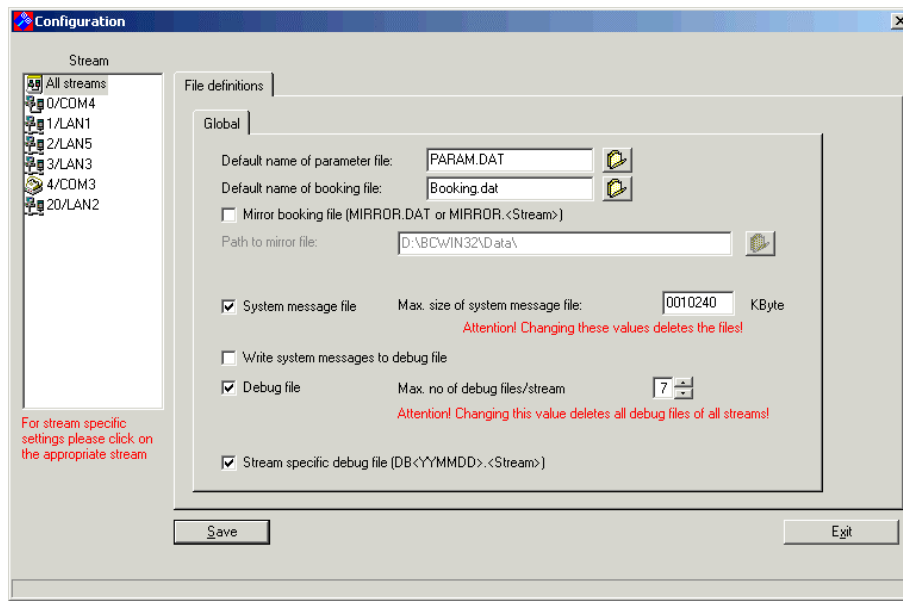


Figure 3-40: Global file definitions

In the **Default name of parameter file** text box the name of the parameter file is entered which shall be displayed as a default for parameter download resp. parameter upload (see chapter 3.4.5 Parameter setting).

In the **Default name of booking file** text box the name of the booking file (see chapter 7.6 Booking file BOOKING.DAT) is entered. This file is then presented by BCWIN in several functions as a default file.

**When BCOMMIO cannot access the booking file, the data is saved in the file BOOKSAV.DAT.**

When the **Mirror booking file** check box is activated, the booking data is mirrored in another file. The filename is MIRROR.DAT. The **Path to mirror file** text box is used for defining the path where the mirror files should be saved. When the check box **Stream specific booking file...** is activated, then also a specific mirror file is written (MIRROR.<Stream>).

The **System message file** check box defines whether a system message file (see chapter 7.4 System Message File SYSMESS.<Stream>) is created. The **Max. size of system messages file** text box defines the size of the system message file. By reaching the defined size, the oldest messages will be overwritten.

**Attention: Changing these values deletes the files !**

With the check box **Write system messages to debug file** you can specify whether system messages are also written into the debug file.

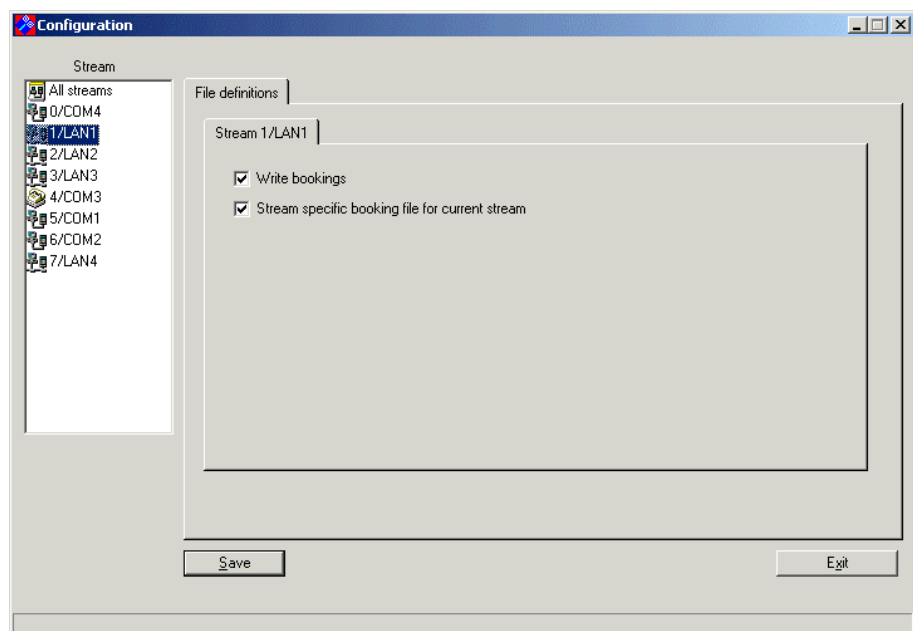
The **Debug file** check box defines whether a debug file is created. The debug files have the filename DB<YYMMDD>.DBG, where <YYMMDD> is the date of the creation of the file.

With the text box **Max. no. of debug files/stream** you can specify the max. number of debug files which are written by BCOMMIO. When this value is e.g. 7 (default), then the debug files of the last seven days are always available, which means that on the 8<sup>th</sup> day, the oldest file is automatically deleted.

**Attention: Changing this value will delete all debug files of all streams and the corresponding entries in the configuration file BCOMM32.INI.**

The check box **Stream specific debug file** specifies whether there's only one debug file for all configured streams or one debug file for each stream. In the latter case, the name of the debug file is DB<YYMMDD>.<Stream>.

When you click on one of the configured streams, you can configure more, stream specific, settings. The following screen mask is displayed when you click on one of the streams:



**Figure 3-41: Specific file definitions**

---

With the check box **Write bookings** you can decide whether bookings from the terminals should be written into a booking file or not.

By use of the check box **Stream specific booking file for current stream** you can specify whether a specific booking file for the currently selected stream should be written. The name of the file then is BOOKING.<Stream> instead of the name entered under **Default name of booking file**.

**Available buttons:**

**Save**

After the **Save** button is activated, the settings are stored.

**Exit**

You can exit the FILE DEFINITIONS... menu item by selecting the **Exit** button.

### 3.3.11. FTP Transfer configuration

By this menu item you can define the default login parameters for the I/O kernel BCOMMFTP to login to a Bedanet 938Ø (B-Net 938Ø) terminal. The following screen is displayed when you select the FTP TRANSFER CONFIGURATION... menu item:

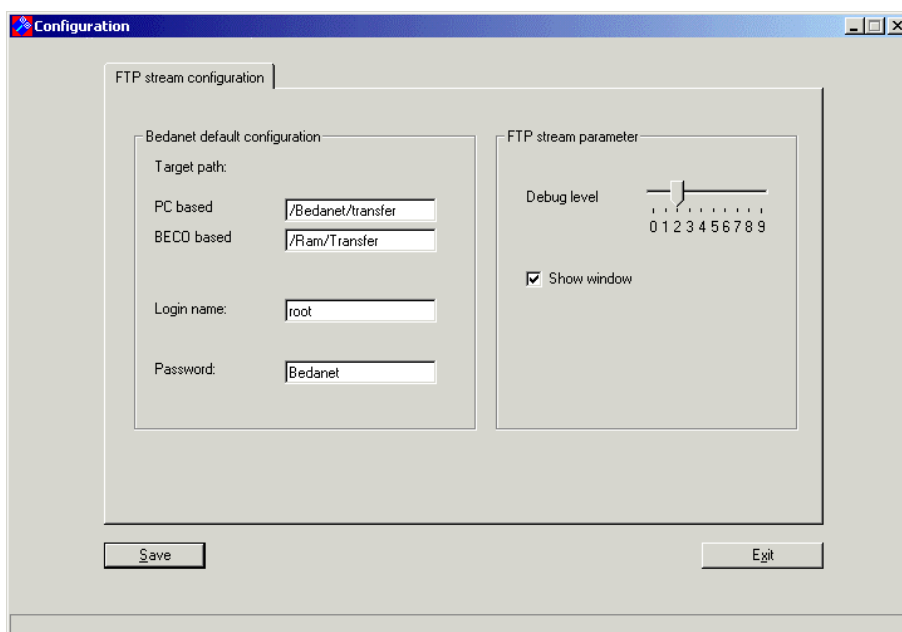


Figure 3-42: FTP Transfer configuration

Into the text boxes **Target path PC based**, **Target path BECO based**, **Login name** and **Password** you have to enter the parameters, with which BCOMMFTP logs in into a Bedanet (B-Net) terminal. These values are used by BCOMMFTP by default, if no special values are entered for a Bedanet (B-Net) during logon (see chapter 3.3.2 Terminal logon).

By use of the slider **Debug level** the debug level of BCOMMFTP can be set. The higher this value is, the more detailed are the outputs of the kernel.

When the check box **Show window** is activated, the window of BCOMMFTP is visible on the Windows desktop.

#### Available buttons:

##### **Save**

After the **Save** button is activated, the settings are stored.

##### **Exit**

You can exit the FTP TRANSFER CONFIGURATION... menu item by selecting the **Exit** button.

---

### 3.3.12. Change password

A password is used to protect individual areas of BCWIN (terminal logon, terminal or order deletion, interface initialisation, password editing and master record editing). Letters and numbers can be used for the password. Entered characters are not displayed. If no password is specified, all functions can be executed without a password query.

You can modify the password. After selecting the CHANGE PASSWORD... menu item, you must enter the previous password, if a password has been entered during installation. The following screen is then displayed:



**Figure 3-43: Change password**

You must specify the new password in the **Enter new password** text box and confirm it in the **Confirm password** text box.

#### **Available buttons:**

**OK**

The new password is accepted when you select the **OK** button.

**Cancel**

You can cancel the CHANGE PASSWORD... menu item by selecting the **Cancel** button.

### 3.3.13. Backup

By this menu item you can create backups of the current configuration of B-COMM WIN32. The following screen is displayed when you select the BACKUP... menu item:

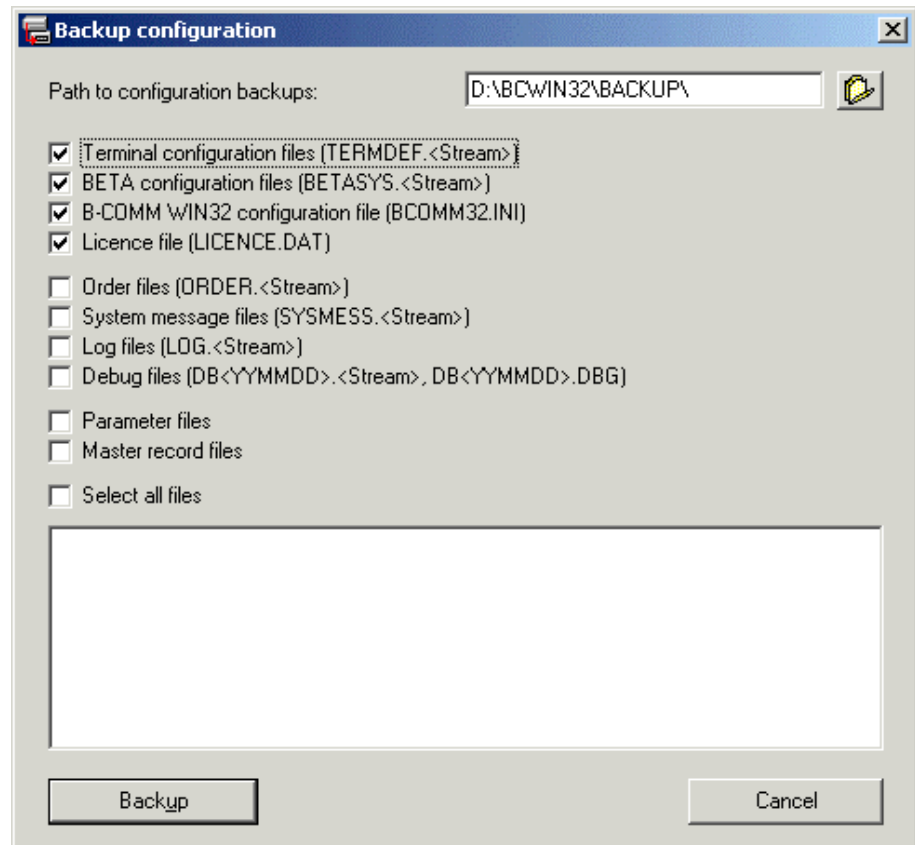


Figure 3-44: Backup

With the textbox **Path to configuration backups** you can select the target directory, where the files should be stored. By default, this is the directory you configured by the menu item DIRECTORIES... (see chapter 3.3.8 Directories). **If the selected directory does not exist, it will be created automatically when you start the backup process.**

By use of the various checkboxes you can select, which files should be saved. **You should at least backup the terminal configuration files, the BETA configuration files and the configuration file BCOMM32.INI, because these files contain the complete configuration of your system.**

---

**Available buttons:**

**Backup**

With the button **Backup** the selected files are saved in the target directory.

**Attention: if a file already exists in the target directory, it is overwritten without asking you. If you want to save different configurations, you have to create different subdirectories for the backups.**

**Cancel**

With the button **Cancel** you can cancel the menu item BACKUP.

### 3.3.14. Restore

By this menu item you can restore a configuration of B-COMM WIN32, which was previously saved with the menu item BACKUP. The following screen is displayed when you select the RESTORE... menu item:

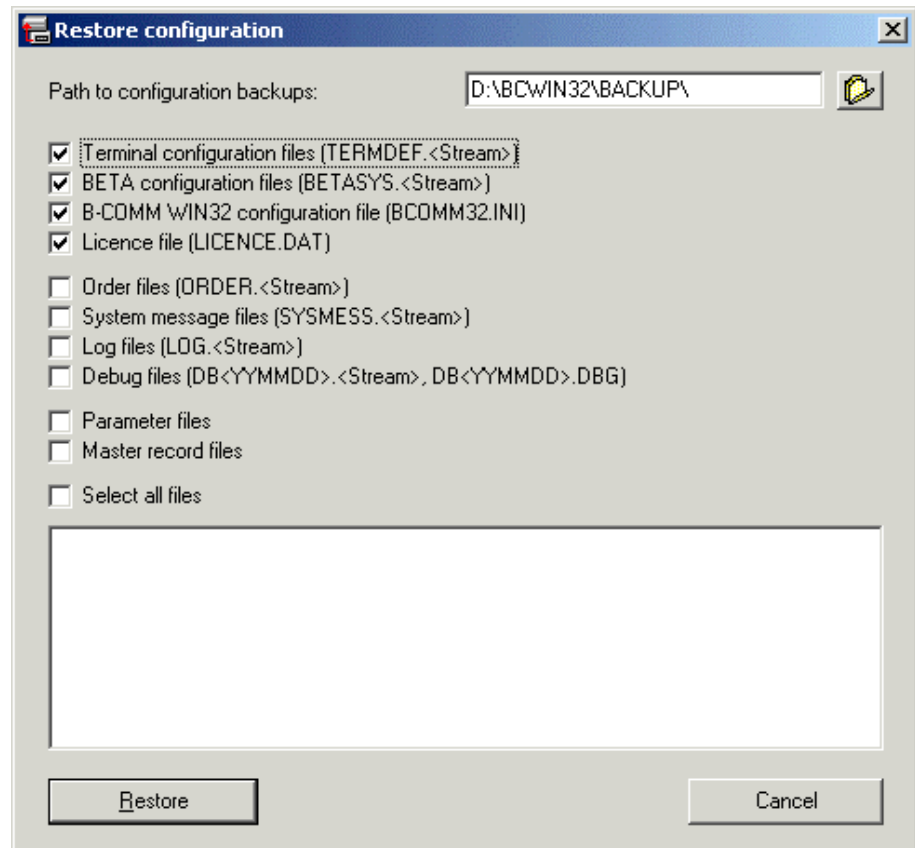


Figure 3-45: Restore

With the textbox **Path to configuration backups** you can select the source directory, where the backups are stored. By default, this is the directory you configured by the menu item DIRECTORIES... (see chapter 3.3.8 Directories).

By use of the various checkboxes you can select, which files should be restored. **You should at least restore the terminal configuration files, the BETA configuration files and the configuration file BCOMM32.INI, because these files contain the complete configuration of your system.**



---

**Available buttons:**

**Restore**

With the button **Restore** the selected files are restored from the source directory.

**Attention: if a file already exists in the target directory, it is overwritten without asking you.**

**Cancel**

With the button **Cancel** you can cancel the menu item RESTORE.

### 3.3.15. Scheduler

You can configure and switch on or off the scheduler by selecting this menu item. The following screen is displayed when you select the SCHEDULER... menu item:

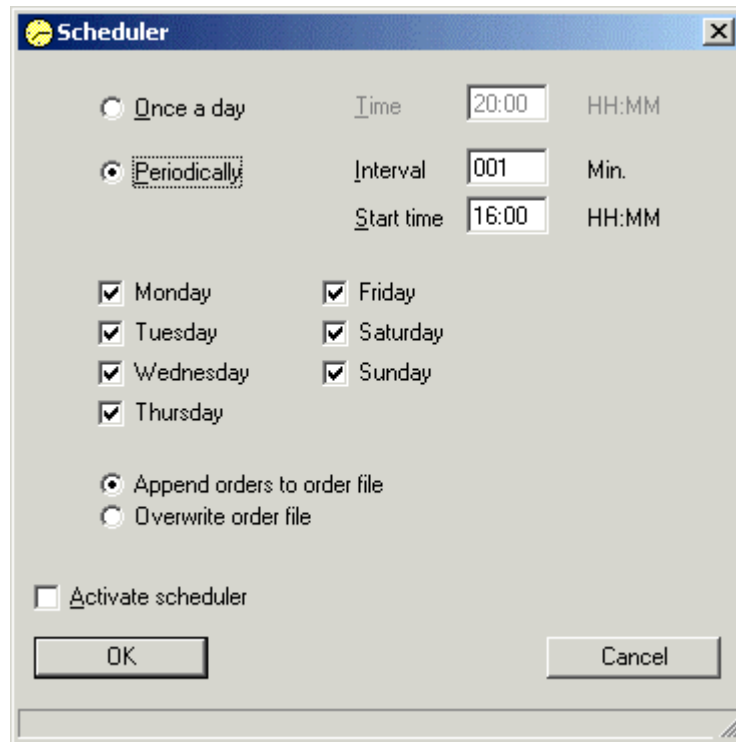


Figure 3-46: Scheduler

In this dialog you can specify whether time-controlled orders will be executed **Once a day** or **Periodically**. Also you can select the days of the week, on which the orders will be executed..

By use of the radio buttons **Append orders to order file** and **Overwrite order file** it's specified, whether time-controlled orders are appended at the end of the „normal“ order file or if the order file is overwritten by the time-controlled order(s).

Attention! If **Append orders to order file** radio button is selected, the size of the order file can grow very quickly.

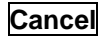
With the check box **Activate scheduler** the scheduler is activated or deactivated.

---

**Available buttons:**



With the button **OK** the settings are written into the configuration file and, if the I/O kernel is currently active, sent to BCOMMIO.



You can cancel the SCHEDULER... menu item by selecting the **Cancel** button.

**Notes about time-controlled orders:**

To execute an order time-controlled, the orders must be set with the option „time-controlled“. The scheduler must be activated. The communication must be started. Time-controlled orders will be executed by the next valid moment set in the scheduler menu. Time controlled orders will be executed by every moment set in the scheduler menu until they are deleted (see chapter 3.5.2 Delete all orders).

When the start time is overstayed when the communication is started, BCOMMIO behaves like follows:

when „Once a day“ is active, the order is executed immediately, when “Periodically” is active, then the order is executed at the next possible interval.

Example for „Periodically“:

Start time 16:00, Interval 120 minutes, current time 17:00: the order is executed first at 16:00 + 120 minutes = 18:00.

### 3.3.16. Services

This menu item is only available under Windows NT 4.0, Windows 2000 or Windows XP.

With this menu item the configured streams can be entered into the Service Control Manager of Windows or removed from it. Additionally it's possible to start or stop a stream which is entered as a service.

After selecting the SERVICES... menu item, the following screen is shown:

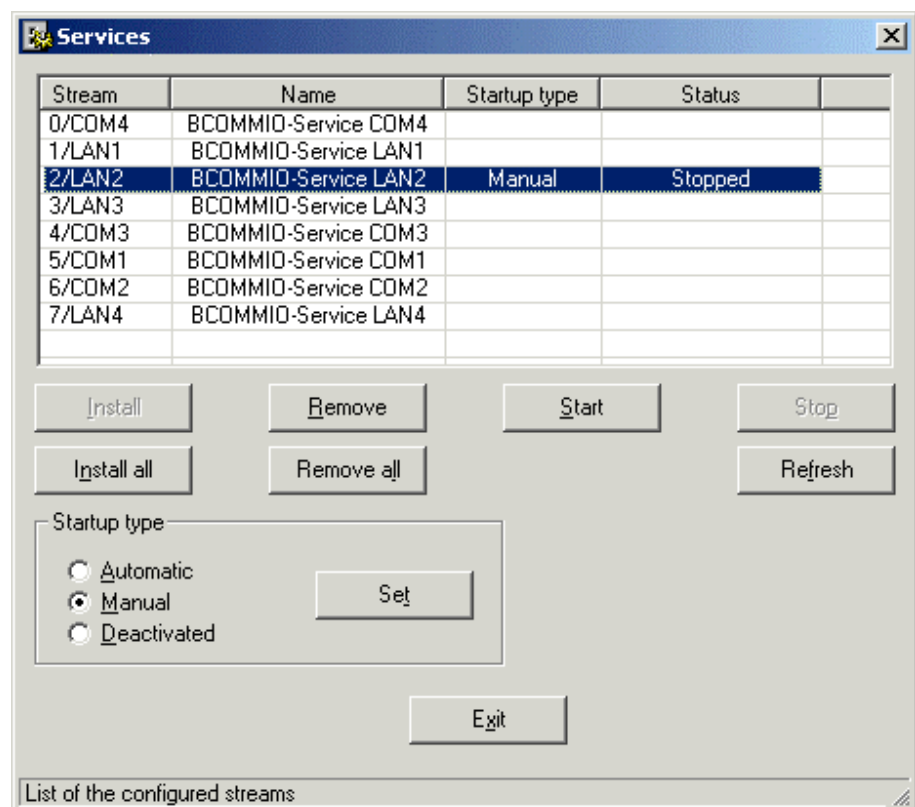


Figure 3-47: Services

By use of list of streams you can select one of the configured streams.

#### Available buttons:

##### **Install**

With the button **Install** the selected stream is entered into the Service Control Manager. You can select whether the service should be started automatically on system start or manually or if it should be deactivated.

##### **Remove**

With the button **Remove** a stream installed as a service is removed from the Service Control Manager. If it is currently active, it's stopped first.

---

**Start**

With the button **Start** a stream installed as a service is started.

**Stop**

With the button **Stop** a stream as a service is stopped.

**Install all**

With the button **Install all** all streams are entered into the Service Control Manager. You can select whether the services should be started automatically on system start or manually or if they should be deactivated.

**Remove all**

With the button **Remove all** all streams installed as services are removed from the Service Control Manager. If one of them is currently active, it will be stopped first.

**Set**

With the button **Set** you can change the startup type of a service at any time. The change will take effect as soon as the service is stopped.

**Refresh**

With the button **Refresh** you can refresh the list of streams.

**Exit**

With the button **Exit** you can exit the dialog SERVICES....

**Instead of this dialog you can also use the appropriate item in the Windows Control Panel.**

---

### 3.3.17. Deutsch/English GB/English US

You can switch to texts in other languages by selecting this menu item. You can switch over to other texts and back to English texts again at any time. For implementing another language some modifications and extensions must be made in the file LANGUAGE.INI (see chapter 8 Translation into other Languages).



Figure 3-48: Change language

---

### 3.3.18. Set PC date/time

You can modify the PC date and time with this menu item. The following screen is displayed when you select the SET PC DATE/TIME... menu item:

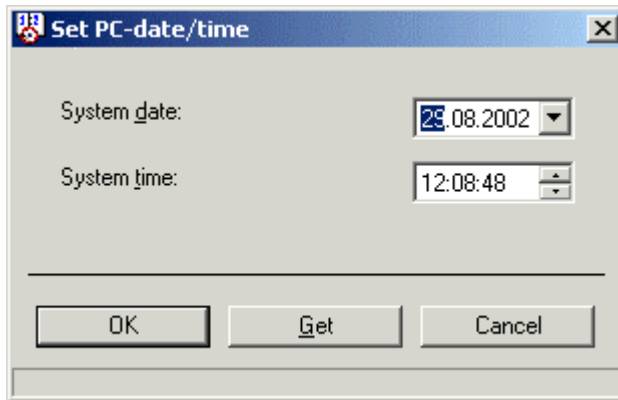


Figure 3-49: Set PC date/time

You can enter the date and time in the **System date** and **System time** entry fields. The input formats depend on the settings made in CONTROL PANEL – REGIONAL SETTINGS.

#### Available buttons:

##### **OK**

The setting is accepted when you select the **OK** button.

##### **Get**

When you select the **Get** button, the currently valid system date and time are displayed in the **System date** and **System time** entry fields.

##### **Cancel**

You can cancel the SET PC DATE/TIME... menu item by selecting the **Cancel** button.

### 3.4. System setup

When you select the SYSTEM SETUP menu, you can choose between the AT MODEM..., CREATE MODEM PARAMETER FILE..., DEVICE INTERFACE..., SOFTWARE DOWNLOAD... and PARAMETER SETTING... menu items.

If the FTP stream 30 is currently selected, only menu item PARAMETER SETTING... is available.

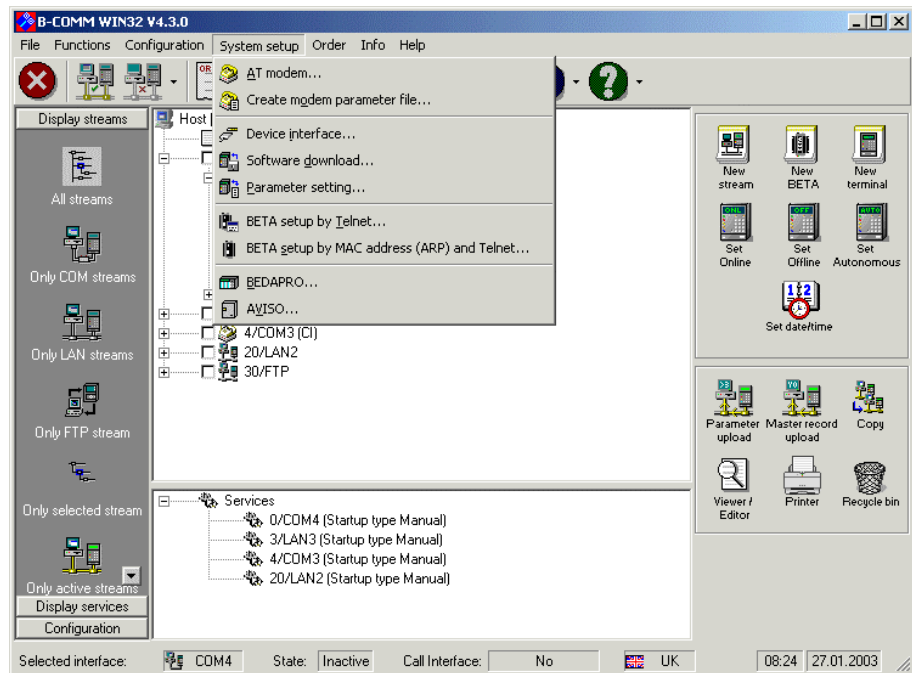


Figure 3-50: Menu System setup

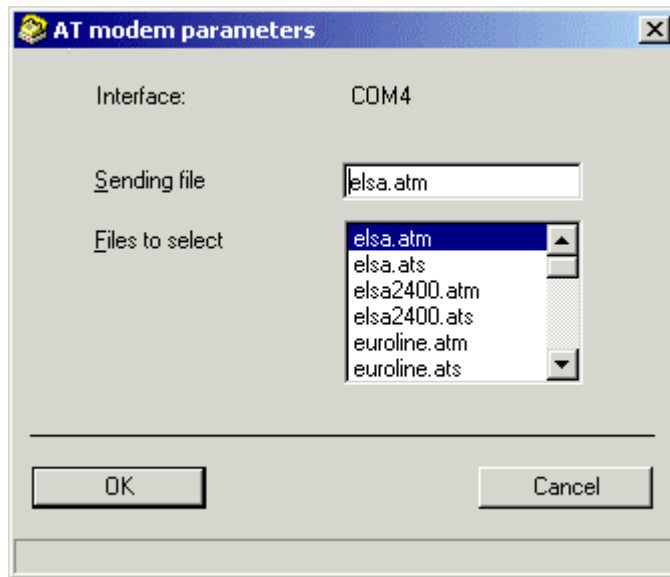


---

### 3.4.1. AT Modem

You can set the parameters of an AT modem in this menu item. The AT modem whose parameters you wish to set must be connected directly to the computer. It is not possible to set parameters via the modem link.

The following screen is displayed when you select the AT MODEM... menu item:



**Figure 3-51: AT modem**

In the **Sending file** text box, specify the file containing the parameters you wish to send. Alternatively you can select a file from the drop-down combo box **Files to select**.

Files with the extension .ATM contain the parameters for the master modem which is connected to the Windows system. Files with the extension .ATS contain the parameters for the remote modem where the terminals will be connected. You can create own parameter files copying a pair ATM/ATS files with another file name and modify the included AT commands for your modem or by selecting the menu item CREATE MODEM PARAMETER FILE...(see chapter 3.4.2 Create modem parameter file)

The following parameters should be included in the ATM file:

- commands are not echoed (usually with ATE0)
- modem hangs up if DTR changes from ON to OFF
- no hardware flow control
- no automatically call acceptance (usually with ATS0=0)
- modem acknowledgement as text (not coded in numbers)

In the ATS file for the slave modem automatically call acceptance must be set (usually with ATS0=1).

---

**Available buttons:**

**OK**

Setting the modem's parameters is started when you activate the **OK** button. If the corresponding data transfer kernel is not active, you can start it in a screen that follows. Modem parameter setting is successful if no error messages appear.

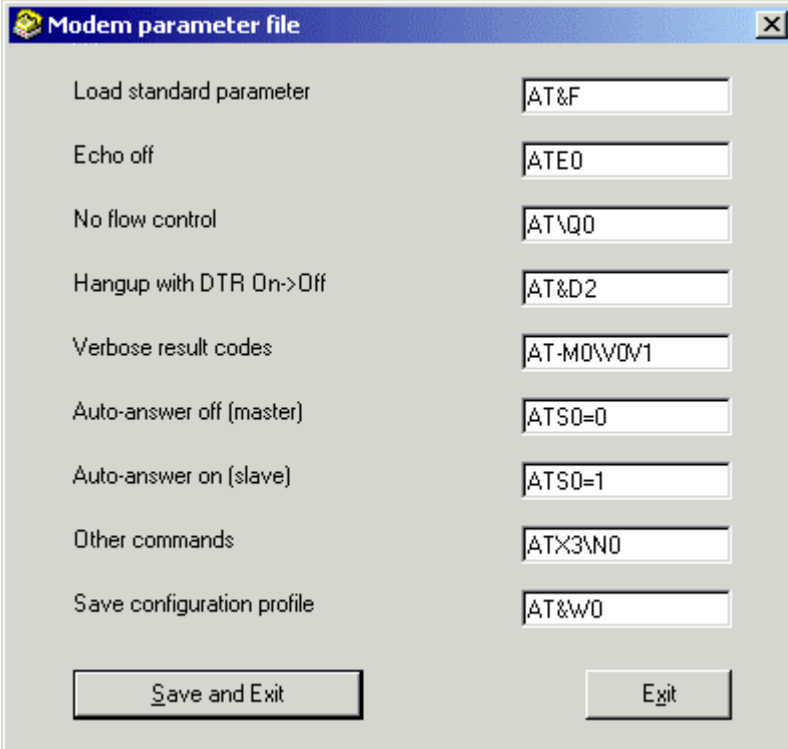
**Cancel**

You can cancel the AT MODEM... menu item by selecting the **Cancel** button.

### 3.4.2. Create modem parameter file

With this menu item you can create your own modem parameter file, in case the parameter files delivered with B-COMM don't fit your needs. The settings from the files ELSA.ATM and ELSA.ATS are the basis for this menu item.

After selecting the menu item CREATE MODEM PARAMETER FILE... the following screen appears:



Label	Value
Load standard parameter	AT&F
Echo off	ATE0
No flow control	AT\Q0
Hangup with DTR On->Off	AT&D2
Verbose result codes	AT-M0W0V1
Auto-answer off (master)	ATS0=0
Auto-answer on (slave)	ATS0=1
Other commands	ATX3\N0
Save configuration profile	AT&W0

Buttons: Save and Exit, Exit

Figure 3-52: Create modem parameter file

Except the text box **Other commands** you must enter something into the various fields, because these settings are necessary to make the modem work with B-COMM.

#### Available buttons:

##### **Save**

After pressing the button **Save** and entering a filename the parameter files ATM und ATS will be written and the dialog ends.

##### **Exit**

You can exit the menu item CREATE MODEM PARAMETER FILE... with the **Exit** button.

### 3.4.3. Device interface

You can set a terminal's data transfer and BPA/9 parameters in this menu item. You can only set a terminal's parameters if you have selected the interface to which the terminal is connected.

The following screen is displayed when you select the DEVICE INTERFACE... menu item:

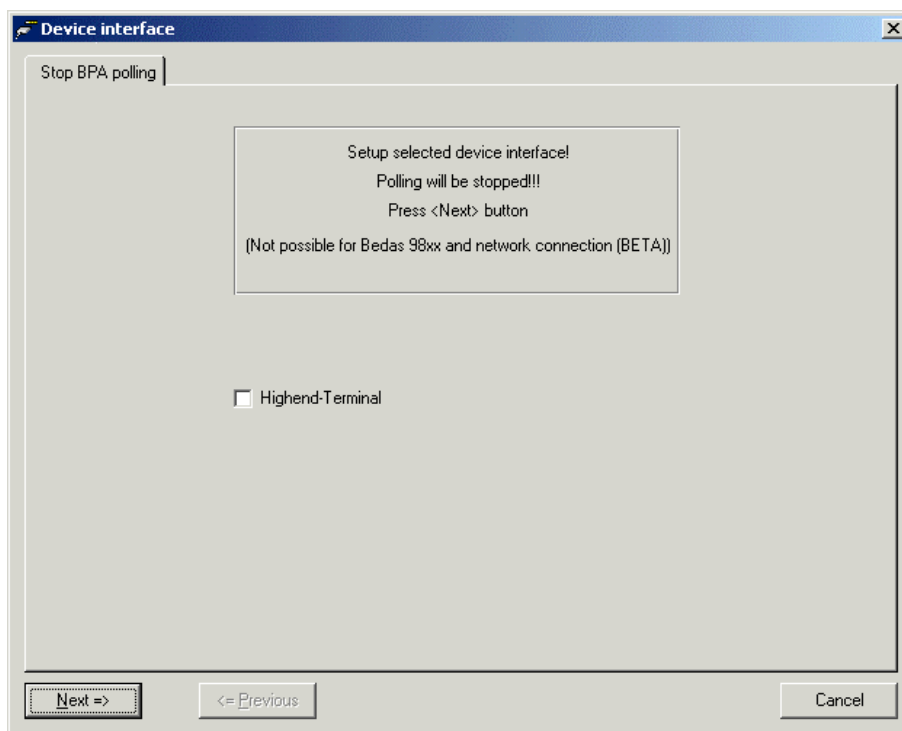


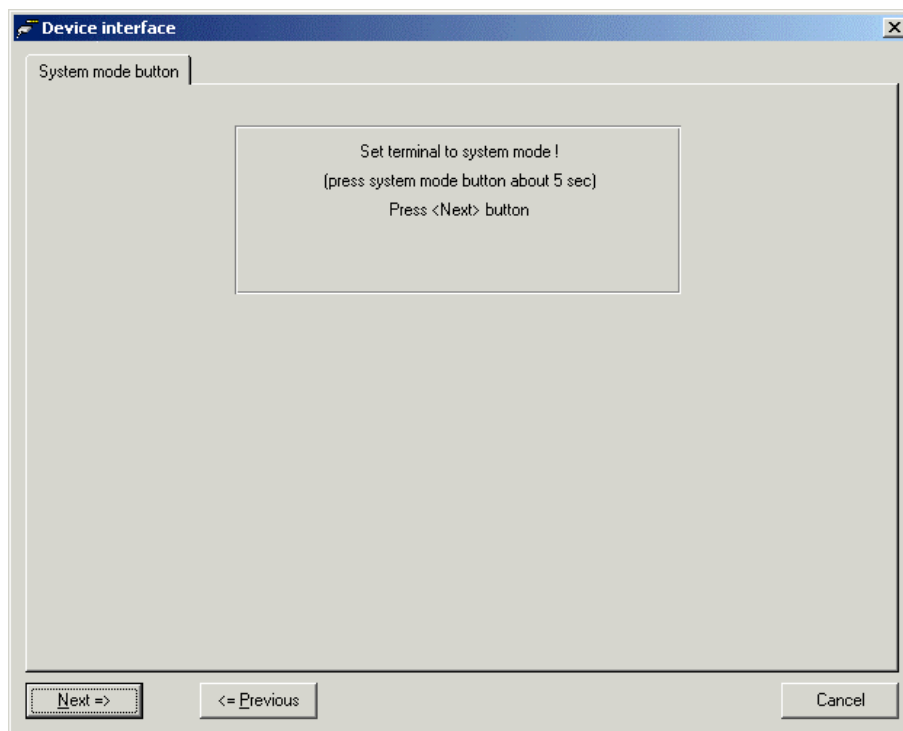
Figure 3-53: Device interface 1

To configure the device interface of a terminal, at first the device has to be switched into System mode. For that, usually the system mode button in the terminal has to be pressed for about 5 seconds. In case the terminal is a Highend device, it can be switched to system mode also by software, e.g. by BCOMMIO, so that a user will not have to open the terminal and press the button manually.

For that reason, you first have to tell BCWIN by use of the checkbox **Highend-Terminal**, whether the device you want to setup is a Highend device or not.

When you activate the **Next =>** button, you must start the data transfer program in a screen that follows, provided it has not already been started.

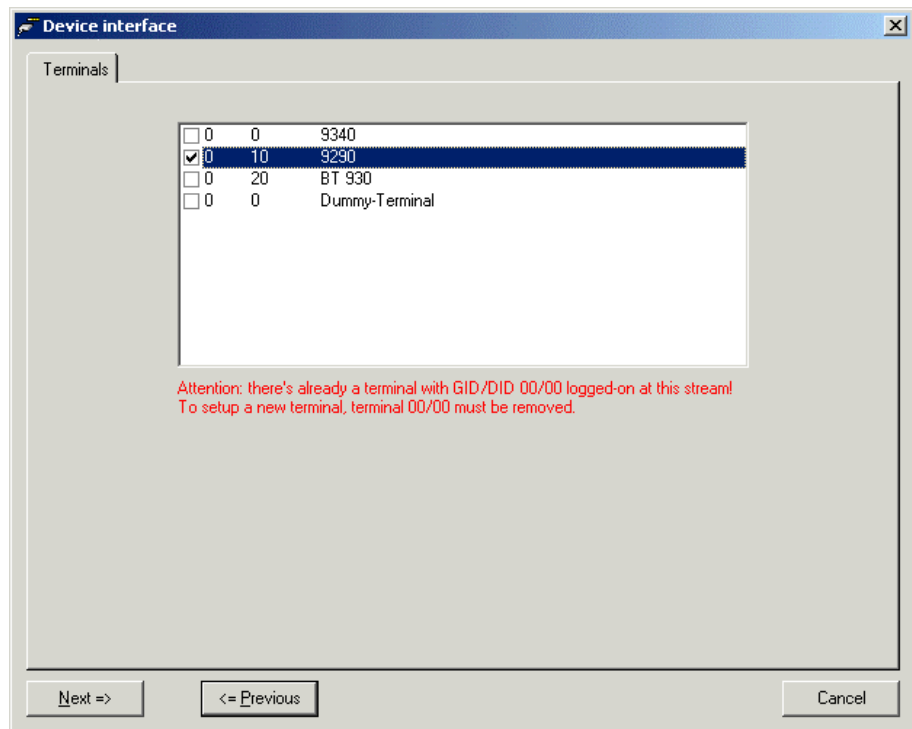
When you didn't activate the checkbox **Highend-Terminal**, the following screen is then displayed:



**Figure 3-54: Device interface 2**

You must now switch the terminal to the system mode. To do this, press the System Mode button for about 5 seconds.

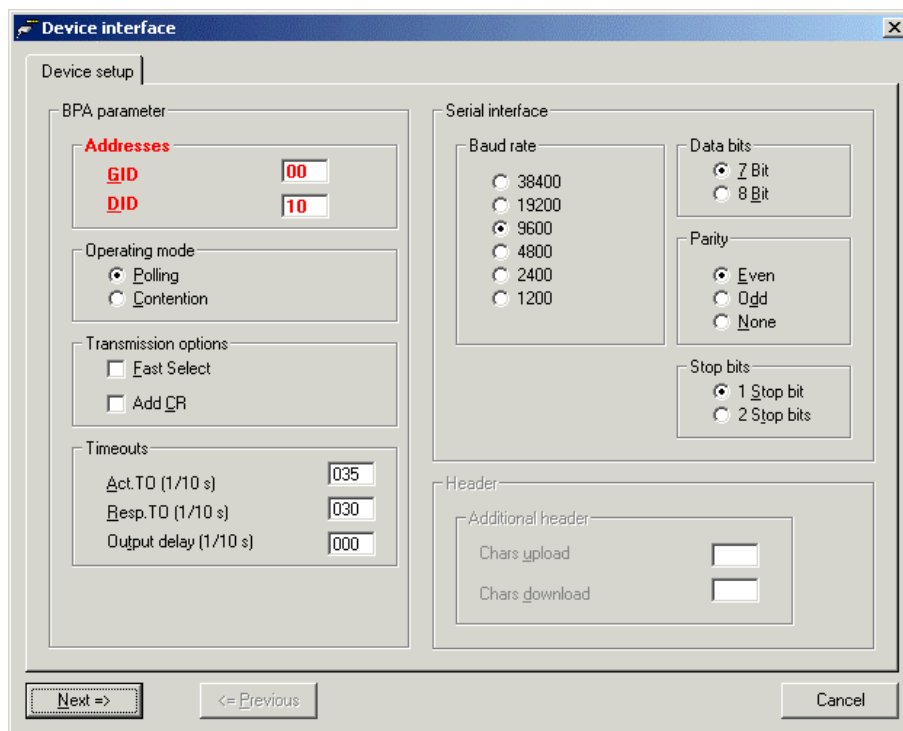
If you activated the checkbox **Highend-Terminal** in the first dialog, in case of the above screen mask BCWIN now displays a list of the terminals, which are logged on at the stream. You can now select the terminal you want to setup:



**Figure 3-55: Device interface 3**

Attention: if you want to setup a new terminal, you have to make sure, that no terminal with the address 00/00 is already logged on at the stream, because a new terminal also has the address 00/00. In such a case, you will have to remove the terminal from the terminal system before you can start the setup of the new terminal.

The following screen is displayed when you activate the **Next =>** button and after the "Receiving device settings" message:



**Figure 3-56: Device settings**

In this dialog you can now specify the BPA parameters, the settings for the serial interface and – in case the terminal is a subterminal – additional header informations.

The parameters are transferred to the terminal or a subterminal when you activate the **Next =>** button. To this end, the "Sending device settings" and "Starting BPA polling..." messages are displayed.

### 3.4.4. Software download

You can perform a program download to a terminal with this menu item. A program download to a terminal is only possible if you have selected the interface to which the terminal is connected. The terminal programs must be in the path that you specified (see chapter 3.3.8 Directories).

The following screen is displayed when you select the SOFTWARE DOWNLOAD... menu item:

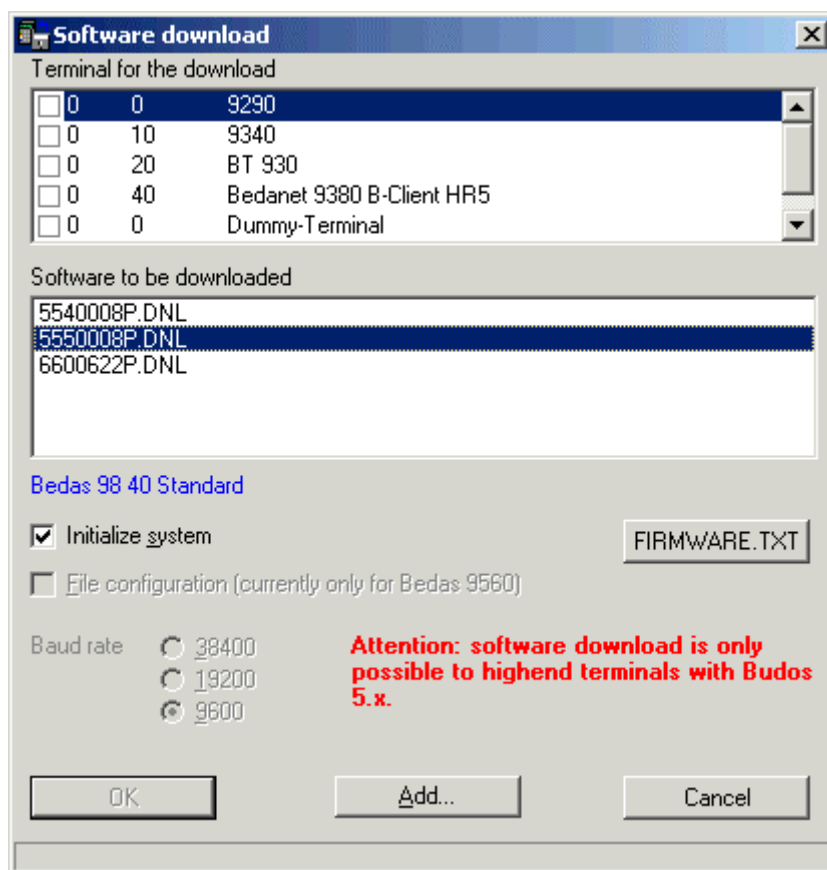


Figure 3-57: Software download

All terminals that are connected to the selected stream are displayed along with the GID, DID, Interface and device designation information. You can highlight one specific terminal for a program download by means of the cursor keys or by clicking with the mouse.

You can highlight specific software to be downloaded to the terminal by means of the cursor keys or by clicking with the mouse.

Below the list, in which the available software is listed, you can see (in blue letters), for which terminal type the currently selected software is applicable.



---

This information is taken from the file TERMTYP.TXT, which resides in the installation directory of B-COMM WIN32.

In the **Initialize system** check box, you define whether or not you wish to initialize the terminal's memory. The memory must be initialized whenever a new program is downloaded.

In the **File configuration** check box, you define if the table-definition for a Bedas 956Ø should be transmitted to a terminal too.

With the **Baud rate** radio buttons you have to define the baud rate for the software download.

**Available buttons:**

**OK**

You can store the order in the order file by selecting the **OK** button.

**Add**

You can add a new terminal program by selecting the **Add** button.

**Cancel**

You can cancel the SOFTWARE DOWNLOAD... menu item by selecting the **Cancel** button.

**FIRMWARE.TXT**

Clicking the button **FIRMWARE.TXT** displays a file, in which you find a description of which software is suitable for which terminal.

---

### 3.4.5. Parameter setting

You can fetch parameters of a terminal in this menu item. You can also send a parameter file to a terminal. The following screen is displayed when you select the **PARAMETER SETTING...** menu item:

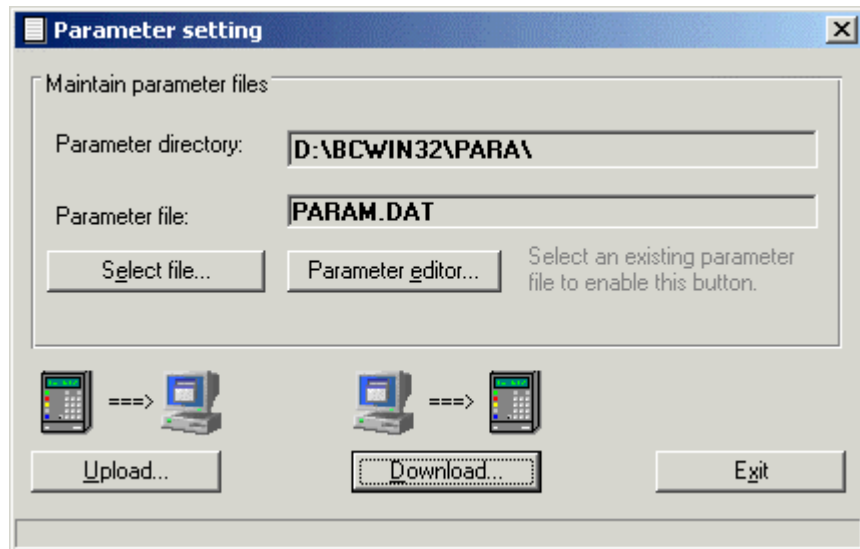


Figure 3-58: Parameter setting

The button **Upload** is not available if the FTP stream 3Ø is currently active.

The default parameter directory and the name of the default parameter file are inserted into the fields **Parameter directory** and **Parameter file** when this dialog is displayed the first time (see chapters 3.3.8 Directories and 3.3.10 File definitions). Into these two fields always the directory and parameter file selected last are inserted, whereby it doesn't matter whether the directory and file are selected by the button **Select file** in this dialog or if that is done in one of the dialogs **Parameter upload** or **Parameter download**.

---

#### Available buttons:

##### **Select file**

You can select a parameter file which should be worked with the parameter editor (option) by selecting the **Select file** button.

##### **Parameter editor**

If a parameter editor (option) is installed for the parameter file selected with the **Select file** button you can start the parameter editor by selecting the **Parameter editor** button. Then the parameter file can be worked by the parameter editor.

##### **Exit**

You can exit the PARAMETER SETTING... menu item by selecting the **Exit** button.

##### **Upload**

You can fetch parameters of the highlighted terminal by selecting the **Upload** button (see chapter 3.4.5.1 Parameter upload).

##### **Download**

You can send parameters from a file to a terminal by selecting the **Download** button (see chapter ).

### 3.4.5.1 Parameter upload

At first, target path and file for the uploaded parameters must be entered in a dialog. Afterwards the following screen is displayed:

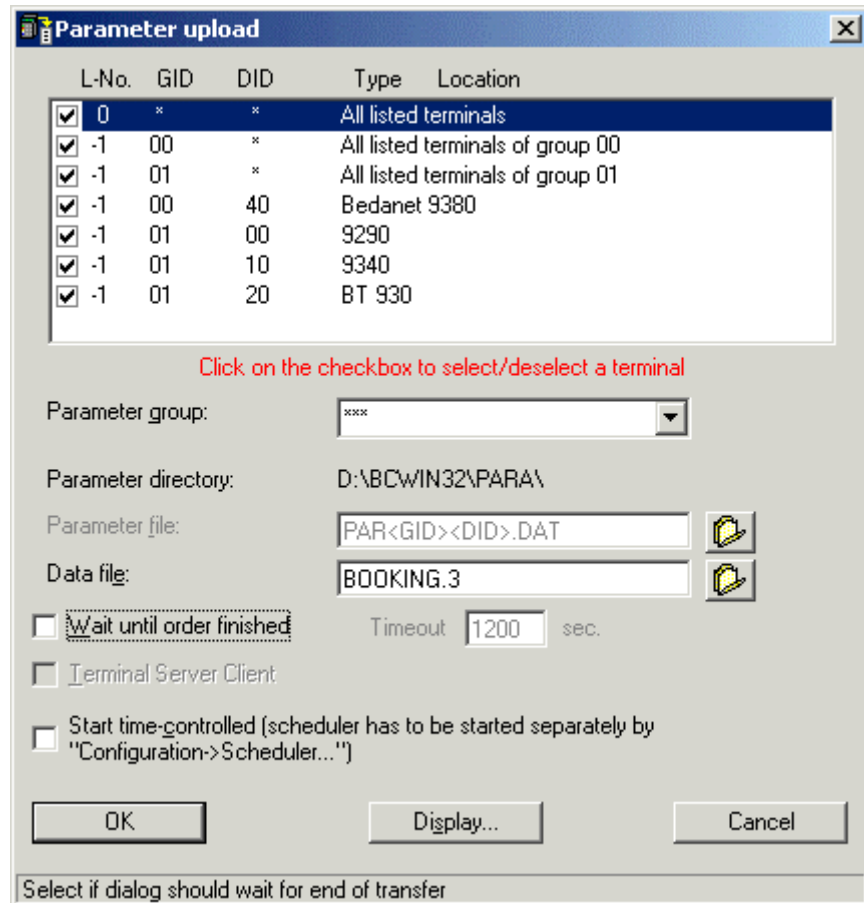


Figure 3-59: Parameter upload

Select the terminals from which you want to upload the parameters. In the **Parameter group** drop-down combo box, you must enter which parameters of a terminal you wish to fetch (refer also to the Parameter Setting Manual of the respective terminal). The use of wildcards (\*) is permitted. You must enter „xxx“ here if you wish to fetch all parameters of a terminal.

In the **Parameter file** text box, enter the file in which you wish to store the received parameters. This file is stored as an **ASCII file**. If this file exists, it is overwritten. The program number is entered in the first record of this file. The program number indicates whether the parameters originate from a time recording terminal, for example. If you edit the file later on and send it to a terminal by means of a parameter download, you must make sure that the parameter file matches the terminal concerned.

---

If more than one terminal is selected for upload, you can't enter a filename. In this case for each terminal a separate file is created. The filename consists of the GID and DID of the terminal, e.g. PAR0001.PAR for parameter records of terminal 00/01.

**Example of a parameter file:**

```
570-00-X-K08
@@0>3B01Mailbox-Text 1
@@0>3B02Mailbox-Text 2
@@0>3B03Mailbox-Text 3
@@0>3B04Mailbox-Text 4
@@0>3B05Mailbox-Text 5
@@0>3B06Mailbox-Text 6
@@0>3B07Mailbox-Text 7
@@0>3B08Mailbox-Text 8
```

All data transferred from the terminal during execution of this function is stored in the file that you specify in the **Data file** text box.

If you activate the **Wait until order finished** check box, you must wait until the parameter upload has been completed. If the parameter upload is not completed within the defined **timeout** (default 300 seconds), it is aborted and an error message is displayed.

Shall the parameter upload be time-controlled, activate the **Start time-controlled** check box (see chapter 3.3.15 Scheduler).

**Available buttons:**

**OK**

You can store the order in the (time-controlled) order file by selecting the **OK** button.

**Display**

With the button **Display** you can view detailed information for the selected terminal.

**Cancel**

The upload can be cancelled by selecting the **Cancel** button.

### 3.4.5.2 Parameter download

At first, source path and file for the parameters to download must be entered in a dialog. Afterwards the following screen is displayed:

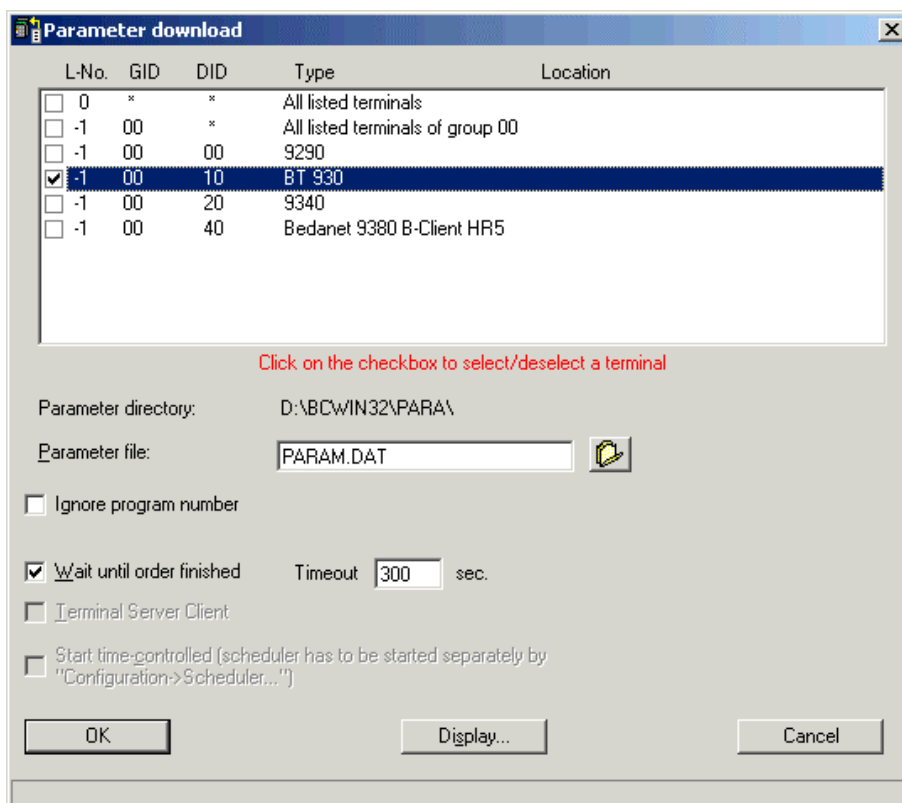


Figure 3-60: Parameter download

Select the terminals to which you want to download the parameters. In the **Parameter file** text box, you must enter the file containing the parameters you wish to send. This file must be stored as an **ASCII file**. **Each record must be terminated with CR/LF**. The first record of this file must contain the program number. The first three characters of the program number must agree with the addressed terminal's program number. This prevents a time recording terminal, for example, from receiving parameters of a terminal server. Wildcards (\*) are also permitted for the first three characters of the program number. In this case, **the user** must make sure that the parameter file matches the terminal concerned. The first three characters in the download records should be reserved for the addressing information (e.g. blanks). The terminal addresses entered in the first three positions during transfer. Existing addressing information in the download records (e.g. after an upload) is ignored.

---

### Example of a parameter file:

```
570-00-X-K08
>3B01Mailbox-Text 1
>3B02Mailbox-Text 2
>3B03Mailbox-Text 3
>3B04Mailbox-Text 4
>3B05Mailbox-Text 5
>3B06Mailbox-Text 6
>3B07Mailbox-Text 7
>3B08Mailbox-Text 8
>3F0013130011A0
>3F0113130011A1
>3F0213130011A2
>3I01Display-Info-Text 1
>3I02Display-Info-Text 2
>3I03Display-Info-Text 3
>3I04Display-Info-Text 4
```

The parameter file must be in the path that you specified in the **Path to parameter files** text box during installation.

If the check box **Ignore program number** is activated, then the first line of the parameter file, the program number, will be ignored and the file will be sent to the terminal even if the program number does not fit for the terminal.

If you activate the **Wait until order finished** check box, you must wait until the parameter download has been completed. If the parameter download could not be completed within the defined timeout (default 300 seconds), it is aborted and an error message is issued.

Shall the parameter download be time-controlled, activate the **Start time-controlled** check box (see chapter 3.3.15 Scheduler).

### Available buttons:

**OK**

You can store the order in the (time-controlled) order file by selecting the **OK** button.

**Display**

With the button **Display** you can view detailed information for the selected terminal.

**Cancel**

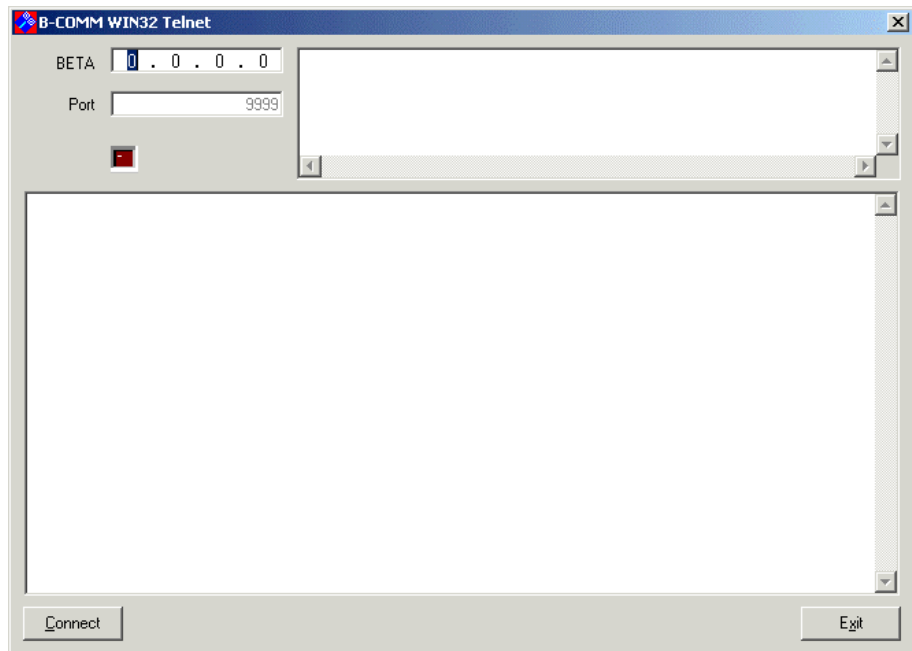
You can quit the download by selecting the **Cancel** button.

---

### 3.4.6. BETA setup by Telnet

With this menu item you can configure a BETA with a Telnet client, which is integrated into BCWIN. For this setup you must know the IP address of the BETA.

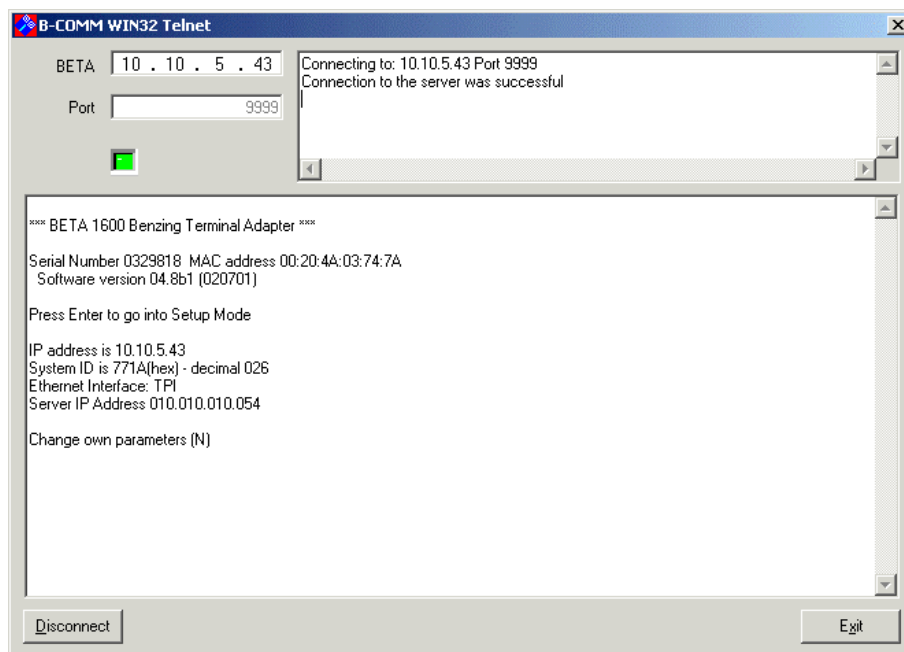
When selecting the menu item BETA SETUP BY TELNET... the following dialog is displayed:



**Figure 3-61: BETA setup by Telnet 1**

At first you have to enter the IP address of the BETA you want to configure into the field **BETA**. Then you can establish a connection to the BETA by pressing the **Connect** button. After that, a few messages from the BETA should be displayed within the output window and you should must press the Return key within 4 to 5 seconds in order to enter the configuration of the BETA.





**Figure 3-62: BETA setup by Telnet 2**

If you have done the complete configuration sequence, you are automatically disconnected from the BETA. Nevertheless you can disconnect any time by use of the **Disconnect** button.

**Auswählbare Buttons:**

**Connect** or **Disconnect**

With the button **Connect** a Telnet connection to the given BETA will be established, the button **Disconnect** will disconnect a Telnet connection.

**Exit**

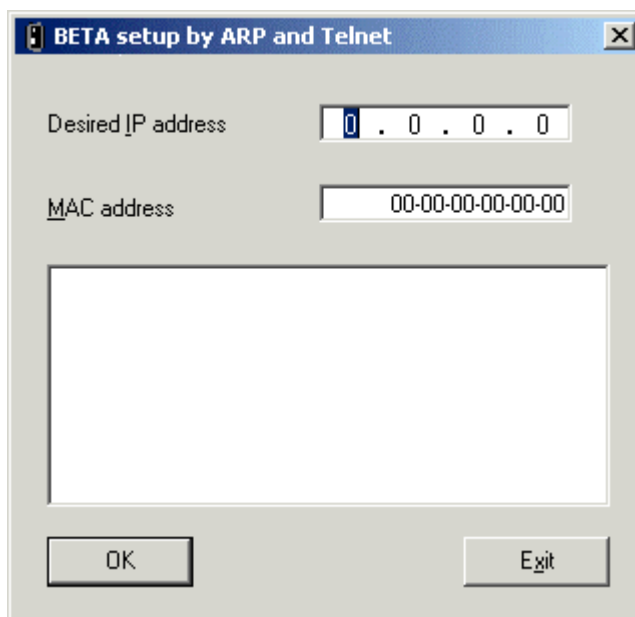
With the button **Exit** you can leave the Telnet client.

---

### 3.4.7. BETA setup by ARP and Telnet

With this menu item you can configure a BETA with a Telnet client, which is integrated into BCWIN. For this setup you must not know the IP address of the BETA, but you have to know the MAC address of the BETA.

When selecting the menu item BETA SETUP BY ARP AND TELNET... the following dialog is displayed:



**Figure 3-63: BETA setup by ARP and Telnet 1**

At first you have to enter the desired IP address of the BETA you wish to configure into the field **Desired IP address**. With this IP address the Telnet client tries to establish a connection later.

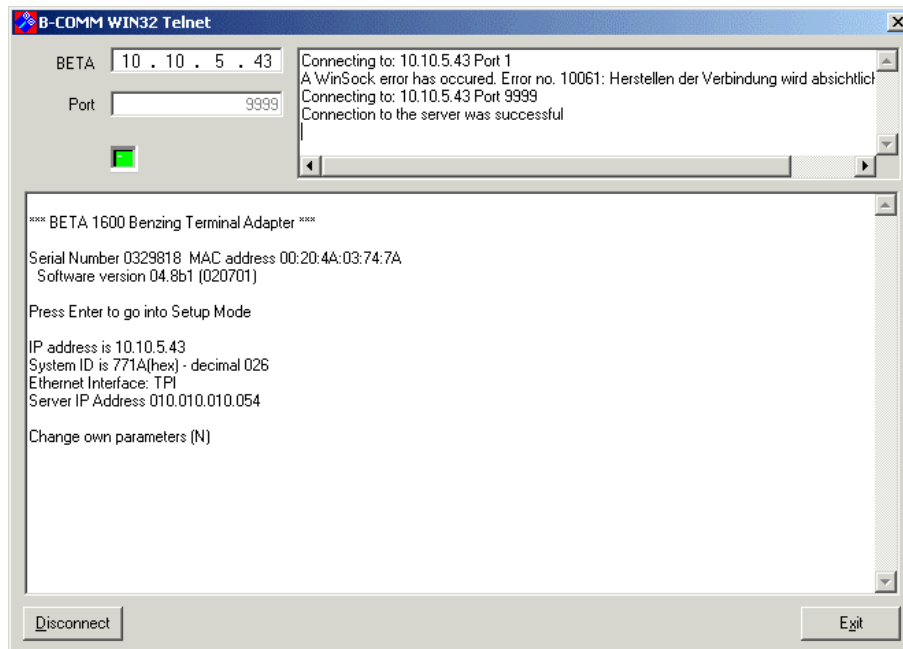
Then you have to enter the MAC address of the BETA into the field **MAC address** and then press the **OK** button.

Thereupon the application will execute the following command in the background:

```
arp -s <IP address> <MAC address>
```

Subsequently the Telnet client is called and tries to establish a connection to the BETA on port 1. The connection will fail, but that's not a problem!

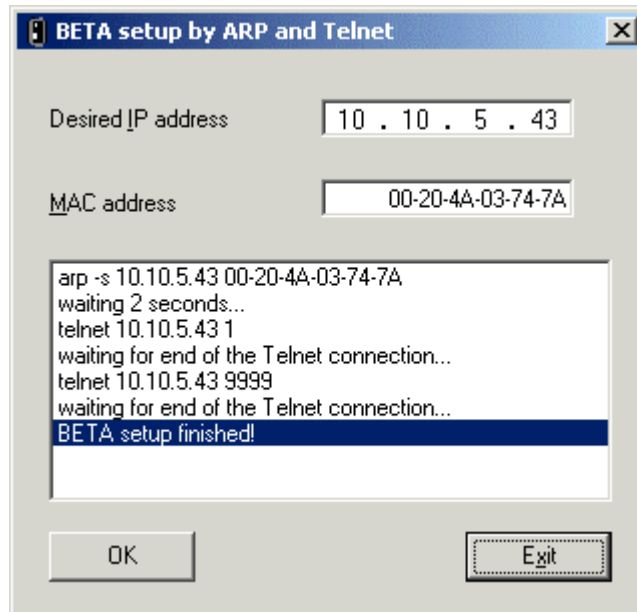
Now the application starts a new connection, this time on port 9999. This connection should now be successful and you have to press the Return key within 4 to 5 seconds in order to get into the configuration of the BETA. (see chapter 3.4.6 BETA setup by Telnet).



**Figure 3-64: BETA setup by ARP and Telnet 2**

---

In the output area of the first dialog you can see the sequence of the various steps of the configuration:



**Figure 3-65: BETA setup by ARP and Telnet 3**

**Available buttons:**

**OK**

With the button **OK** you start the sequence of the configuration. The output area is cleared with each button press.

**Exit**

With the **Exit** button you can leave the menu item BETA SETUP BY ARP AND TELNET....

### 3.5. Order

When you select the ORDER menu, you can choose between the DISPLAY ORDERS..., DELETE ALL ORDERS..., START COMMUNICATION... and STOP COMMUNICATION... menu items.

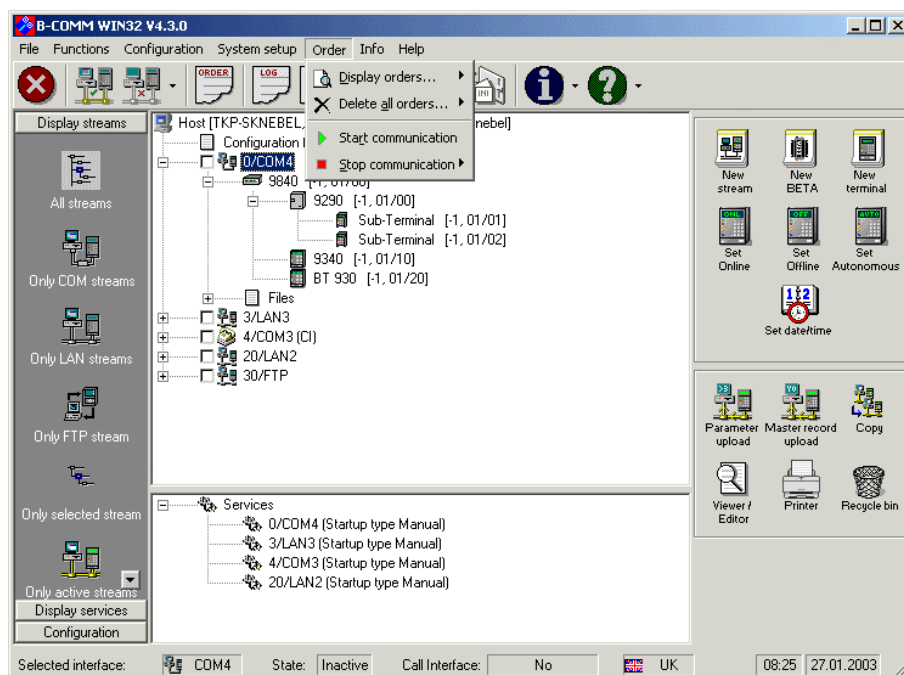


Figure 3-66: Menu Order

### 3.5.1. Display orders

You can display all (time-controlled) orders assigned to the selected communication stream. The following screen is displayed when you select the DISPLAY ORDERS... menu item and then one of the two submenu items NORMAL ORDERS... or TIME-CONTROLLED ORDERS...:

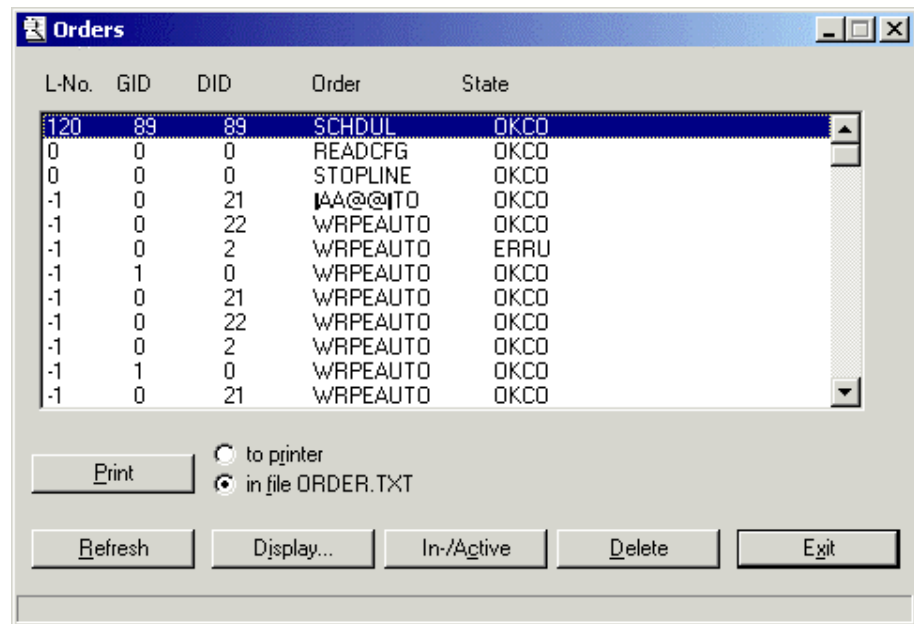


Figure 3-67: Display orders

Orders are displayed with the Line no., GID, DID, the order and its execution status. The meanings of negative line numbers are explained in 7.1 Terminal configuration file TERMDEF.<Stream>.

---

The following are defined as an „order“::

REDAAUTO	Get bookings in Autonomous mode
REDAOFF	Get bookings in Offline mode
WRPEAUTO	Master record download – father/son principle
WRPEHAAU	Master record download
ANALYSE	Logon terminals
SETMODE	Set operating mode
WRPARA	Parameter download from file
REPARA	Parameter upload to file
WRDATRA	Transmit transparent data from file
MESSAGE	Transmit display record
SWMEM	Control buffer memory
TIMEDAT	Set date/time
MODEMPAR	Set modem parameters
READTDF	Read terminal configuration file
READCFG	Read configuration file- BCOMM32.INI
STATUS	Request state of BCOMMIO or of a terminal
STOPLINE	Stop communication
MRUPLD	Master record upload

---

The following „execution status“ is defined:

No execution status	Order not yet executed
OKCO	Order executed correctly
ERRU	An error occurred during order execution. This may be because of a system error or a data transfer error
ERCO	Invalid parameters given on order transfer
OKME	Order executed correctly but terminal sent a system message
HOLD	Order is deactivated
RUN	Order is currently executed

A specific order can be highlighted with the cursor keys or by clicking with the mouse.

**Available buttons:**

**Refresh**

The screen can be updated by selecting the **Refresh** button.

**Display**

The highlighted order is explained in detail when you select the **Display** button (see also chapter 7.3 Order file ORDER.<Stream>).

**In-/Active**

By selecting the **In-/Active** button, you can reactivate orders that have already been executed and assigned an execution status. You can also deactivate orders that have not yet been executed.

**Delete**

You can delete the highlighted order by selecting the **Delete** button.

**Exit**

You can exit the DISPLAY ORDERS... menu item by selecting the **Exit** button.

**Print**

You can print out the order file formatted by selecting the **Print** button. The file is either printed directly to the system **printer** or in a **file** called ORDER.TXT.



---

### 3.5.2. Delete all orders

Each order that is added to the order file increases its size. You can delete the (time-controlled) order file by selecting the DELETE ALL ORDERS... menu item and then one of the submenus NORMAL ORDERS... or TIME-CONTROLLED ORDERS.... In doing so, the order file assigned with the selected stream is deleted.

#### Available buttons:

**Yes**

You can delete the order file by selecting the **Yes** button.

**Attention: Orders that have not yet been executed are also deleted.**

**No**

You can cancel the DELETE ALL ORDERS... menu item by selecting the **No** button.

---

### 3.5.3. Start communication

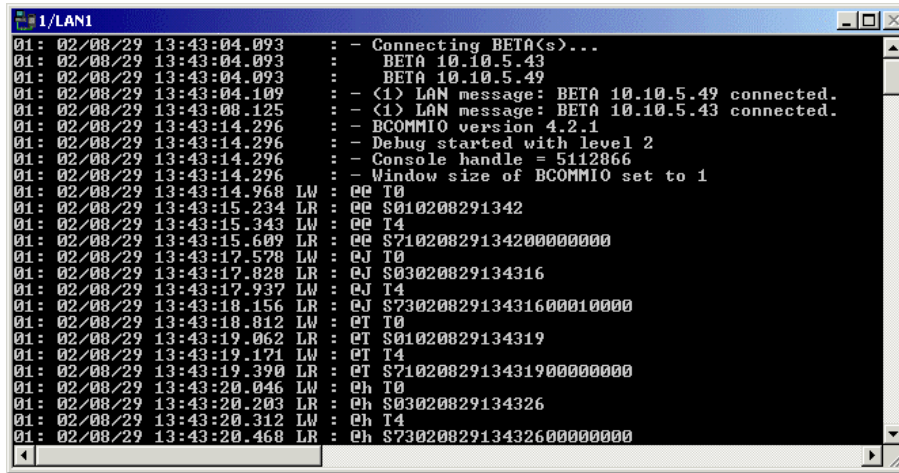
You can start communication by selecting the START COMMUNICATION menu item.

Starting communication results in the following sequence for the selected communication stream:

- The corresponding data transfer kernel BCOMMIO is started.
- The data transfer kernel BCOMMIO checks whether the corresponding communication stream is available. Communication is ended if the communication stream is not available.
- If you specified a call interface during configuration, its readiness to receive is checked. If you have additionally specified a program name, this program is started.
- All terminals in the online mode transfer existing records consecutively. This also holds true for all terminals in the offline mode provided they are not connected to a modem. In the case of a Bedas 9800 server, the memory files you have specified are selected. See chapter 3.3.2 Terminal logon.
- All data that has been fetched from the terminals is stored in a file.
- All orders are executed that are contained in the order file and which have not yet been executed. In doing so, the order file assigned to the communication stream concerned is used.
- When using a call interface, all data records are transferred to it.
- All data records that are transferred from the call interface to the data transfer kernel BCOMMIO are transferred to the corresponding terminals.

In the bottom part of the screen, the status of the displayed interface changes to active, e.g. Selected Interface: LAN8 State: Active Call Interface: No.

Via the task list you can change to the transmission kernel, if start option hidden is not set. A data transmission kernel is displayed as follows:



```
1/LAN1
01: 02/08/29 13:43:04.093 : - Connecting BETA(s)...
01: 02/08/29 13:43:04.093 : BETA 10.10.5.43
01: 02/08/29 13:43:04.093 : BETA 10.10.5.49
01: 02/08/29 13:43:04.109 : - (1) LAN message: BETA 10.10.5.49 connected.
01: 02/08/29 13:43:08.125 : - (1) LAN message: BETA 10.10.5.43 connected.
01: 02/08/29 13:43:14.296 : - BCOMMIO version 4.2.1
01: 02/08/29 13:43:14.296 : - Debug started with level 2
01: 02/08/29 13:43:14.296 : - Console handle = 5112866
01: 02/08/29 13:43:14.296 : - Window size of BCOMMIO set to 1
01: 02/08/29 13:43:14.968 LW : 00 T0
01: 02/08/29 13:43:15.234 LR : 00 S010208291342
01: 02/08/29 13:43:15.343 LW : 00 T4
01: 02/08/29 13:43:15.609 LR : 00 S71020829134200000000
01: 02/08/29 13:43:17.578 LW : 0J T0
01: 02/08/29 13:43:17.828 LR : 0J S03020829134316
01: 02/08/29 13:43:17.937 LW : 0J T4
01: 02/08/29 13:43:18.156 LR : 0J S7302082913431600010000
01: 02/08/29 13:43:18.812 LW : 0T T0
01: 02/08/29 13:43:19.062 LR : 0T S01020829134319
01: 02/08/29 13:43:19.171 LW : 0T T4
01: 02/08/29 13:43:19.390 LR : 0T S7102082913431900000000
01: 02/08/29 13:43:20.046 LW : 0h T0
01: 02/08/29 13:43:20.203 LR : 0h S03020829134326
01: 02/08/29 13:43:20.312 LW : 0h T4
01: 02/08/29 13:43:20.468 LR : 0h S7302082913432600000000
```

Figure 3-68: BCOMMIO window

Depending on the debug level several information is displayed. The window is scrollable horizontally and vertically, so you can see the last 250 outputs of BCOMMIO.

**Attention:** this window is not allowed to be closed manually. The window is closed automatically when the communication is stopped (see chapter 3.5.4 Stop communication).

---

### 3.5.4. Stop communication

You can end communication by selecting the STOP COMMUNICATION menu item.

When ending communication, the sequence for the selected communication stream is as follows:

- The corresponding data transfer kernel BCOMMIO is ended.
- When using a Bedas 9800 server, all memory file selections are cleared.

In the bottom part of the screen, the status of the displayed interface changes to Inactive, e.g. Selected Interface: LAN8 State: Inactive Call Interface: No.

With this menu item you can either stop the currently selected stream or all active streams. In the latter case really all active streams are stopped, including COM and LAN streams and the FTP stream.

### 3.6. Info

When you select the INFO menu, you can choose between the ABOUT B-COMM..., SYSTEM INFORMATION..., VERSION INFORMATIONS, TERMINAL STATUS and LICENCE REGISTRATION... menu items.

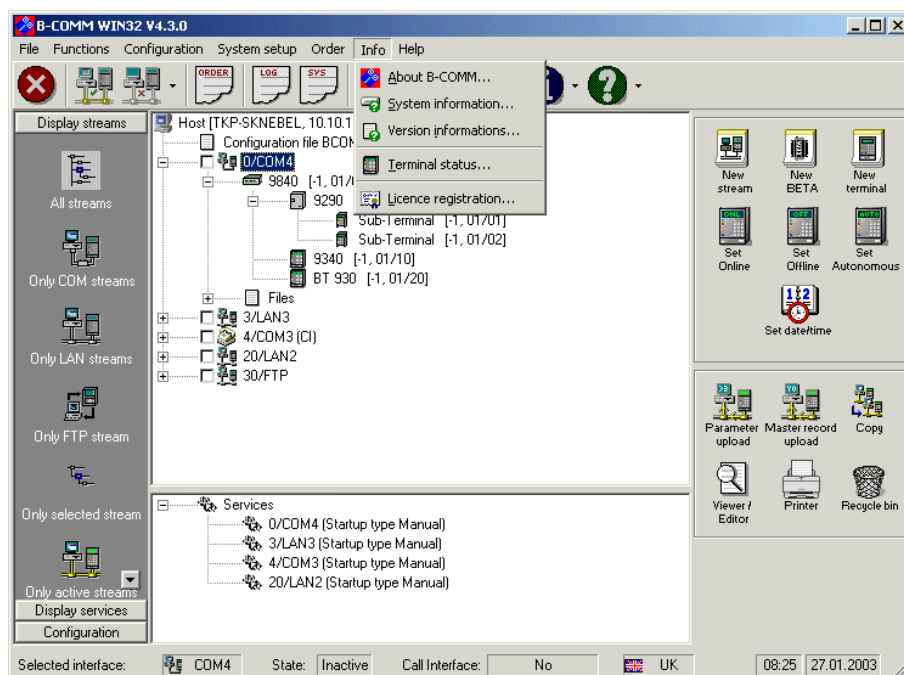


Figure 3-69: Menu Info

---

### 3.6.1. About B-COMM

When you select the ABOUT B-COMM... menu item, the version and licence information are displayed. These are needed when making inquiries, for example.

### 3.6.2. System information

The Windows version and free memory etc. are displayed when you select the SYSTEM INFORMATION... menu item.

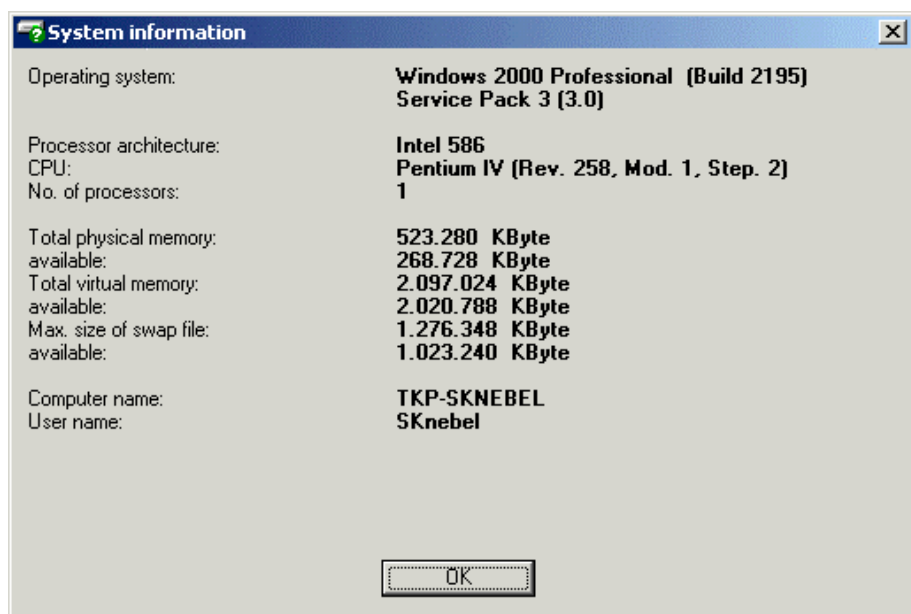


Figure 3-70: System information

---

### 3.6.3. Version informations

Version informations concerning the various modules of B-COMM WIN32 are displayed when you select the VERSION INFORMATIONS... menu item.

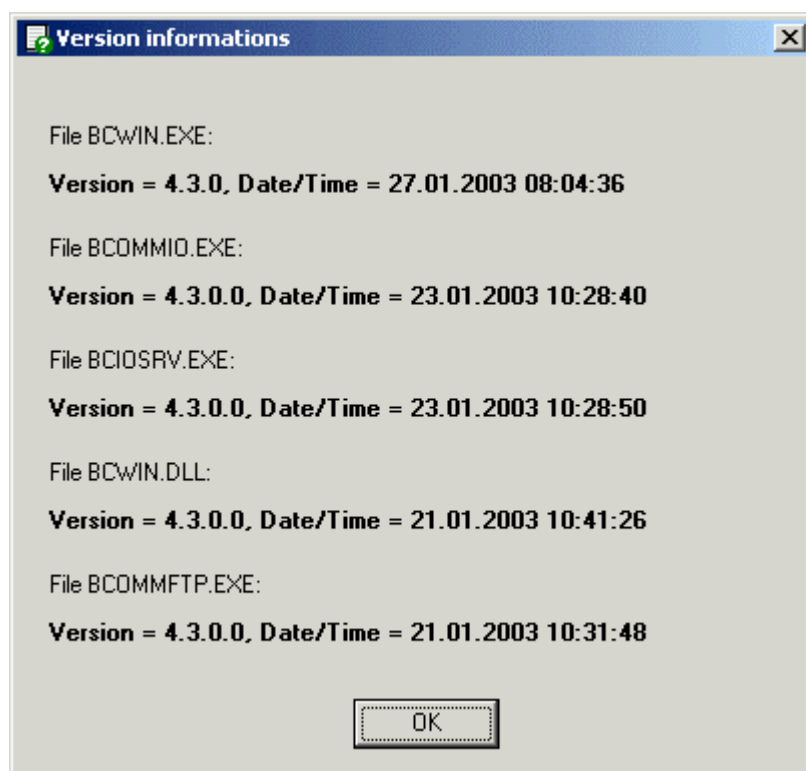


Figure 3-71: Version informations

### 3.6.4. Terminal status

After selecting the menu item TERMINAL STATUS... the external program TERMSTAT is launched to display the stream and terminal status (see chapter 5 The application TERMSTAT).

### 3.6.5. Licence registration

You can do licence registration by selecting the LICENCE REGISTRATION... menu item. To do this, Kaba Benzing must reassign a licence number and a code. The following screen, for example, is displayed when you select the LICENCE REGISTRATION... menu item:

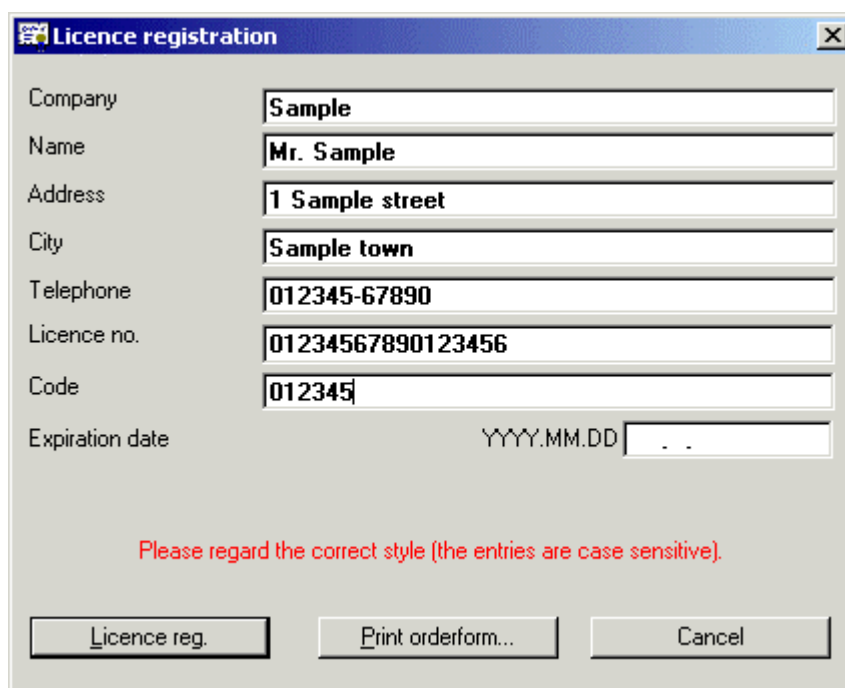
A screenshot of a Windows-style dialog box titled "Licence registration". It contains several text input fields for registration details. The fields are labeled on the left: Company, Name, Address, City, Telephone, Licence no., Code, and Expiration date. The corresponding values entered in the fields are: Sample, Mr. Sample, 1 Sample street, Sample town, 012345-67890, 01234567890123456, 012345, and YYYY.MM.DD . . . . Below the fields, there is a red text message: "Please regard the correct style (the entries are case sensitive).". At the bottom of the dialog, there are three buttons: "Licence reg.", "Print orderform...", and "Cancel".

Figure 3-72: Licence registration

#### Available buttons:

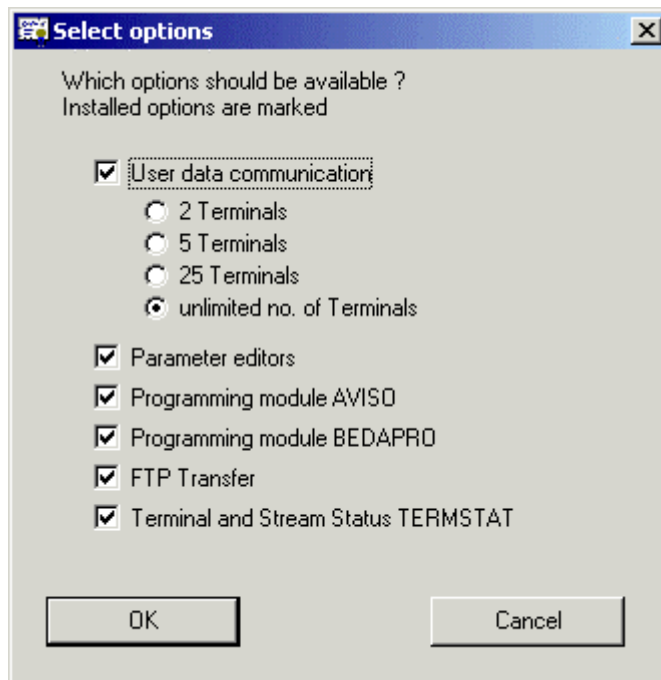
##### **Licence reg.**

When you select the **Licence reg.** button you can execute registration after the licence no. and the code has been entered. The message „Authorization successful“ is displayed after successful registration. Please shut down now any active communication streams and restart them to accept the new licence data.

##### **Print Orderform**

You can print a orderform on your system printer by selecting the **Print Orderform** button.





**Figure 3-73: Select options**

Activate the desired options (already installed options are checked) and press the button **OK** to start the printout.

**Cancel**

You can cancel the LICENCE REGISTRATION... menu item by selecting the **Cancel** button.

## 3.7. Help

After selecting the HELP menu, you can choose between MANUAL... and TOOLTIP HELP.... menu items.

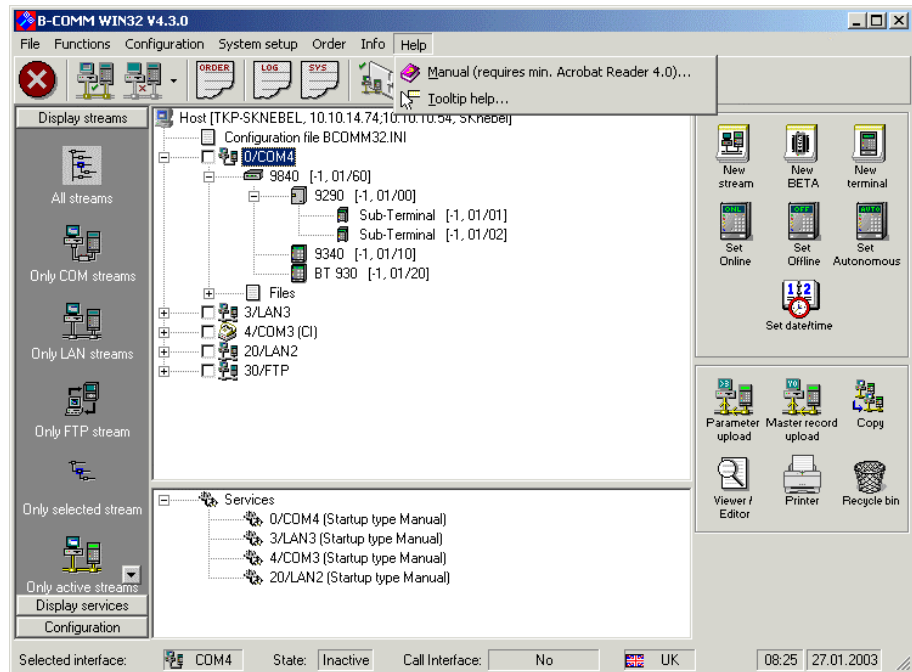


Figure 3-74: Menu Help

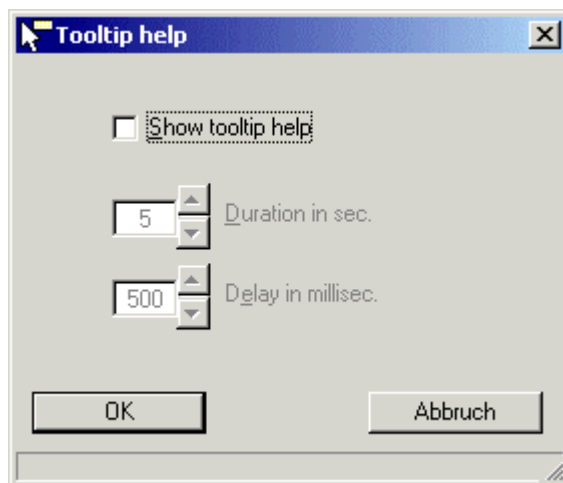
### 3.7.1. Manual

By selecting the MANUAL... menu item this manual is displayed.

### 3.7.2. Tooltip help

With the menu item TOOLTIP HELP... you can switch on/off the so called tooltip help of B-COMM and setup some parameters for the tooltip help. A tooltip help is a little help window that appears when the mouse stays over a text box etc. for a while and disappears after a configurable time.

After selecting this menu item the following screen is shown:



**Figure 3-75: Tooltip help**

With the check box **Show tooltip help** you can switch on/off the tooltip help. If it is switched on, the behaviour can be configured with the text boxes **Duration in sec.** and **Delay in msec.**.

With the text box **Duration in sec.** you can specify how long the help is shown until it disappears automatically.

With the text box **Delay in msec.** you can specify how long the mouse must stay over a text box etc. before the little help window is shown.

---

#### **Auswählbare Buttons:**

**OK**

With the button **OK** the settings are stored.

**Cancel**

You can cancel the TOOL TIP HELP... menu item by selecting the **Cancel** button.

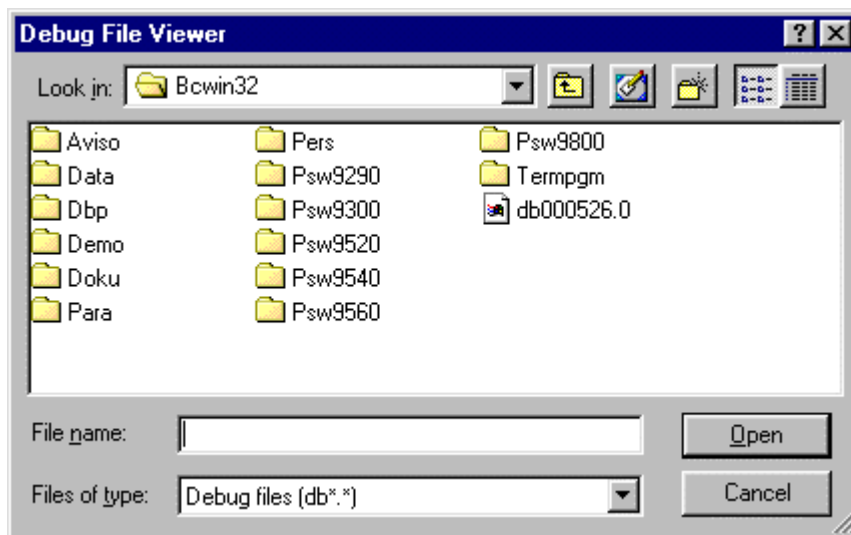
---

## 4. The application SHOWDBG

This program is used for the comfortable analysis of the debug files which are written by BCOMMIO/BCOMMFTP (see chapter 3.3.10 File definitions). It can be launched either with the menu item FILE – DEBUG FILE(S)... or by the Windows start menu: START – PROGRAM FILES - B-COMM WIN32 – DEBUG VIEWER.

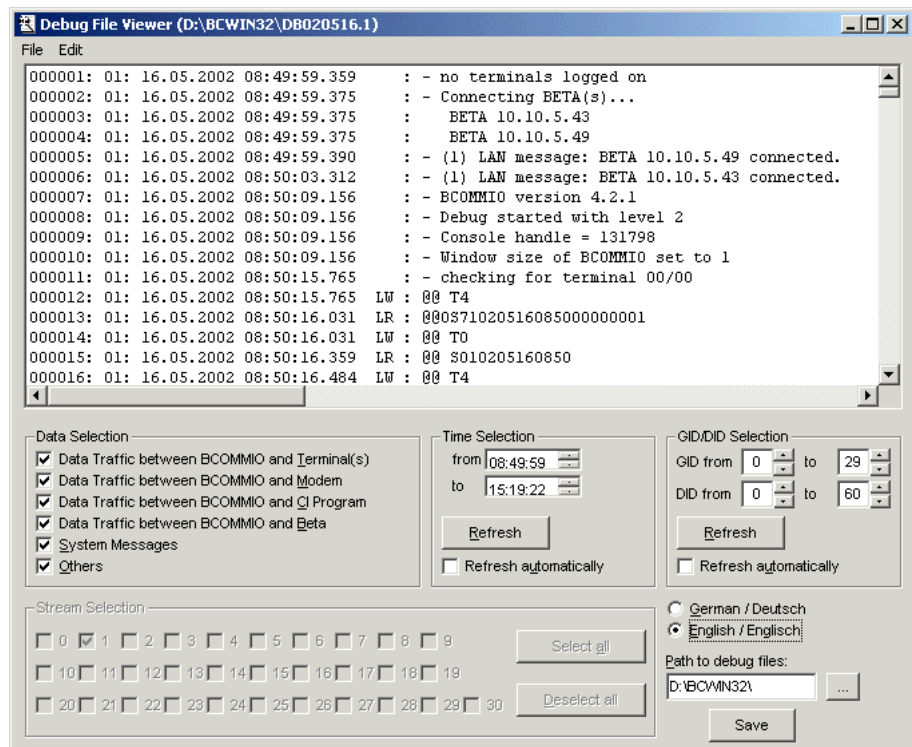
For the program there exists a separate setup, so that you can install it on a separate computer. SHOWDBG is then completely independent from a B-COMM installation.

After the program start you first have to select a debug file by use of the standard Windows file dialog.



**Figure 4-1: Select debug file**

After that the program appears with the following screen:



**Figure 4-2: Main window of Debug File Viewer**

In the upper list the complete debug file is shown. Each entry has the following format:

<line>: <stream>: <date> <time> <tag>: <data record>

where

<line>	a continuous line number
<Kanal>	number of the communication stream (this number is only useful if you didn't specify a stream specific debug file but you're analysing a file which contains information from all configured streams)
<date>	date of the file creation
<Uhrzeit>	time at which BCOMMIO wrote the entry
<tag>	this tag consists of max. 3 characters/digits. The following tags are possible:

---

<b>SR</b> ( <b>S</b> erial <b>R</b> ead)	Data record from a terminal to the PC with serial communication
<b>SW</b> ( <b>S</b> erial <b>W</b> rite)	Data record from the PC to a terminal with serial communication
<b>LR</b> ( <b>L</b> AN <b>R</b> ead)	Data record from a terminal to the PC with network communication
<b>LW</b> ( <b>L</b> AN <b>W</b> rite)	Data record from the PC to a terminal with network communication
<b>BR</b> ( <b>B</b> eta <b>R</b> ead)	UDP data record from a Beta to BCOMMIO
<b>BW</b> ( <b>B</b> eta <b>W</b> rite)	UDP data record from BCOMMIO to a Beta
<b>MR</b> ( <b>M</b> odem <b>R</b> ead)	data from a modem to the PC
<b>MW</b> ( <b>M</b> odem <b>W</b> rite)	data from the PC to a modem
<b>CR</b> ( <b>C</b> all Interface <b>R</b> ead)	data record or command from CI program to BCOMMIO
<b>CW</b> ( <b>C</b> all Interface <b>W</b> rite)	Data record or answer from BCOMMIO to the CI program
<b>SM1</b> ( <b>S</b> ystem <b>M</b> essage Teil <b>1</b> )	1st part of a system message
<b>SM2</b> ( <b>S</b> ystem <b>M</b> essage Teil <b>2</b> )	2nd part of a system message
<b>RR</b> ( <b>R</b> eset <b>R</b> ead)	This entry only appears when BCOMMIO has shut down because of a missing answer from the CI program and then reads out its own data record
<b>Empty</b>	All other outputs of BCOMMIO

By use of the various check boxes in the **Data selection** group you can hide/show entries to increase readability or simply hide uninteresting entries. The list is updated immediately.

---

By use of the **Time selection** entry fields the number of visible entries can be decreased or increased by setting the times in the **from** and **to** entry fields appropriate to your needs. When changing these two entry fields, the upper list is by default not updated immediately but only when pressing the button **Refresh**. By activating the check box **Refresh automatically** you can force an immediate refresh at any time. This is only advisable when analysing relatively small debug files.

With the **GID/DID selection** you can show/hide data records from or to specific terminals by using the entry fields **GID from, to** and/or **DID from, to**. When changing these four entry fields, the upper list is by default not updated immediately but only when pressing the button **Refresh**. By activating the check box **Refresh automatically** you can force an immediate refresh at any time. This is only advisable when analysing relatively small debug files.

When analysing a debug file which is not stream specific you can switch on/off the display of a specific stream by use of the check boxes of the **Stream selection** group.

By use of the option buttons **Deutsch/German** and **Englisch/English** you can change the language of the application.

If the debug files to analyze are not located in the default directory (the B-COMM directory), you can set another path with the **Path to debug files** text box and save that location with the **Save** button in the configuration file of SHOWDBG.

When selecting the FILE menu you can selecte between the following menu items: OPEN..., SAVE AS..., PRINT..., DELETE... und EXIT zur Verfügung.

OPEN...:

You can open a new debug file to analyze.

SAVE AS...:

By use of this menu item you can save exactly those entries, which are currently visible in the upper list, into a separate file.

PRINT...:

By use of this menu item you can print exactly those entries, which are currently visible in the upper list.

DELETE...:

You can delete any debug file.

EXIT:

Ends the SHOWDBG application.

The menu item EDIT contains some functions to search specific strings within the debug file.



## 5. The application TERMSTAT

This application is used to display the state of the configured communication streams and the connected terminals graphically. Therefore TERMSTAT communicates with one or more streams via UDP data records. **As a result, TERMSTAT does not work on PCs, on which TCP/IP is not installed.** A separate setup is available for TERMSTAT, so it must not be installed on the same PC as B-COMM but can be installed on any PC in the network.

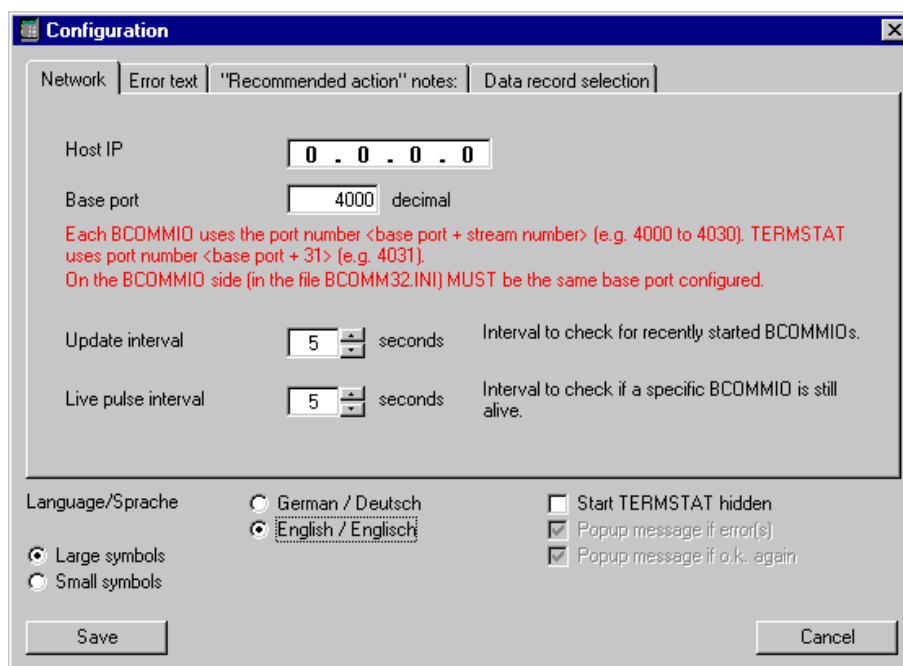
The application knows two optional commandline parameters:

**TERMSTAT.EXE [-C<Client>] [-P<Port number>]**

If none of the parameters is given, TERMSTAT observes the Default client, the port number will be determined from the section [Configuration], key BasePort= of the TERMSTAT.INI configuration file.

If only the parameter -C<Client> is given, TERMSTAT observes the streams and terminals of the client <Client>. The port number is then either determined from the section [Client], key BasePort=, as far as that section was created manually, or from the section [Configuration], key BasePort=.

After the first start of TERMSTAT the following configuratiuon dialog is shown:



**Figure 5-1: TermStat configuration**

In this dialog you at first have to enter the **host IP** address of the PC, on which the corresponding stream is active. Only when you change the base port in the configuration file BCOMM32.INI (see 7.2 Configuration file BCOMM32.INI) you have to adjust the **base port** in this dialog too. Each of the 31 possible streams (30 BCOMMIOs + 1 BCOMMFTP) uses this port

number plus an offset. This offset is the number of the stream. The streams 2 and 5 shown in the figure below use the ports 4002 and 4005. The program TERMSTAT itself uses the port number base port + 31, in this case 4031.

With the field **Update interval** you can specify the interval in which TERMSTAT checks whether another communication stream has started or if a stream has shut down.

With the field **Live pulse interval** you can specify the interval in which TERMSTAT sends so called "live pulses" to the active streams to check, whether they are still running.

When you activate the **Start TERMSTAT hidden** field, then TERMSTAT is only represented by an icon in the system tray when you restart it. This icon shows the stream and terminal status with the following icons:



currently no stream is active



at least one stream is active, currently no errors



at least one stream is active, at least one error occurred



you have no licence to display the terminals

By use of the fields **Popup message if error(s)** and **Popup message if o.k. again** you can specify whether the user should be informed by a popup message box in case of an error or in case a stream is o.k. again.

Using the fields **Large symbols** and **Small symbols**, you can switch the icon size of the terminals in the main window of TERMSTAT.

With the button **Save** the settings are written into the configuration file TERMSTAT.INI and the program window is shown:

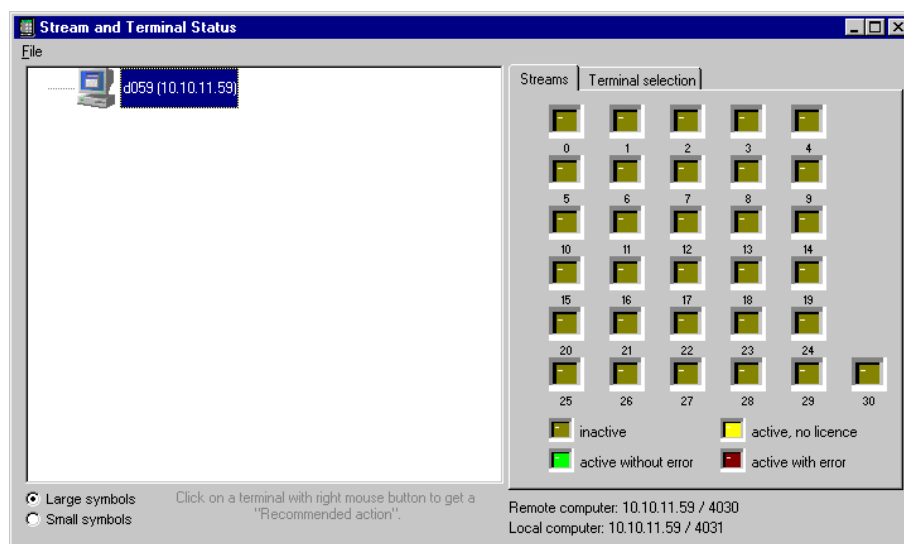
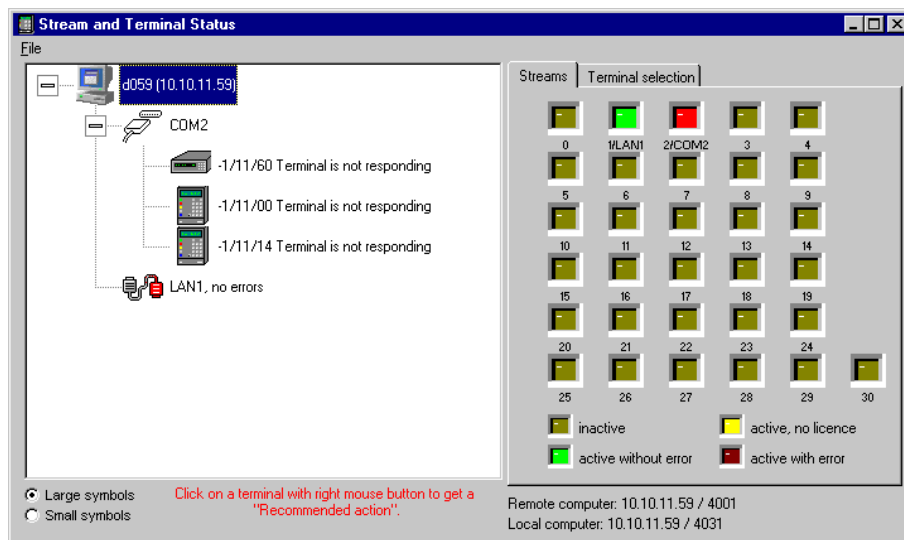


Figure 5-2: Main window of TermStat

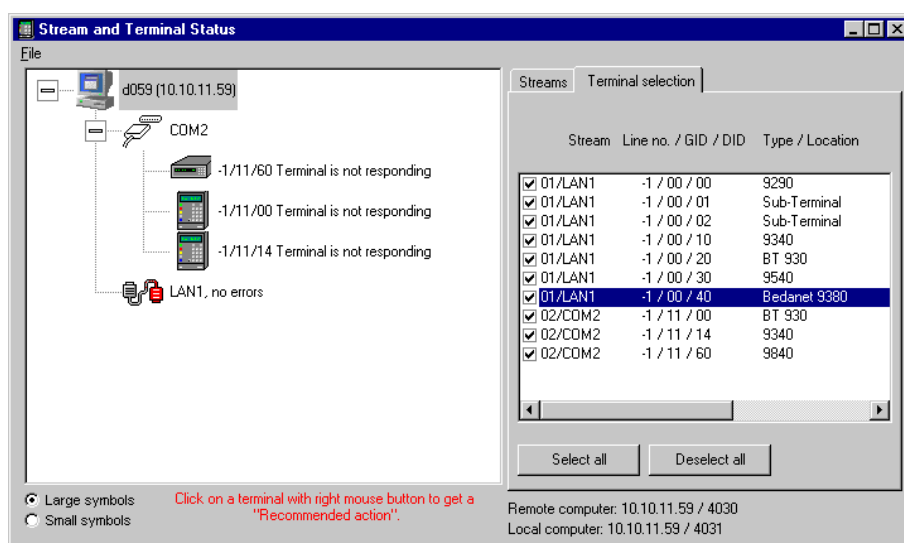
In the left part of the window all currently active streams are shown, as well as those terminals, which are currently not available or have an error. In the right part each stream is represented by a LED. In the above figure no communication stream is currently active. By use of the fields **Large symbols** and **Small symbols** you can switch the icon size of the terminals temporarily. If you want this setting to be permanent, you have to go to the configuration dialog.

The next figure shows the communication stream 1 and 2 with the interfaces LAN1 and COM2, where stream 2 reports errors:



**Figure 5-3: Streams and Terminals in TermStat**

If you select the register tab **Terminal selection** in the right part of the screen you can see all terminals which are logged on at the stream. Here you can now deselect those terminals, whose status you don't want to see.



---

**Figure 5-4: Logged-on terminals in TermStat**

---

## 5.1. Configuration

Via the menu DATEI you can reach the menu item CONFIGURATION... we talked about at the start of this chapter.

On the other tabs of this dialog you can change the following settings:

Register tab **Error text**: here you can enter your own text for the six possible error messages of TERMSTAT. These texts are shown beside the symbol of the terminal, in case it reports an error.

Register tab „**Recommended action**“ **text**: these texts will be presented to the user, when he clicks the symbol of a terminal with the right mouse button. Here you can enter tips for the user, what he can do in case of error situations.

Register tab **Data record selection**: here you can specify, if a terminal should be displayed with error in case of alarm or O2 records.

---

## 5.2. Data records from TERMSTAT to BCOMMIO

- **TS\_LOGON**

This record is sent to all possible port numbers (base port + 0 to base port + 30) in a configurable interval to check, whether a communication stream was started. If a **TS\_LOGON\_ACK\_<stream name>\_<stream no.>\_<lic>\_<host name>** was received as an answer to a **TS\_LOGON**, no more **TS\_LOGON** is sent to this port.

- **TS\_LOGOFF\_<stream name>**

This record is sent by TERMSTAT to the appropriate stream, when the application ends. It is answered by the stream with **TS\_LOGOFF\_ACK\_<stream name>**.

- **TS\_LIVE\_PULSE\_<stream name>**

This record is sent by TERMSTAT to the appropriate stream to check if it is still active. It is answered by the stream with **TS\_LIVE\_ACK\_<stream name>**.

---

### 5.3. Data records from BCOMMIO to TERMSTAT

- as an answer to **TS\_LOGON**:

**TS\_LOGON\_ACK\_<stream name>\_<stream no.>\_<lic>\_<host name>**

The stream no. Is the real stream number (0 to 30) under. B-COMM WIN32, under B-COMM UNIX and B-COMM OS/400 „FF“ is entered here and the stream numbers have to be configured into the configuration file TERMSTAT.INI.

- as an answer to **TS\_LIVE\_PULSE\_<stream name>**:

**TS\_LIVE\_ACK\_<stream name>**

- when a terminal is reported as „not available“:

**<stream name>\_<“000”><Beta IP><line no.><GID><DID>  
<type>\_<location>**

- when a terminal is reported as „o.k. again“:

**<stream name>\_<“001”><Beta IP> <line no.><GID><DID>**

- when BCOMMIO was started (BCOMMIO\_UP):

**<stream name>\_<“002”>**

- when BCOMMIO has shut down (BCOMMIO\_DOWN):

**<stream name>\_<“003”>**

- when BCOMMIO reads the terminal configuration file:

**<stream name>\_<“004”>**

- when a terminal is reported as „Full with master records“:

**<stream name>\_<“005”><Beta IP> <line no.><GID><DID>  
<type>\_<location>**

- when a „Door opened too long“ is reported by a terminal:

**<stream name>\_<“006”><Beta IP> <line no.><GID><DID>  
<type>\_<location>**

- when a „Door broken open“ is reported by a terminal:

**<stream name>\_<“007”><Beta IP> <line no.><GID><DID>  
<type>\_<location>**

- when a „Door closed“ is reported by a terminal:

**<stream name>\_<“008”><Beta IP> <line no.><GID><DID>  
<type>\_<location>**

- when „Start of sabotage alarm“ is reported by a terminal:

**<stream name>\_<“009”><Beta IP> <line no.><GID><DID>  
<type>\_<location>**

- when „End of sabotage alarm“ is reported by a terminal:

**<stream name>\_<“010”><Beta IP> <line no.><GID><DID>  
<type>\_<location>**

- when a BETA is reported as „not available“:

**<stream name>\_<“011”><Beta IP>**

- When a BETA is reported as „o.k. again“:

**<stream name>\_<“012”> <Beta IP>**

- When a terminal definition is send:

**<stream name>\_<“013”><line no.><GID><DID>  
<type>\_<location>**

<b>&lt;stream name&gt;</b>	=	„COM1“ to „COM8“ or „LAN1“ to LAN30 “ or “FTP”
<b>&lt;stream no.&gt;</b>	=	number of the stream („00“ to „29“) or „FF“
<b>&lt;lic&gt;</b>	=	“0” if there is no licence for TERMSTAT, otherwise “1”
<b>&lt;host name&gt;</b>	=	name of the host PC
<b>&lt;line no.&gt;</b>	=	line number of the terminal („FFF“ or „001“ to „999“)
<b>&lt;GID&gt;</b>	=	„00“ to „29“ (corresponds to GID 00 to 29)
<b>&lt;DID&gt;</b>	=	„00“ to „60 “ (corresponds to DID 00 to 60)
<b>&lt;type&gt;</b>	=	the terminal type stored in the TERMDEF file
<b>&lt;location&gt;</b>	=	the location stored in the TERMDEF file
<b>&lt;Beta IP&gt;</b>	=	IP address of a Beta (000.000.000.000 = no Beta)



---

## 5.4. Communication between BCOMMIO and TERMSTAT

### 5.4.1. Data records of TERMSTAT

The TERMSTAT program sends the logon data record **TS\_LOGON** to the addresses "BasePort" + stream number in the interval "LogonInterval". If it receives an answer **TS\_LOGON\_ACK\_<stream name>\_<stream no.>\_<host name>**, to this address no more logon record will be send. When TERMSTAT ends, the data record **TS\_LOGOFF\_<stream name>** is send, which must be answered with **TS\_LOGOFF\_ACK\_<stream name>** by BCOMMIO. Additionally TERMSTAT sends live pulse data records **TS\_LIVE\_PULSE\_<stream name>** to BCOMMIO in a configurable interval ("LiveInterval") to check, if BCOMMIO is still active. These records must be answered with **TS\_LIVE\_ACK\_<stream name>** by BCOMMIO.

### 5.4.2. Data records of BCOMMIO

If BCOMMIO receives a **TS\_LOGON**, it answers with **TS\_LOGON\_ACK\_<stream name>\_<stream no.>\_<host name>** and then sends data about the connected terminals by use of data records of the following type to TERMSTAT:

**<stream name>\_<"Ø13"><line no.><GID><DID><type>\_<location>**

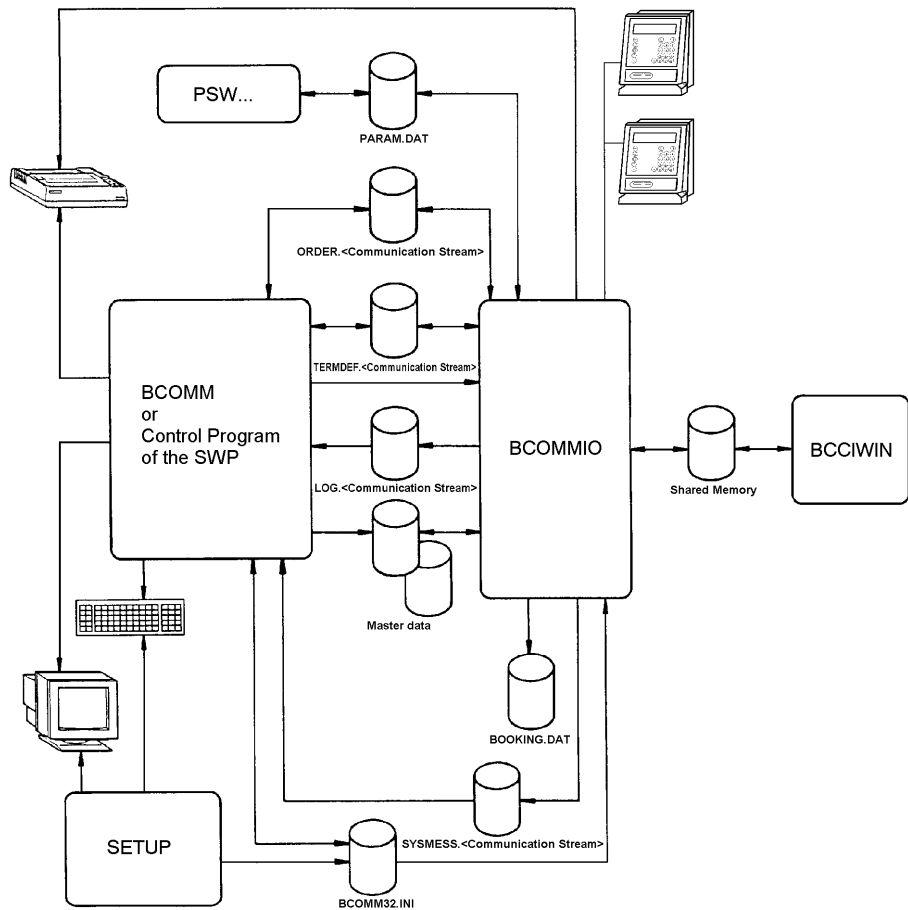
With these information TERMSTAT can display the terminals graphically.

If TERMSTAT receives a data record **<stream name>\_<"ØØ4">**, BCOMMIO has read the terminal configuration file again. TERMSTAT should then clear all information of that stream because it will then receive the new terminal configuration from BCOMMIO by use of data records of type **<stream name>\_<"Ø13">....**

BCOMMIO answers the live pulse records **TS\_LIVE\_PULSE\_<stream name>** sent by TERMSTAT in intervals with **TS\_LIVE\_ACK\_<stream name>**, and a logoff record **TS\_LOGOFF\_<stream name>** with **TS\_LOGOFF\_ACK\_<stream name>**.

## 6. The B-COMM WIN32 file system

Files are used to transfer all data and commands and to make the system configuration (terminal, computer) available. The following figure elucidates the relationships between the individual programs and the files used.



**Figure 6-1: File system of B-COMM WIN32**

Each interface is managed by its own data transfer kernel BCOMMIO.

---

## 7. Data Transfer Kernels BCOMMIO and BCOMMFTP

The BCOMMIO/BCIOSRV and BCOMMFTP programs represent the kernel of the B-COMM WIN32 program package. BCOMMIO/BCOMMFTP require no keyboard inputs. The BCOMMIO data transfer kernel handles communication with the terminals in conformity with Kaba Benzing's BPA/9 protocol and stores received data in the files intended for this purpose. For each configured stream there is one BCOMMIO active. The BCOMMFTP is active only in the system and serves for the FTP transfer of master record, parameter records and transparent data records to Bedanet (B-Net) terminals. The BCOMMIO and BCOMMFTP communication programs can be called up by application programs that provide the corresponding order and configuration files. The start of BCOMMIO/BCOMMFTP can be executed by the application of the SWP (Software Partner) or via other Windows mechanism's, e.g. autostart or job scheduler.

The FTP kernel BCOMMFTP has only one commandline parameter:

### **BCOMMFTP [-C<Client>]**

where <Client> is the name of the client whose BCOMMFTP should be started. If the parameter is not given, the BCOMMFTP of the Default client will be started.

Calling up "normal" BCOMMIO:

### **BCOMMIO [<Client>,<Interface> [optional parameters]]**

The following optional parameters are available:

-ddebuglevel	Debug level = 0 to 9 0 = no output within BCOMMIO window 9 = very detailed output within BCOMMIO window
-wwindowsize	window size 0, 1 or 6 0 = hidden (invisible) 1 = standard (visible) 6 = minimized (in task bar)
-t	shows a little traffic light in the system tray (see chapter 3.3.6 Initialize communication stream)
-c0	disables the Close button of the BCOMMIO window and removes the CLOSE menu item from the system menu

If the parameter <Client> is not given, the BCOMMIO of the default client will be started. As a interface COM1 to COM8 and LAN1 to LAN30 can be entered. BCOMMIO is active as long as the command STOPLINE occurs.

---

Calling up the transfer kernel as a service (BCIOSRV.EXE):

**BCIOSRV –install [<Client>,<Interface> or  
BCIOSRV –install [<Client>,<Interface> -A**

Installs the stream for <Interface> of the client <Client> as a service with startup type automatic, i.e. the transfer kernel is started automatically with each start of the PC.

Example: BCIOSRV –install Test,LAN1 -A

**BCIOSRV –install [<Client>,<Interface> -M**

Installs the stream for <Interface> of client <Client> as a service with startup type manual, i.e. the transfer kernel has to be started manually either from within the Windows service manager or by use of the menu item CONFIGURATION->SERVICES from within BCWIN.

Example: BCIOSRV –install Test,LAN1 -M

**BCIOSRV –remove [<Client>,<Interface>**

With this command the stream for <Interface> of client <Client> is removed from the service manager.

Example: BCIOSRV –remove Test,LAN1

**BCIOSRV –start [<Client>,<Interface>**

With this command you can start the service for stream <Interface> of client <Client>.

Example: BCIOSRV –start Test,LAN1

**BCIOSRV –stop [<Client>,<Interface>**

With this command you can stop the service for stream <Interface> of client <Client>.

Example: BCIOSRV –stop Test,LAN1

All the above commands can also be executed by use of the menu item CONFIGURATION->SERVICES in BCWIN.

---

After calling BCOMMIO the following procedures are running:

- The data transmission kernel checks whether another process is accessing the interface. If another process is accessing the interface, the communication is quit.
- When a call interface has been entered during configuration, it is called now. The inter process communication between BCOMMIO and the CI process is realized via a shared memory area. During the start BCOMMIO is writing the request `:STATUS:` in this area. During a timeout the data transmission kernel BCOMMIO is waiting until `:OK` is read from this area. When `:OK:` is read from the shared memory area, this is the response for BCOMMIO. That means, the CI program is ready to receive and the transmission of terminal data can be started. When BCOMMIO gets 3 times no response during the timeout, BCOMMIO is cancelled and a system error message is generated.
- All terminals in the online mode transfer existing records consecutively. This also holds true for all terminals in the offline mode provided they are not connected to a modem. When using a Bedas 9Ø xx server, the memory files that you have specified are selected.
- All data that has been fetched from the terminals is stored in a file.
- All orders that are in the order file and have not been executed are executed. In doing so, use is made of the order file that is assigned to the communication stream concerned.
- When using a call interface, all records are transferred to it. The shared memory area is used for this purpose.
- For synchronization and safety between BCOMMIO and the CI process a simple handshake procedure is used. The CI program must send a response after data records are received from BCOMMIO. The CI process can send `:ACK:` or a data record as a response. Normally, a data record is send as a online response to the terminal. Additionally the CI process can also send data asynchronous to BCOMMIO, for example a configuration record to any terminal which is defined in BCOMMIO. The data record length must be 255 characters. If not so, the data record must be completed by `ØxØØ` characters to the required length. The function `WriteCISharedMem` in the library `BCWIN.DLL` is doing this automatically. This asynchronous data records are confirmed by BCOMMIO with `:ACK:`.

The following example shows how data could be transferred between the BCOMMIO data transfer kernel and a call interface.

---

Data records that BCOMMIO is writing to  
the shared memory area.

:STATUS:

IO:@@ B11950819120001234567891234

:ACK:

IO:@A0B11950819120101234567894321

IO:@B B21950819120201234567894555

Data records that a call interface is writing  
to the shared memory area.

:OK:

@@ R1Saldo +2:34

@@ >3F0003130021A0

:ACK

@B R1Saldo +4:18

---

**BCOMMIO needs the following files:**

- the terminal configuration file TERMDEF.<Stream>
- the configuration file BCOMM32.INI
- the order file ORDER.<Stream>
- the LAN configuration file BETASYS.<Stream>

**BCOMMIO generates and edits the following files:**

- the Log file Log.<stream>
- the system message file SYSMESS.<Stream>
- the terminal configuration file TERMDEF.<Stream>
- the order file ORDER.<Stream>
- the parameter file PARAM.DAT
- the booking file BOOKING.DAT or BOOKING.<Stream>
- the son file SON<GID><DID>.REC

The files with the extension <Stream> are files which are assigned to a specified communication stream. The assignment between a communication stream and a interface is defined in the configuration file BCOMM32.INI (see chapter 7.2 Configuration file BCOMM32.INI). If, for example, the interface **COM2** is defined as communication stream Ø, all files with the extension Ø are assigned to the interface **COM2**. So the assigned order file has the name ORDER.Ø.

**General notes about files which are used by B-COMM WIN32:**

B-COMM WIN32 does not delete the files explained in the following chapters (exception: deleting a communication stream). The file management must be done by the SWP or the user. Some files, e.g. the system message file, are increasing very fast during operation. We recommend the observation of this file and to generate any reaction or information for the user if serious error messages occur (see chapter 7.4 System Message File SYSMESS.<Stream>). Because the management of this files is close depending on the application and its operating mode the management of this files must be central integrated there.

---

The data transmission kernel BCOMMIO is terminated when the command STOPLINE is placed in the corresponding order file.

The following sequence takes place when it is ended:

- When using a Bedas 98xx server, all memory file selections are deleted.
- The operating modes of the terminals are checked and compared with the desired entries in the terminal configuration file. Terminals defined for online operating mode are set to offline. Terminals defined for offline and autonomous are set to the corresponding status if it differs. When the terminals are connected via a server Bedas 98xx, or if the terminals are subterminals Bedas 91xx the operating mode is not changed. In that cases the operating mode can only be changed via the function „Set Operating Mode“.
- The connection to the call interface is cancelled.

There are a view possibilities to stop BCOMMIO:

- by writing the order STOPLINE into the order file (see chapter 7.3 Order file ORDER.<Stream>)
- when using the Call Interface by sending the command :STOPLINE: (see chapter 9.8 Commands from the CI Program to BCOMMIO)
- when using the Call Interface by calling the function ImmediateSTOPLINE (see chapter 9.2 DLL functions)
- when using the Call Interface, BCOMMIO additionally stops by itself, when it send a data record to the CI program and didn't receive an answer within the CI timeout (see chapter 9 Call Interface)



## 7.1. Terminal configuration file TERMDEF.<Stream>

The data of the terminal connected to the system is entered in the TERMDEF.<Stream> terminal configuration file. These entries are made, on one hand, by the BCWIN maintenance program and, on the other hand, by the BCOMMIO data transfer kernel. This file's name is fixed and cannot be changed by the user. One complete record is entered in this file for each terminal.

**A record in the terminal configuration file has the following structure:**

Field	Length (bytes)	Data type	Example	Possible assignment
Line no.	4	Num	234	-3 ... 999
Location	31	Char	Hall 3	Any
GID	1	Num	Ø	ØØ - 29
DID	1	Num	Ø	ØØ - 6Ø
Server	1	Num	1	Ø, 1, 2
Memory file selection	32	Num	1ØØØØØØØ	Ø, 1, -1
Modem	3	Char	AT	--, AT
Call-No or IP address	21	Char	Ø772Ø6Ø11Ø 192.168.1ØØ.1	
Mode	3	Char	OF	ON, OF, AT
Type	16	Char	Bedanet 9380	Terminal type
Prog. no.	23	Char	56Ø-Ø5-KØ1	Program no.
Stream	9	Char	1/LAN1	
Software type	18	Char	B-Client HR5	
Locked	1	Log	1	Ø, 1
Index	1	Num	1	Ø...n
Offset	2	Num	+120	-72Ø ...+78Ø

The „Line no.“ field serves the purpose of alternative addressing (instead of GID/DID) and of managing newly added terminals. The following meanings are defined for this field:

Assignment	Meaning
-3	Terminal is to be analyzed by BCOMMIO.
-2	Terminal has been analyzed by BCOMMIO and entered in the terminal configuration file with current data.
-1	Line no. is not used.
1-999	A terminal can be addressed by this number.

---

A valid **line no.** cannot be assigned until a terminal has been analyzed. If a valid line number is specified, it has a higher priority than GID/DID. A line number can only be assigned once.

The terminal's location can be entered in the **Location** field.

The **GID/DID** fields serve the purpose of terminal addressing.

If the terminal is a server or is connected to a server, this is entered in the **Server** field. The following entries are possible in this field:

Entry	Meaning
Ø	No server
1	TS 986 / BITS 1600 or via TS 986 / BITS 1600
2	Bedas 98xx or via Bedas 98xx

When a Bedas 98xx server is used, the memory file out of which booking data is read and which booking data is to be deleted in the server after sending, must be specified in the **Memory file selection** field. The Memory file selection field consists of 8 long values (= 32 Bytes). The first long value is assigned to memory file 1, the second long value is assigned to memory file 2 etc. The following entries are possible:

Entry	Meaning
-1	Do not select memory file
1	Select memory file Transfer booking data and do not delete the data from the memory file
Ø	Select memory file Transfer booking data and delete the data from the memory file

When a modem is used, this is entered in the **Modem** field. The following entries are possible in this field:

Entry	Meaning
--	No modem
AT	AT modem

---

If the terminal is connected via modem on a COM stream, the phone number of the modem used must be entered in the **Call no.** field. The dial string without ATD can also be entered in the Call no. field, e.g. PØ:Ø772Ø55179.

Alternatively it's possible to enter the IP address of the BETA the terminal is connected to, if the stream is a LAN stream.

The terminal's operating state is entered in the **Mode** field. The following entries are possible in this field:

Entry	Meaning
ON	ONLINE mode
OF	OFFLINE mode
AT	AUTONOMOUS mode

On logon, BCOMMIO enters the terminal type in the **Type** field, e.g. Bedanet (B-Net) 9380.

The **Prog. no.** field serves to accommodate the terminal program number. This field is filled out on analysis by BCOMMIO. This field is blank if a terminal has not yet been analysed or if a program number could not be read during analysis.

The field **Stream** contains the complete name of the stream in the form <stream no>/<interface>, e.g. 1/LAN1.

On logon, a possible software type is entered into the **Software type** field by BCOMMIO, e.g. B-Client HR5.

The **Locked** data field is used internally in the current version of B-COMM WIN32.

The **Index** field defines the index of the configured time zone within the Windows registry.

The **Offset** is the offset of the configured time zone from GMT in minutes, e.g. -120 for time zone (GMT-2).

**In the case of string fields, one character more is additionally reserved to accommodate the final ASCII „NUL“ character, i.e. the last character must always be the ASCII „NUL“ character. Fields or partials of fields that are not needed must also be assigned the ASCII „NUL“ character.**

## 7.2. Configuration file BCOMM32.INI

The file BCOMM32.INI contains all the relevant configuration parameters for the data transmission kernel BCOMMIO and the maintenance program BCWIN. This file is installed into the Windows directory during Setup of B-COMM WIN32, but up from version 4.3.0 of B-COMM WIN32 it may be moved into any directory, which includes also a directory on another PC in the network. If you move the BCOMM32.INI file into another directory, you have to change an entry in the configuration file BCWIN.INI, which resides in the B-COMM directory, so that B-COMM WIN32 has access to the BCOMM32.INI file: entry INIPath=<path to BCOMM32.INI> in section [PATH].

### Content of the configuration file BCOMM32.INI:

Content (Default value)	Meaning
<b>[BCOMM]</b> Version=4.0.0 Language=1  Line=  Act_Lines=  IO_CHECK_TIME=10	Version number of B-COMM WIN32  Language of B-COMM WIN32 (1 = German, 2 = English) (see chapter 3.3.17 Deutsch/English) last active communication stream active communication streams, when the program BCWIN was finished interval (in seconds), in which BCWIN checks whether a communication stream is still active
<b>[OTHERS]</b> logDatei=Y  logGroesse=500  debugDatei=N  debugfile=C:\BCWIN32\ debuglevel=0  smGroesse=10240  protokolldruck=0  SystemDrucker=  bciostart=6	turn on (Y) or off (N) the writing of log file(s) (see chapter 3.3.10 File definition) max. number of entries in the log file(s) (see chapter 3.3.10 File definition) turn on (Y) or off (N) the writing of debug file(s) (see chapter 3.3.10 File definitions) path to the debug file(s) the debug level (see chapter 3.3.7 Start options) max. size of the system message file(s) (in kilobytes) (see chapter 3.3.10 File definition) turn on (1) or off (0) the printer protocol (see chapter 3.3.9 Printer log) name of the system printer (see chapter Printer log) initial window size of BCOMMIO 0 = hidden (invisible)

spiegelIn=N	1 = default (visible) 6 = minimized (in task bar) turn on (Y) or off (N) the writing of the mirror file(s) (see chapter 3.3.10 File definitions)
paramDat=PARAM.DAT	default name of the parameter file (see chapter 3.3.10 File definition)
bookingDat=BOOKING.DAT	default name of the booking file (see chapter 3.3.10 File definition)
cvtstart=6	initial window size of the converter program (see bciostart)
show_tooltip_help=N	turn on (Y) or off (N) the tooltip help (see chapter 3.7.2 Tooltip help)
tooltip_help_time=5	duration of the tooltip help in seconds (see chapter 3.7.2 Tooltip help)
tooltip_help_delay=500	delay of the tooltip help in milliseconds (see chapter 3.7.2 Tooltip help)
beta_single_logon=N	turn on (Y) or off (N) the possibility to logon a single terminal on a BETA (see chapter 3.3.2 Terminal logon)
send_sysmess_to_ci=N	send system messages :SYSMESS_xxx: to the CI program (xxx = error number) (see chapter 3.3.10 File definition)
termmode_time=90	timeout in seconds when changing the terminal mode
termmode_initsyst_time=90	timeout in seconds while switching to system mode
termmode_startapplic_time=120	timeout in seconds while starting a terminal application
max_download_time=1200	timeout in seconds during program download to a terminal
CanDeleteOrders=Y	the user is allowed to delete orders from the order file (Y) or not (N) (see chapter 3.5.1 Display orders and 3.5.2 Delete all orders)
WriteOrderLog=N	turn on (Y) or off (N) the writing of an order log file
send_orderlog_to_ci=N	turn on (Y) or off (N) the sending of start and end time of an order to the CI program
ModemHangupTimeout=2	wait time in seconds after hanging up a modem before a new connection is initiated
TermCheckTime=60	intervall in seconds in which BCOMMIO checks a terminal to be o.k. again
PollCount=20	max. number of poll tries while waiting for a data record from a terminal (only with serial communication)
debug_poll=N	the polling of the terminals is logged in the debug file (Y) or not (N)
watchdog_time=0	interval in seconds in which BCOMMIO writes entries into the watchdog file BCIOWD.<Stream> (0 = writing is

termstat_time=5	disabled) interval in seconds in which BCOMMIO sends erroneous terminals to the external program TERMSTAT
termstat_base_port=4000	base port for the UDP communication with the TERMSTAT program (must be set to the same value in the TERMSTAT.INI file)
empty_term=Y	terminal will be readout before switching the terminal mode (Y) or not (N)
write_sysmess_file=Y	turn on (Y) or off (N) the writing of the system message file(s) (see chapter 3.3.10 File definition)
write_sysmess_to_debug=N	log system messages also in the debug file (Y) or not (N) (see chapter 3.3.10 File definition)
max_debug_days=7	max. number of debug files
disable_service_buttons=N	disable all buttons concerning services under Windows NT/2000/XP (Y) or not (N) (see chapter 3.3.16 Services)
disable_autostart_buttons=N	disable all buttons concerning Autostart (Y) or not (N)
stream_specific_debug=Y	debug files are stream specific (extension <Stream>) (Y) or not (N) (see chapter 3.3.10 File definition)
stop_after_first_error=Y	when an error occurs while executing an order for more than one terminal, the execution is stopped with error (Y) or not (N)
write_booking_to_cinohost=N	all bookings are written into the file CINOHOST.<Stream> when no CI program is configured
write_all_data=N	when this switch is set to Y, then BCOMMIO writes ALL received data records into the booking file (see chapter 7.6 Booking file BOOKING.DAT)
UseLineNumber=Y	when line numbers are configured for the connected terminals, the complete addressing is done using these line numbers (Y) or the line numbers are ignored and the addressing is done with GID/DID (N)
send_enq=N	when BCOMMIO doesn't receive an answer to a data record within the CI timeout, it sends the :ENQ: record to the CI program (Y) or shuts down immediately (N) (see chapter 9 Call Interface)
FTPDebugLevel=0	Debug level für BCOMMFTP (0 to 9)
ShowForm=FALSE	TRUE shows the window of BCOMMFTP
extended_orders=N	N=uses normal structure of the order file Y=extended structure of order file is used (see chapter 7.3 Order file

<p>mr_block_cnt=1Ø</p> <p>par_block_cnt=1Ø</p> <p>add_cr=N</p> <p>soh_for_dle=N</p> <p><b>[Default]</b>  TargetPath=/Bedanet/transfer  LoginName=localuser  Password=Bedanet</p> <p><b>[PATH]</b>  bcommPfad=C:\BCWIN32\  paramPfad=C:\BCWIN32\PARA\  nutzPfad=C:\BCWIN32\DATA\  stammPfad=C:\BCWIN32\PERS\  spiegelPfad=C:\BCWIN32\DATA\  progPfad=C:\BCWIN32\TERMPGM\  bedaproPfad=C:\BCWIN32\PARA\  avisoPfad=C:\BCWIN32\AVISO\  orderPfad=C:\BCWIN32\  logPfad=C:\BCWIN32\  </p> <p><b>[LIZENZ]</b>  VORNAME=Example  NAME=Example  STRASSE=Example street 7  PLZ/ORT=Example town 123  TELEFON=  LIZENZNR=12345612341234567  ACTCODE=123456  ABLAUFDATUM=</p> <p><b>[Line&lt;x&gt;]</b>  Autostart=N</p> <p>actto=6  respto=2  cito=6Ø</p> <p>callInterface=N</p> <p>callProg=</p>	<p>ORDER.&lt;Stream&gt;)</p> <p>amount of master records/block during master record download</p> <p>amount of parameter records/block during parameter download</p> <p>setting this switch to Y causes BCOMMIO to add a &lt;CR&gt; to each data record in serial communication</p> <p>setting this switch to Y causes BCOMMIO to replace the &lt;DLE&gt; character in the BPA protocol by &lt;SOH&gt;</p> <p>Default parameter for BCOMMFTP to log onto a Bedanet 938Ø (B-Net 938Ø) terminal</p> <p>(see chapter 3.3.8 Directories)</p> <p>Installation path of B-COMM WIN32</p> <p>path to parameter files</p> <p>path to booking files</p> <p>path to master record files</p> <p>path to mirror files</p> <p>path to terminal programs</p> <p>path to BEDAPRO files</p> <p>path to AVISO files</p> <p>path to (time-controlled) order files</p> <p>path zu log, system message and debug files</p> <p>(see chapter 3.6.5 Licence registration)</p> <p>launch stream automatically when BCWIN is started (see chapter 3.3.6 Initialize communication stream)</p> <p>activity timeout in seconds</p> <p>response timeout in seconds</p> <p>Call Interface timeout in seconds (see chapter 9 Call Interface)</p> <p>BCCOMMIO uses a Call Interface (Y) or not (N) (see chapter 9.11 Configuration of the Call Interface)</p> <p>name of the CI program (see chapter 9.11 Configuration of the Call Interface)</p>
---	--

cvtProg=C:\BCWIN32\DDE2SMEM.EXE	name of the converter program, when the Call Interface uses DDE (see chapter 9.11 Configuration of the Call Interface)
CI_via_SM=Y	the Call Interface uses Shared Memory (Y) or DDE (N)
schnittstelle=COMx	name of the interface (COM1 to COM8 or LANØ to LAN29) (see chapter 3.3.6 Initialize communication stream)
serialPara=baud data parity Stopp	settings of the terminal interface
scheduler_active=N	the scheduler is active (Y) or inactive (N) (see chapter 3.3.15 Scheduler)
scheduler_daily=Y	time-controlled orders are executed once a day (Y) or periodically (N)
starttime_daily=ØØ:ØØ:ØØ	start time if scheduler_daily=Y
starttime_interval=ØØ:ØØ:ØØ	start time if scheduler_daily=N
start_interval=12Ø	interval in seconds when scheduler_daily=N
scheduler_days=62	weekdays on which time-controlled orders should be executed
scheduler_append=Y	time-controlled orders are appended to the order file (Y) or overwrite the order file (N)
separate_timedats=N	the order „Set date/time“ is sent as one record with General Select (N) or a record with GID/DID is sent to each terminal (Y) (see chapter 3.2.6 Set date/time)
BPADelay=Ø	delay in milliseconds between polls to the terminals (only with serial communication)
restart_ci_prog=N	when this switch is set to Y, BCOMMIO restarts the CI program, when it detected that the CI program has shut down. This switch has only an effect, when a CI program is configured with the switch callProg.
start_without_ci_prog=N	BCOMMIO doesn't launch a CI program when this switch is set to Y
switch_mode=Y	BCOMMIO switches ONLINE terminals automatically to ONLINE during startup
switch_mode_down=Y	BCOMMIO automatically switches ONLINE terminals to the mode given by the switch down_mode during shutdown
down_mode=OF	mode to which terminals should be switched during shutdown of BCOMMIO, when the switch switch_mode_down is set to Y (OF = OFFLINE, AT = AUTONOMOUS)
WriteBookings=Y	BCOMMIO writes received bookings into a booking file (Y) or not (N) (siehe 3.3.10 File definition)
WriteOrderError=N	when this switch is set to Y, BCOMMIO writes an error number into the order file



---

auto_para_modem=N	if an order could not be executed coorrectly.
auto_para_file=elsa.atm	when this switch is set to Y, BCOMMIO automatically sends the file given by the switch auto_para_file to the modem, before it starts a connection the first time this file is sent to a modem by BCOMMIO when the switch auto_para_modem is set to Y
send_ATZ=Y	after sending parameters to a modem BCOMMIO sends automatically the command ATZ (Y) or not (N)
mr_id_len=7	lebgth of the ID number in master records
DebugDay=2	BCOMMIO internal switch, DON'T CHANGE!!!
DebugFile1=..... : :	BCOMMIO internal switches, DON'T CHANGE!!!
DebugFile<n>=..... smFP=...	BCOMMIO internal switch, DON'T CHANGE!!!

---

### 7.3. Order file ORDER.<Stream>

Each interface is assigned an order file. All orders selected by the BCWIN maintenance program or the user program are placed in the order file ORDER.<Stream>. BCOMMIO executes all orders in the order file that do not have an execution status. Orders are placed in the order file in the form of commands. By use of the switch **extended\_orders** in the section **[Others]** of the configuration file BCOMM32.INI (see chapter 7.2 Configuration file BCOMM32.INI) you can set the format of the order file.

**Attention: after changing the value of the switch `extended_orders` all order files must be deleted to prevent the order files from including orders of different formats.**

When the switch is set to **N** (Default) the following format for the order file is used:

Field	Length (bytes)	Data type	Example
Command	9	Char	MESSAGE
Subcommand	5	Char	D2
Line no.	2	Num	-1
GID	1	Num	Ø
DID	1	Num	Ø
File	13	Char	
Data	25	Char	Hello
External memory field	2	Log	Ø
Execution status	6	Char	OKCO
Reserved	2	Log	1

- **Command:** here a command has to be entered
- **Subcommand:** the field serves to accommodate the record type and the record type modification
- **Line no:** contains the line number of the terminal
- **GID:** contains the group address of the terminal
- **DID:** contains the device address of the terminal
- **File:** contains a file name depending on the command
- **Data:** the content of this field depends on the command
- **External memory field:** the content of this field depends on the command
- **Execution status:** contains the execution status of the order, which is inserted by BCOMMIO after execution of the order

When the switch `extended_orders` is set **Y**, the order file has the following (extended) structure:

Field	Length (bytes)	Data type	Example
Command	9	Char	MESSAGE
Subcommand	5	Char	D2
Line no.	2	Num	-1
GID	1	Num	Ø
DID	1	Num	Ø
File	13	Char	
Data	25	Char	Hello
External memory field	2	Num	Ø
Execution status	6	Char	OKCO
Reserved	2	Num	1
No. Of repeats	1	Num	5
Repeat interval (in minutes)	1	Num	15
Priority (Ø to 9)	1	Num	Ø
Creation date (format MMDDYYYY)	9	Char	Ø2222ØØ1
Creation time (format HHMMSS)	7	Char	Ø823Ø5
Start date (format MMDDYYYY)	9	Char	Ø2222ØØ1
Start time (format HHMMSS)	7	Char	Ø823Ø9
End date (format MMDDYYYY)	9	Char	Ø2222ØØ1
End time (format HHMMSS)	7	Char	Ø82317
Error number (bit coded)	2	Num	16
Terminals with error (GID and DID)	1Ø * 2	Num	
Repeat date (format MMDDYYYY)	9	Char	Ø2222ØØ1
Repeat time (format HHMMSS)	7	Char	Ø835ØØ

---

The fields **Command** to **Reserved** are similar to the fields in the normal order file.

- **No. of repeats:** here you can specify how many times a failed order should be repeated automatically by BCOMMIO. During startup of BCOMMIO this field is set to Ø for each order with order state ERRU (no repeats). **This field is not used by the FTP kernel BCOMMFTP.**
- **Repeat interval:** here you can specify in which interval failed errors will be repeated if the field **No. of repeats** contains a value > Ø. When this field contains Ø and the field **No. of repeats** a value > Ø, the failed order will be repeated immediately, in case there's no other order with a higher **priority** at that time. **This field is not used by the FTP kernel BCOMMFTP.**
- **Priority:** here each order could be assigned a priority between Ø (lowest priority) and 9 (highest priority). When there are more than one orders with no execution status in the order file, they are executed in the sequence of their priorities. **This field is not used by the FTP kernel BCOMMFTP.**
- **Creation date:** the application which creates the order, can insert the creation date into this field
- **Creation time:** the application which creates the order, can insert the creation time into this field
- **Start date:** BCOMMIO inserts the start date when it starts executing the order
- **Start time:** BCOMMIO inserts the start date when it starts executing the order
- **End date:** BCOMMIO inserts the end date when it has finished executing the order
- **End time:** BCOMMIO inserts the end time when it has finished executing the order
- **Error number:** in case of a failed order (ERRU) BCOMMIO inserts a bit coded error code into this field:
  - bit Ø: Error while sending to a terminal
  - bit 1: Error while receiving from a terminal
  - bit 2: Error during modem communication
  - bit 3: Error while opening a file
  - bit 4: Error while writing into a file
  - bit 5: Error while reading from a file
  - bit 6: The addressed terminal is not logged-on
  - bit 7: The addressed terminal is currently not available
  - bits 8 to 15: not used
- **Terminals with error:** BCOMMIO inserts the addresses in format GID/DID of those terminals, which caused an error during execution of the order. This field can contain a max. of 1Ø terminal addresse. Free entries contain the values 255/255.
- **Repeat date:** BCOMMIO inserts the date of the next repetition if the fields **No. of repeats** and **Repeat interval** contain values > Ø. **This field is not used by the FTP kernel BCOMMFTP.**
- **Repeat time:** BCOMMIO inserts the time of the next repetition if the fields **No. of repeats** and **Repeat interval** contain values > Ø. **This field is not used by the FTP kernel BCOMMFTP.**

The following orders are available:

Commando	Subcommand	Meaning
ANALYSE		Check the actual terminal configuration and update the terminal configuration file
LOWPAR	STOP, GPAR, SPAR, STRT	Internal command: should only be written into the order file by the maintenance program BCWIN
MESSAGE	DØ, D1, D2	Transmit display record
MODEMPAR	AT	Parameter setting of a modem
MRUPLD	ALL, PSEU, FILE	Master record upload to file
PGMDNLD		Internal command: should only be written into the order file by the maintenance program BCWIN
READCFG		Read configuration file BCOMM32.INI
READTDF		Read terminal configuration file
REDAAUTO		Request data from terminals in operating mode AUTONOMOUS
REDAOFF	No entry, tnm	Request data from terminals in operating mode OFFLINE
REPARA		Parameter upload to file
SCHDUL		Internal command: should only be written into the order file by the maintenance program BCWIN
SETMODE	TØ, T1, T2	Set terminal to operating mode ONLINE, OFFLINE or AUTONOMOUS
STATUS	No entry, DLVL, SIZE, TRLT	Get state of I/O kernel BCOMMIO or of a terminal
STOPLINE		Stop communication
SWMEM	MØ, M1, M2, tnm	Control the buffer memory of terminals, at which BUFFER BACKUP is set
TERMMODE	SSRV, SAPL, SSYS, SETB, RESB	Internal command: should only be written into the order file by the maintenance program BCWIN
TIMEDAT	UØ, U1, SYSØ, SYS1	Send date and time to all terminals. The transmission is done to all terminals simultaneously by use of General Select
UPDPRGNO		Updates the program numbers of the terminals in the terminal definition file in case a program update has been made
WRDATRA		Send transparent data from file
WRPARA		Parameter download from file
WRPEAUTO		Master record download according to the father-son principle (see Kaba Benzing documentation: "Principles of data communication")
WRPEHAU		Master record download. The Y records to be transferred are taken from the specified file and are sent to the specified terminals

---

The **Line no.** or **GID** and **DID** fields serve the purpose of addressing. If GID/DID and Line no. are used for addressing, the Line no. has the higher priority.

Data received from the terminal during execution of this command is placed in the file that is specified in the **File** field.

Data that is to be sent is in the **Data** field.

A „1“ must be entered in the **External memory field** if it is intended to send an external data file. The name of the file must be entered in the **Data** field.

BCOMMIO assigns an execution status to the **Execution status** field after execution of the order.

The following execution status can occur:

Execution status	Meaning
No execution status	Command not yet executed
OKCO	Command executed successfully
ERCO	The command name is invalid or contains invalid parameters
ERRU	An error occurred during execution of the command (e.g. Terminal not responding)
OKME	A valid command, but a system message was received from the terminal
HOLD	If the „HOLD“ identifier is entered in the <b>Execution status</b> field prior to execution of the command, this command is not executed
RUN	Command is currently being processed

The **Locked** data field is reserved for expansion.

**In the case of string fields, one character more is additionally reserved to accommodate the final ASCII „NUL“ character, i.e. the last character must always be the ASCII „NUL“ character. If the „Line No.“ field is not used, it must be assigned „-1“ (FF<sub>hex</sub>). Fields or partials of fields that are not needed must also be assigned the ASCII „NUL“ character.**

The individual orders have the following record structure:

### 7.3.1. Order ANALYSE

Field	Assignments
Command	ANALYSE
Subcommand	No entry
Line No.	-3 (terminal is to be analysed)
GID	Group address
DID	Device address or $\Delta$ (7F <sub>hex</sub> ) for all terminals of the respective group
File	Booking file
Data	No entry
External memo field	Ø
Execution status	Is assigned as execution status by BCOMMIO after execution of the command
Locked	Reserved

**Note:** more exact data describing the terminal (e.g. modem, phone No., server etc.) must be in the Terminal Configuration File TERMDEF.<Stream> (see chapter 7.1 Terminal configuration file TERMDEF.<Stream>).

### 7.3.2. Order MESSAGE

Field	Assignments
Command	MESSAGE
Subcommand	DØ : Static display D1 : Flashing display D2 : Flashing display and hooting
Line No.	Line No. or Ø for all terminals
GID	Group address or $\Delta$ (7F <sub>hex</sub> ) for all groups
DID	Device address or $\Delta$ (7F <sub>hex</sub> ) for all terminals of the respective group
File	No entry
Data	„Hello“
External memo field	Ø: The data in the <b>Data</b> field is sent to the terminal, taking the <b>Subcommand</b> into account 1 : The entry in the <b>Data</b> field consists of a file containing display records along with the record type and the record type modification. During transmission, the terminal address is entered in the first 3 positions of the records in this file. The entry in the <b>Subcommand</b> field is ignored.
Execution status	Is assigned an execution status by BCOMMIO after execution of the command
Locked	Reserved

---

### 7.3.3. Order MODEMPAR

Field	Assignments
Command	MODEMPAR
Subcommand	AT
Line No.	No entry
GID	No entry
DID	No entry
File	Parameter file to send
Data	No entry
External memo field	Ø
Execution status	Is assigned an execution status by BCOMMIO after execution of the command
Locked	Reserved

### 7.3.4. Order MRUPLD

Field	Assignments
Command	MRUPLD
Subcommand	ALL = upload all master records PSEU = upload pseudo master record FILE = get master record with ID number, the ID number is in file IDNO.REC
Line No.	Line no.
GID	Group address
DID	Device address
File	Booking file
Data	Master record file in which the uploaded master records are to be placed
External memo field	Ø
Execution status	Is assigned as execution status by BCOMMIO after execution of the command
Locked	Reserved



---

### 7.3.5. Order READCFG

Field	Assignments
Command	READCFG
Subcommand	No entry
Line No.	No entry
GID	No entry
DID	No entry
File	No entry
Data	No entry
External memo field	Ø
Execution status	Is assigned an execution status by BCOMMIO after execution of the command
Locked	Reserved

### 7.3.6. Order READTDF

Field	Assignments
Command	READTDF
Subcommand	No entry
Line No.	No entry
GID	No entry
DID	No entry
File	No entry
Data	No entry
External memo field	Ø
Execution status	Is assigned an execution status by BCOMMIO after execution of the command
Locked	Reserved

### 7.3.7. Order REDAAUTO

Field	Assignments
Command	REDAAUTO
Subcommand	No entry
Line No.	Line No. or Ø for all terminals
GID	Group address or $\Delta$ ( $7F_{hex}$ ) for all groups
DID	Device address or $\Delta$ ( $7F_{hex}$ ) for all terminals of the respective group
File	Booking file
Data	No entry
External memo field	Ø
Execution status	Is assigned as execution status by BCOMMIO after execution of the command
Locked	Reserved

### 7.3.8. Order REDAOFF

Field	Assignments
Command	REDAOFF
Subcommand	No entry : Request data from the terminals that are in the OFFLINE mode tnm : Request data from the Bedas 98xx server. The value n specifies the file to read (1...8) The value m specifies whether the file is to be deleted after reading (Ø=Delete file; 1=Do not delete file)
Line No.	Line No. or Ø for all terminals
GID	Group address or $\Delta$ ( $7F_{hex}$ ) for all groups
DID	Device address or $\Delta$ ( $7F_{hex}$ ) for all terminals of the respective group
File	Booking file
Data	No entry
External memo field	Ø
Execution status	Is assigned as execution status by BCOMMIO after execution of the command
Locked	Reserved

---

### 7.3.9. Order REPARA

Field	Assignments
Command	REPARA
Subcommand	Parameter group to be fetched (e.g. ... for all parameters, F+ for all function key parameters)
Line No.	Line No.
GID	Group address
DID	Device address
File	Booking file
Data	Parameter file in which the uploaded parameters are to be placed
External memo field	Ø
Execution status	Is assigned as execution status by BCOMMIO after execution of the command
Locked	Reserved

### 7.3.10. Order SETMODE

Field	Assignments
Command	SETMODE
Subcommand	TØ : Switch Terminal Online T1 : Switch terminal Offline T2 : Switch terminal to Autonomous
Line No.	Line No. or Ø for all terminals
GID	Group address or Δ (7F <sub>hex</sub> ) for all groups
DID	Device address or Δ (7F <sub>hex</sub> ) for all terminals of the respective group
File	Booking file
Data	No entry
External memo field	Ø
Execution status	Is assigned as execution status by BCOMMIO after execution of the command
Locked	Reserved

### 7.3.11. Order STATUS

Field	Assignments
Command	STATUS
Subcommand	No entry, DLVL (for internal purpose), SIZE (for internal purpose), TRLT (for internal purpose)
Line No.	No entry
GID	Ø...29: Request status of the terminal with the entered GID
DID	Ø...59: Request status of the terminal with the entered DID
File	No entry
Data	No entry
External memo field	Ø
Execution status	Is assigned an execution status by BCOMMIO after execution of the command OKCO: terminal is logged on and available ERCO: terminal not logged on ERRU: terminal logged on but not available
Locked	Reserved

### 7.3.12. Order STOPLINE

Field	Assignments
Command	STOPLINE
Subcommand	No entry
Line No.	No entry
GID	No entry
DID	No entry
File	No entry
Data	No entry
External memo field	Ø
Execution status	Is assigned an execution status by BCOMMIO after execution of the command
Locked	Reserved

---

### 7.3.13. Order SWMEM

Field	Assignments
Command	SWMEM
Subcommand	MØ : Read buffer memory M1 : Delete buffer memory M2 : Initialise buffer memory tnm : (Bedas 98xx) The first value n specifies the file (1...8). The second value m contains the subcommand (2=Fetch data and delete from file; 3=Fetch data and do not delete from file; 4=Do not fetch data or delete from file)
Line No.	Line No. or Ø for all terminals
GID	Group address or $\Delta$ (7F <sub>hex</sub> ) for all groups
DID	Device address or $\Delta$ (7F <sub>hex</sub> ) for all terminals of the respective group
File	Booking file
Data	No entry
External memo field	Ø
Execution status	Is assigned an execution status by BCOMMIO after execution of the command
Locked	Reserved

### 7.3.14. Order TIMEDAT

Field	Assignments
Command	TIMEDAT
Subcommand	UØ : Date/time without seconds U1 : Date/time with seconds SYSØ : Use PC system date/time without seconds SYS1 : Use PC system date/time with seconds
Line No.	Ø (all terminals)
GID	Δ (7F <sub>hex</sub> )
DID	Δ (7F <sub>hex</sub> )
File	Booking file
Data	<p><b>Subcommand = UØ:</b> Date to send (YYMMDD), time to send without seconds (HHMM), date on storage (YYMMDD), time on storage (HHMMSS)</p> <p>Example: 9705151635970515163406</p> <p>The date 15.5.97 and the time 16:35 are to be sent to the terminals. These two items of data were stored in the order file at 16:34: 06 h on 15.5.97.</p> <p><b>Subcommand = U1:</b> Date to send (YYMMDD), time to send with seconds (HHMMSS), date on storage (YYMMDD), time on storage (HHMMSS)</p> <p><b>Subcommand = SYSØ:</b> Offset in 15 minutes steps, e.g. +4 or "TIMEZONE" if the time zone configured in the terminal should be used</p> <p><b>Subcommand = SYS1:</b> see SYSØ</p>
External memo field	Ø
Execution status	Is assigned an execution status by BCOMMIO after execution of the command
Locked	Reserved

---

### 7.3.15. Order UPDPRGNO

Field	Assignments
Command	UPDPRGNO
Subcommand	No entry
Line No.	Line No. or Ø for all terminals
GID	Group address or $\Delta$ (7F <sub>hex</sub> ) for all groups
DID	Device address or $\Delta$ (7F <sub>hex</sub> ) for all terminals of the respective group
File	No entry
Data	No entry
External memo field	Ø
Execution status	Is assigned as execution status by BCOMMIO after execution of the command
Locked	Reserved

### 7.3.16. Order WRDATRA

Field	Assignments
Command	WRDATRA
Subcommand	No entry
Line No.	Line No. or Ø for all terminals
GID	Group address or $\Delta$ (7F <sub>hex</sub> ) for all groups
DID	Device address or $\Delta$ (7F <sub>hex</sub> ) for all terminals of the respective group
File	Booking file
Data	Date whose contents are to be sent. During transmission, the terminal addresses entered in the first three positions. Any addressing information in the file is ignored.
External memo field	Ø
Execution status	Is assigned as execution status by BCOMMIO after execution of the command
Locked	Reserved

### 7.3.17. Order WRPARA

Field	Assignments
Command	WRPARA
Subcommand	No entry
Line No.	Line No. or Ø for all terminals
GID	Group address or $\Delta$ (7F <sub>hex</sub> ) for all groups
DID	Device address or $\Delta$ (7F <sub>hex</sub> ) for all terminals of the respective group
File	No entry
Data	Parameter file to be sent The first line must contain the program No. of the terminal to be set. During transmission, the address is entered in the first three positions of the parameter records. Any addressing information in the file is ignored.
External memo field	Ø = check program number 1 = ignore program number
Execution status	Is assigned as execution status by BCOMMIO after execution of the command
Locked	Reserved

### 7.3.18. Order WRPEAUTO

Field	Assignments
Command	WRPEAUTO
Subcommand	No entry
Line No.	Line No. or Ø for all terminals
GID	Group address or $\Delta$ (7F <sub>hex</sub> ) for all groups
DID	Device address or $\Delta$ (7F <sub>hex</sub> ) for all terminals of the respective group
File	Booking file
Data	Master record file (father file) Each master record must contain addressing information (wildcards permitted)
External memo field	Ø
Execution status	Is assigned as execution status by BCOMMIO after execution of the command
Locked	Reserved



---

### 7.3.19. Order WRPEHAAU

Field	Assignments
Command	WRPEHAAU
Subcommand	No entry
Line No.	Line No. or Ø for all terminals
GID	Group address or $\Delta$ (7F <sub>hex</sub> ) for all groups
DID	Device address or $\Delta$ (7F <sub>hex</sub> ) for all terminals of the respective group
File	Booking file
Data	Master record file to be sent Each master record must contain addressing information (wildcards permitted)
External memo field	Ø
Execution status	Is assigned as execution status by BCOMMIO after execution of the command
Locked	Reserved

---

## 7.4. System Message File SYSMESS.<Stream>

Each interface is assigned a system message file. System messages (S, X and O records) of the terminals are entered in the system message file SYSMESS.<Stream>. Further error messages originating from BCOMMIO are also placed in this file, e.g. interface not ready etc.

The messages contain the date and the time when the error occurred. Each error message is assigned an error code, the record received from the terminal and a text.

To be able to establish a cross reference to the order file, the command in which the error occurred is also entered in this file.

**The system message file SYSMESS.<Stream> has the following structure:**

Field	Length (bytes)	Data type	Example
Command	9	Char	REDAOFF
Subcommand	5	Char	
Line no.	4	Num	2
GID	1	Num	
DID	1	Num	
Date	7	Char	15Ø497
Time	7	Char	163Ø45
Error code	4	Num	1Ø5
RT / RTM	76	Char	OB
Plain language text	76	Char	Terminal not responding

The **Command** and **Subcommand** fields serve to accommodate the command during whose execution a system message occurred. The record type and record type modification are entered in the **RT / RTM** field. The **Line no.**, **GID** and **DID** fields serve the purpose of addressing. The **Date** and **Time** of a system message and an **Error code** are entered in the corresponding fields.

The error codes have the following meanings:

Error code	RT / RTM	Plain language text	Possible cause/remedy
-1		Error when saving Record	Check write/read privileges in the bcomm directory.
1Ø1	O1	Data Record Contents wrong	The addressed terminal has received a record that was not understood. Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications.
1Ø2	O2	Data Record cannot be executed at the moment	The addressed terminal has received a record that can currently not be executed. Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications.
1Ø3	O3	Data Record cannot be executed	The addressed terminal has received a record that can not be executed. Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications.
1Ø4	OA	Terminal not logged on	An attempt has been made to address a none-existent terminal via a terminal server. Determine the meaning of the record in the corresponding server documentation. In the log file, analyse the corresponding data communications.
1Ø5	OB	Terminal does not respond	The terminal connected to a terminal server is configured, but is not responding to a record. Determine the meaning of the record in the corresponding server documentation. In the log file, analyse the corresponding data communications.
1Ø6	OC	Terminal responds with RVI	The terminal connected to a terminal server is configured, but is responding to a record with RVI. Determine the meaning of the record in the corresponding server documentation. In the log file, analyse the corresponding data communications.
1Ø7	Oa	File cannot be opened/deleted	An attempt has been made to select a file in a BMD975 terminal that can currently not be accessed. Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications.
1Ø8	Ob	File does not exist	An attempt has been made to select a file in a BD975 terminal that is not configured. Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications.
1Ø9	Oc	File already exists	An attempt has been made to select a file in a BMD975 terminal that already exists
11Ø	Od	False Record Type Modification. VØ/v9 expected	Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications.

Error code	RT / RTM	Plain language text	Possible cause/remedy
111	SØ	Buffer memory empty	Informal: Terminal's data memory is not empty. Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications.
112	S2	Buffer memory full	The specified terminal has no further space for bookings. The terminal's buffer memory must be deleted as soon as possible. Check whether the terminal is in autonomous mode and, if yes, fetch data in autonomous mode via bcomm. When working with buffer backup, the buffer memory must be suitably controlled via bcomm. Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications.
113	S3	Start of parameter setting	Informal: The parameter setting mode has been activated on the terminal. Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications.
114	S4	End of parameter setting	Informal: The parameter setting mode has been terminated on the terminal. Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications.
115	S5	Start of diagnosis	Informal: The diagnostic mode has been activated on the terminal. Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications.
116	S6	End of diagnosis	Informal: The diagnostic mode has been terminated on the terminal. Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications.
117	SA	Terminal OK again	The terminal specified in the record reports OK again. If such messages occur frequently together with SB type messages, have the terminal configuration checked by after-sales service. The interface timeouts in bcommio may be wrongly set. Check the interface configuration. Check the interface parameters of the terminal(s).
118	SB	Terminal does not respond	The terminal specified in the record is not responding. If such messages occur frequently together with RT type messages, have the terminal configuration checked by after-sales service. The interface timeouts in bcommio may be wrongly set. Check the interface configuration. Check the interface parameters of the terminal(s).
119	X1	Door open too long	Informal: Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications.

Error code	RT / RTM	Plain language text	Possible cause/remedy
12Ø	X2	Door broken open	Informal: Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications
121	X3	Door closed again	Informal: Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications
122	X4	Start of sabotage alarm	Informal: Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications
123	X5	End of sabotage alarm	Informal: Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications
124	O4	Unknown data record	A record of an unknown type has been send to a terminal. Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications. If the record has been send by the SWP application, have a damp of the log file send to SWP. If the record has been send by bcommio, send the damp of the Log file to Kaba Benzing.
125	X6	Duress alarm	Informal: Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications
126	Sa	Subterminal ok again	Informal: Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications
127	Sb	Subterminal does not respond	Informal: Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications
128	SH	Report serial no. /hardware parameters	Informal: Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications
129	SV	Report program no.	Informal: Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications
13Ø	TM	Report status change	Informal: Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications
131	X8	Report bolt position error	Informal: Determine the meaning of the record in the corresponding terminal documentation. In the log file, analyse the corresponding data communications
2ØØ		Block check character (BCC) wrong	The specified terminal send a record with a wrong checksum (Block-check). If these errors occur frequently, inform after-sales service and have the data transfer medium and the terminals checked.

Error code	RT / RTM	Plain language text	Possible cause/remedy
2Ø1		Data string too long	The specified terminal has sent a record which is longer than 255 bytes. If these error occurs inform after-sales service and have the terminals checked.  Note: Kaba Benzing terminals are sending no data records longer than 255 bytes.
2Ø2		COM error during transmission to a terminal	The selected interface is not ready. Check if the interface is in use with other programs. Check if the terminals are connected correct.
2Ø5		No >9 record received after parameter upload	After a parameter upload no >9 record was received. In the log file, analyse the corresponding data communications.
2Ø6		No Y9 record received after master record upload	After a master record upload no Y9 record was received. In the log file, analyse the corresponding data communications.
2Ø7		Operating mode cannot be changed	The required terminal mode (ONLINE, OFFLINE, AUTONOM) could not be set for the specified terminal. In the log file, analyse the corresponding data communications.
2Ø8		Parameter does not fit for this Terminal	The records contained in the parameter file to be sent are intended for a different type of terminals. Check the terminal type from which the parameters were loaded (the first record in the file contains the program number of the terminal from where the parameters were loaded) and compare it against the program number of the terminal to be loaded. If the file does not contain any terminal type information, do not change the file and send it again -> risk of wrong parameter settings of the terminal. Clarify the origin of the file and, if necessary, discuss the further procedure with the SWP. If the file does not contain a program number as the first record, this file can not be downloaded and downloading is also not advisable. If the file is nevertheless to be sent, a record with a suitable program number or ... can be entered using an editor.
2Ø9		No Terminal is in Operating Mode OFFLINE	This message occurs when the "Request Bookings" command has been issued and bcommio was not able to find a terminal that is configured in the OFFLINE mode.  Remedy: Send the "Request bookings" command again and specify the correct mode.  Note: The terminal mode can be displayed in bcomm display terminal configuration.

Error code	RT / RTM	Plain language text	Possible cause/remedy
21Ø		No Terminal is in Operating Mode AUTONOMOUS	This message occurs when the "Request Bookings" command has been issued and bcommio was not able to find a terminal that is configured in the AUTONOM mode.  Remedy: Send the "Request bookings" command again and specify the correct mode.  Note: The terminal mode can be displayed in bcomm display terminal configuration.
211		Dialog answer received	BCOMMIO received a dialog answer to the transmission of a data record (e.g. O2 record). In the log file, analyse the corresponding data communications.
212		Alarm record received	Informal: BCOMMIO received an alarm record from a terminal.
213		Status record received	Informal: BCOMMIO received a status record from a terminal.
3ØØ		Printer not ready	This message occurs when BCOMMIO could not create the print file. Check for enough disk space on hard disk.
3Ø1		Stream not ready/ or terminal does not respond	BCOMMIO could not call the corresponding communication stream or a terminal has been called which is not existing in the communication stream under the GID/DID used in the command.  Remedy: for serial interfaces: check interface pinning (RxD, TxD...) and compare with the Kaba Benzing serial connection. A false communication stream has been entered (COM1 to COM4 or LAN1 to LAN3Ø) or another program is accessing the interface. Compare interface parameters (Baud rate...) with interface parameters of the terminals. Parameter setting must be equal. Check errors and do corrections.  Remedy for Ethernet interfaces: check IP addresses and the UDP port no. in the file BETASYS.<>. Remedy if the error is no interface problem: check terminal addresses and repeat operation with correct terminal address. Compare interface parameters with the interface parameters of the used interface.
3Ø2		Terminal does not respond	BCOMMIO cannot call the corresponding terminal.  Remedy: repeat operation with a correct group and device identification. If the terminal logs itself frequently on and off, check interface parameters of the terminals and the party-line.
3Ø3		Error accessing the order file	BCOMMIO cannot open or create the file ORDER.<Stream>.  Remedy: check or modify the B-COMM installation directories.

Error code	RT / RTM	Plain language text	Possible cause/remedy
304		Error during system initializing	An error occurs during initializing of serial interface or BETA adapters. Check if the interface is in use with other programs. Check if the BETA is configured correct. Check if the BETA is connected correct.
305		Error reading the master record file	BCOMM cannot open or create the file entered in the order. Remedy: check whether the file entered in the order is existing in the correct path.
307		Error reading the configuration file	BCOMMIO cannot open the configuration file BCOMM32.INI. Remedy: check whether the BCOMM32.INI is existing in the Windows directory.
308		Error opening the terminal configuration file	BCOMMIO cannot open the terminal definition file. Remedy: modify the write/read authorizations of the files and the settings of the used directories.
309		Error opening the master record file	See error 305
310		Error opening the parameter file	BCOMMIO cannot open the parameter file entered in the order. Check whether the file entered in the order has been created in the correct path.
311		Error writing in booking file	BCOMMIO cannot open the booking file which is entered in the order or the configuration file. Remedy: check whether the file entered in the order has been created in the correct path. Because the received data record could not be saved in the planned file, BCOMMIO tries to save the data record in the file BOOKSAV.DAT. If this is also impossible BCOMMIO cancels itself to prevent further data loss. The error source must absolutely be checked !
312		System time and date not up to date	BCOMMIO requests the currently time/date from the system for setting the terminals time and date. The structure of the data record which is sent by the system is invalid. Remedy: check system time and date.
313		Error opening the son file	BCOMMIO cannot open the file SON<GID><DID>.<.>. Check path.
315		Error creating a temporary file	BCOMMIO cannot create a temporary file. Check for enough free disk space on hard disk drive.
316		Beta does not respond	During normal operation an error occurred with a Beta. Check the network.
317		Beta is o.k. again	Informal: a disturbed Beta is o.k. again
318		Error opening the log file	BCOMMIO cannot process the file Log.<Stream>. Maybe the file contains a invalid record structure. Check via B-COMM the number of entries in the file. 100 records are recommended if logging is used.



Error code	RT / RTM	Plain language text	Possible cause/remedy
319		False baud rate has been entered	In the file BCOMM32.INI a invalid baud rate is configured for the communication stream. Remedy: set correct baud rate.
320		False data bit entry	In the file BCOMM32.INI a invalid number of data bits is configured for the communication stream. Remedy: set correct number of data bits.
321		False parity has been entered	In the file BCOMM32.INI a invalid parity is configured for the communication stream. Remedy: set the correct parity.
322		Number of stop bits invalid	In the file BCOMM32.INI a invalid number of stop bits is configured for the communication stream. Remedy: set correct number of stop bits.
323		Cannot write data into booking file(s)! End of communication	BCOMMIO could not write the received data into the configured booking file. This is a data loss, so the communication has been cancelled. Check for enough free disk space on the hard disk drive. If free disk space is existing start communication again.
327		Error during parameter setting of the interface	BCOMMIO could not set the necessary transmission parameters for the communication stream. Remedy: check whether a existing interface (COM1 to COM8) has been entered.
328		Timeouts for the interface could not be set	Check whether the Activity timeout is set to 3 times the Response timeout (in the configuration dialog). If so, it is an internal error of BCOMMIO. In this case, contact Kaba Benzing.
329		The entered communication stream is not configured	BCOMMIO has been started with a communication stream for which no entry in the configuration file COMM32.INI is existing. Remedy: when this communication stream shall absolutely be used, configure a communication stream with B-COMM. When BCOMMIO was started with a incorrect communication stream, start it again with a correct communication stream.
330		Error opening the print file	This message occurs when BCOMMIO could not create the configured file for printing. Remedy: check free disk space. On hard disk drive.
331		Error opening the temporary print file	This message occurs when BCOMMIO could not create the temporary file for printing. Remedy: check free disk space. On hard disk drive.
333		Terminal ok again	Informal: the entered terminal has sent a response to BCOMMIO. See also error code 302.
334		Error opening master record file	BCOMM cannot open the entered file for master records. Remedy: check whether the file entered in the order is existing in the correct path.

Error code	RT / RTM	Plain language text	Possible cause/remedy
335		Order file accessible again	Is reported if the order file is accessible again after error code 303
336		Existing order file could not be opened	The order file exists, but BCOMMIO could not open it within 5 seconds. Perhaps another application has access of the file. Contact the software partner (SWP).
337		File CInoHost could not be opened	Contact Kaba Benzing
338		File CInoHost could not be read	Contact Kaba Benzing
339		File CInoHost could not be written	Contact Kaba Benzing
340		File CInoHost could not be closed	Contact Kaba Benzing
341		9290 sub-terminals could not be logged on	In the configuration dialog of the LAN stream enable "9290 support", then retry logon
400		Data file not existing or defect	The file entered in the command could not be found by BCOMMIO. Remedy: check whether the file is existing in the correct path. If the file is not existing create it and start command again.
401		Terminal does not exist in configuration file	BCCOMMIO received an order from the application to send a record to a terminal that is not configured (not known) in the specified interface.  Remedy: In the log file analyse the corresponding data communications. Inform SWP.
403		No master records in file	BCCOMMIO can open the file contained in the order, but does not contain any master records for a terminal of the interface concerned.  Remedy: Check whether the correct file was specified in the order. If the file is generated by SWP, consult SWP. If the file is maintained manually, generate master records for the required terminals.
406		BCCOMMIO already active for the entered communication stream	BCCOMMIO was started on a communication stream where a BCOMMIO is already running.  Informal: communication is already active. Start of BCOMMIO is not necessary.
407		Error generating a semaphore	BCCOMMIO could not generate an internal semaphore. Contact Kaba Benzing.
408		Error creating security attributes	BCCOMMIO could not create security attributes. Contact Kaba Benzing.
410		CI: error creating the event semaphore	BCCOMMIO could not generate an internal semaphore for synchronisation with the CI program. Contact Kaba Benzing.
411		Bind port error	An error occurs during initializing of BETA adapters. Contact Kaba Benzing.
412		Error opening TCP/IP socket	An error occurred during initialization of a Beta adapter. Contact Kaba Benzing.

Error code	RT / RTM	Plain language text	Possible cause/remedy
500		Modem does not respond	<p>The modem configured for bcommio is not responding via the specified interface. By sending the AT command, bcommio checks whether a modem is present and expects a plain language message such as O.K. from the modem.</p> <p>Remedy: Check whether the modem used has been released for use with B-COMM. Check the modem parameter settings and, if necessary, redefine the settings with bcomm. The corresponding parameter files for released modems are supplied with bcomm (File: *.atm *.ats).</p>
502		Modem: error sending the parameter file	<p>BCOMMIO could not send the parameter for modem parameter setting from the file which was entered in the command to the modem.</p> <p>Remedy: check interface parameters and compare with the modem settings. Check interface pinning (RxD, TxD..) and modem parameter setting.</p>
503		Modem: parameter file does not exist	<p>In the modem parameter setting function, bcommio was not able to send the parameters from the file specified in the command to the modem because the specified parameter file could not be found. Check the file name. Bcommio searches for the parameter files in the main directory bcomm.</p>
504		Modem: cannot be set to command mode (Disconnection)	<p>To establish a connection, bcommio attempted to set the modem to command mode (AT command: +++), upon which the modem responded with not O.K.</p> <p>Remedy: Check the modem parameter settings. This error is a hint to a possible modem defect. Caution! Connections that have not been cleared prevent further functioning of the bcommio program. It is imperative to correct the error!</p>
505		Modem: rejects command for disconnection	<p>To clear a connection, bcommio attempted to set the modem to the command mode (AT command: +++), upon which the modem responded with O.K., but did not execute the ATH (clear connection) command.</p> <p>Remedy: In the log file, analyse the corresponding data communications. Check the modem parameter settings. This error is an indication of a possible modem defect. Caution! Connections that have not been cleared prevent further functioning of the bcommio program. It is imperative to correct the error!</p>
506		Modem: disconnection cannot be executed	<p>This message is informal and follows normally the errors 504 and 505.</p> <p>Remedy: see errors 504 and 505</p>

Error code	RT / RTM	Plain language text	Possible cause/remedy
509		Modem: does not transmit expected answer	The modem send unexpected responses to control commands from bcommio. Remedy: Check the modem parameter settings. Is the ATE0 (modem echo off) parameter set? In the log file, analyse the corresponding data communications.
510		Modem: RESET command could not be sent	The modem command ATZ could not be sent to the modem. Remedy: check interface parameters. Analyse the corresponding data communications In the file Log.<>.
511		Modem: RESET command could not be executed	The modem does not responds the command ATZ with OK. Remedy: check modem parameter settings. Analyse the corresponding data communications In the file Log.<>.
512		Modem: response not correct	The response sent from the modem after commands from BCOMMIO do not correspond to the AT instruction code. Remedy: check modem parameter settings. Analyse the corresponding data communications In the file Log.<>.
516		Invalid modem type has been entered (AT)	The modem type entered in the terminal configuration file TERMDEF.<> is not designated for using with BCOMMIO. Remedy: enter a correct modem type. Until now only AT modems are supported.
517		Baud rate has been modified	The transmission speed which has been configured for a interface, does not correspond to the baud rate which is reported from the terminal after successful connection setup (message CONNECT xxx). Remedy: check modem parameter setting.Set fix baud rate for the connection to the modem. Analyse the corresponding data communications in the file Log.<>. Repeat operation.
600		Received call (AT modem: RING)	Informal: BCOMMIO received the message RING from the local modem . The modem sends this message when it is called from a externally. Remedy. Check where the external came from. Use the modem manual for troubleshooting.
601		No receive signal (AT-Modem: NO CARRIER)	Bcommio was able to dial the phone number specified in the terminal definition file, but received not modem carrier from the distant station. Remedy: Check the modem parameter settings. Check the specified phone number to make sure it corresponds to the dialled connection. Check the modem documentation for possible causes. In the log file, analyse the corresponding data communications.

Error code	RT / RTM	Plain language text	Possible cause/remedy
6Ø2		Invalid command (AT-Modem: ERROR)	The AT command send by bcommio to the modem was not understood by the modem. Remedy: Check the modem parameter settings. Check the modem documentation for possible causes. In the log file, analyse corresponding data communications and analyse the AT command to which the modem responded with ERROR.
6Ø3		Dial tone not recognized (AT-Modem: NO DIAL TONE)	Bcommio was not able to dial the phone number specified in the terminal definition file because the modem did not receive an exchange tone (dial tone). Remedy: Check the modem parameter settings. Check the modem documentation for possible causes. In the log file, analyse corresponding data communications.
6Ø4		Busy Signal recognized (AT-Modem: BUSY)	Bcommio was able to dial the phone number specified in the terminal definition file, but the distant station was busy. Remedy: Check the modem parameter settings. Check the specified phone number to determine whether it corresponds to the dialled connection. Check the modem documentation for possible causes. In the log file, analyse corresponding data of communications. Repeat the operation.
6Ø5		No Answering Tone recognized (AT-Modem: NO ANSWER)	Bcommio was able to dial the phone number specified in the terminal definition file, but the distant station did not send a response tone. Remedy: Check the modem parameter settings. Check the specified phone number to determine whether it corresponds to the dialled connection. Check the modem documentation for possible causes. In the log file, analyse corresponding data of communications. Repeat the operation.
6Ø6		Disconnection via RTS (AT modem does not answer)	BCOMMIO could not instruct the modem via the command +++ATH to cancel the connection. So BCOMMIO has instructed the modem via the command RTS to hang up. Remedy: check wether a modem released by Kaba Benzing for use with B-COMM is connected. Maybe the modem has a fault. When the error occurs several times change the modem.
7ØØ		CI: CI program or converter could not be started! STOP!	BCOMMIO could not start the entered son process for the call interface. Processing is cancelled. Check program name and path via B-COMM.
7Ø1		CI: not existing terminal was adressed	BCOMMIO got the order from the application to send a data record to a terminal which is not configurated for the corresponding interface. Remedy: in the log file, analyse corresponding data of communications. Inform SWP.

Error code	RT / RTM	Plain language text	Possible cause/remedy
702		CI: timeout! STOP!	BCOMMIO has got no response from the CI program within the timeout defined in the configuration file. The communication is quit to prevent data loss. Check wether BCOMMIO is already active. If not so, start BCOMMIO and the CI program again. Check timeout and increase it, if the CI program needs longer for starting. Inform SWP.
703		CI: timeout with :STATUS:! STOP!	During contacting the CI program via :STATUS:/:OK: sequence BCOMMIO receives no :OK: response. BCOMMIO cancelled itself because the CI program could not be contacted. Check wether the CI program has been started and configured for the right communication stream.
704		CI: no connection to the CI! STOP!	Before the connection setup to the CI program via :STATUS:/:OK: sequence (see 703) BCOMMIO has detected via semaphores that the CI program could not be started. Remedy: check wether the CI program is configured for the right communication stream.
705		CI: event cannot be set/reset	Internal error in BCOMMIO: a event semaphore could not be set or reset. To prevent data loss BCOMMIO cancels itself. Remedy: start BCOMMIO and the CI program again. When this error occurs sveral times inform Kaba Benzing.
706		CI: unknown command received from CI	BCOMMIO has received an unkown command from the CI program. Remedy: call your SWP or Kaba Benzing.
707		CI: error writing into Shared Memory	During communication between BCOMMIO and a CI program an error occurred while writing into Shared Memory. Stop and restart communication. Conact Kaba Benzing..
708		CI: error reading from Shared Memory	An error occurs during communication between data transfer kernel and CI-program using shared memory. Stop and restart communication. Contact Kaba Benzing.
709		CI: error reading from Shared Memory	See error 708
710		CI: :STOPLINE: command received	Informal: BCOMMIO has received the :STOPLINE: command and has shut down
711		CI: STOPLINE-Event detected	Informal: the CI program has called the DLL function <i>ImmediateSTOPLINE</i> . BCOMMIO has shut down.
712		CI: received :NACK: 5 times as answer to a data record	When BCOMMIO receives :NACK: 5 times, it will shut down. Conact software partner (SWP)

Error code	RT / RTM	Plain language text	Possible cause/remedy
800		Order thread could not be started!	BCOMMIO could not start the thread for processing the order file. Check whether Windows system messages have been created and maintain any errors. Check error codes in the file BCOMMIO.ERR. Inform Kaba Benzing.
801		Thread for master record download could not be started!	BCOMMIO could not start the thread for processing the master record file. Check whether Windows system messages have been created and maintain any errors. Check error codes in the file BCOMMIO.ERR. Inform Kaba Benzing.
802		Wrong host IP address	BCOMMIO could not communicate with the host because during configuration of the LAN interface a wrong IP address has been entered. Check and correct the configuration of the communication stream with B-COMM or change IP address of the host.
900		Licence error! Please enter correct licence data! STOP!	Licence data is wrong! Call Kaba Benzing and check licence data.
901		Licence error! Licence file contains no data! STOP!	The licence file LICENCE.DAT contains no data. Enter licence data again via B-COMM. Inform Kaba Benzing.
902		Warning! End of licence authorization in a few days!	If you got a limited version of B-COMM in operation, this message means that the authorization is over in a few days. Call Kaba Benzing if you want prolongation or a unlimited version.
903		Licence error! End of licence authorization! STOP!	If you got a limited version of B-COMM in operation, this message means that the authorization is exceeded. Call Kaba Benzing if you want prolongation or a unlimited version.
904		Error opening the licence file!	The licence file LICENCE.DAT could not be opened. Check path settings via B-COMM. Maybe licence data must be entered again via B-COMM.
999			These are system messages from the FTP kernel BCOMMFTP.

**In the case of string fields, one character more is additionally reserved to accommodate the final ASCII „NUL“ character, i.e. the last character must always be the ASCII „NUL“ character. Fields or partials of fields that are not needed must also be assigned the ASCII „NUL“ character.**

---

## 7.5. Parameter File PARAM.DAT

All parameters that are requested by the REPARA (parameter upload, see chapter 3.4.5 Parameter setting) command are stored in the parameter file. The program number is entered in the first line. The data is overwritten if the parameter file already exist.

The name of the parameter file can be defined in the order file. If no file name is defined in the order file, the parameters are placed in the PARAM.DAT file.

### Example of a parameter file:

```
570-00-X-K08
@@0>3B01Mailbox-Text 1
@@0>3B02Mailbox-Text 2
@@0>3B03Mailbox-Text 3
@@0>3B04Mailbox-Text 4
@@0>3B05Mailbox-Text 5
@@0>3B06Mailbox-Text 6
@@0>3B07Mailbox-Text 7
@@0>3B08Mailbox-Text 8
```



---

## 7.6. Booking file BOOKING.DAT

Nearly all data received from the terminal is placed in the booking file. **The following records are NOT written into the booking file: error records (O records), status records (S records), alarm records (Xrecords), master records (Y records), up/download records (> records), AVISO records (i records) and G dialog records (G records).** This behaviour could be changed by setting the switch `write_all_data` of the section **[Others]** in the configuration file `BCOMM32.INI` to `Y`. Then all received records will be written into the booking file. The name of the booking file can be defined in the order file. The data is appended to the end of the file if the booking file already exists. The data is placed in the `BOOKING.DAT` or `BOOKING.<Stream>` file if no file name is defined in the order file.

### Example 1 of a booking file:

```
@@0A01920210160400031234570000
@@0A01920210160500031234570000
@@0A01920211115800031234570000
@@0A01920212080800031234570000
@@0A01920212094900031234570000
@@0A01920217104200031234570000
@@0A01920220134000031234570000
@@0A01920220163100031234570000
@@0A01920220163500031234570000
```

The data has been fetched from the terminal with the group/device address `00/00`.

If a line number has been assigned to a terminal, the line number is used in the booking file.

### Example 2 of a booking file:

```
0040B1192012209490>>>2:35012489
0040B21920122094900000005000033
```

The booking data has been fetched from the terminal with the line number `004`.

**The booking file could have the stream specific extension `<Stream>` if this is configured in the configuration dialog (see chapter 3.3.10 File definitions).**

---

## 8. Translation into other Languages

For adding another language some modifications must be made in the file LANGUAGE.INI file, which is copied into the B-COMM directory during installation of B-COMM WIN32. As a further language, e.g. Spanish shall be added:

[Global]	
count=3	Number of languages
title1=Sprache	Title of the dialogue box for german
text1=Wählen Sie die gewünschte Sprache	Text in the dialogue box for german
name1=Deutsch	Text of the radio button for german
icon1=flggerm.ico	Icon file for Germany
sysmess1=sysmess.ger	File with german system messages
title2=Language	Title of the dialogue box for english
text2=Select the desired language	Text in the dialogue box for english
name2=English	Text of the radio button for english
icon2=flguk.ico	Icon file for Great Britain
sysmess2=sysmess.eng	File with english system messages
title3=...	Title of the dialogue box for spanish
text3=...	Text of the dialogue box for spanish
name3=Espagnol	Text of the radio button for spanish
icon3=flgspain.ico	Icon file for Spain
sysmess3=sysmess.esp	File with spanish system messages

Afterwards, e.g. the section [Language1] must be copied and saved as [Language3]. Now the texts of [Language3] must be translated into spanish.

Wenn the menu item „Deutsch/English...” is selected the next time all 3 languages are displayed.

**A maximum of 5 further languages can be configured in the file LANGUAGE.INI in this manner.**

---

Icon files for the most common languages are delivered with B-COMM and are copied into the B-COMM directory during installation. There have to be two versions of the icon files:

in the **32x32** format, filename **flg<code>.ico**, where <code> is a country code of at max. 3 letters, e.g. SPA for Spain.

in the **16x16** format, filename **flg<code>16.ico**, where <code> is the same country code as in the filename of the 32x32 version.

---

## 9. Call Interface

The Call Interface is used for inter process communication between the communication kernel BCOMMIO and the appropriate CI program (which is written by a software partner (SWP)). The demo programs BCCIWIN, BCCITEST, OCXTEST and BCCIDEMO, which are supplied as Visual Basic 6.0 - and Visual C++ 6.0 sources with the B-COMM WIN32 communication program, show the basic concepts of this interface. The software partner can use these programs as a basis for his specific CI program. The CI program receives data records from BCOMMIO, sends these records in a variable manner to the application, e.g. through a data base, and generates an answer for BCOMMIO. It can also send any data records to BCOMMIO, which BCOMMIO sends to the connected terminal(s) and generates an answer for the CI program.

### 9.1. Shared Memory

The Call Interface in B-COMM WIN32 is realized with Shared Memory (a possibility of interprocess communication which is built in Windows ME and Windows NT/2000/XP), where each communication stream has its own Shared Memory area. There is only one 256-byte-buffer for both directions, so only one partner can write into that buffer at a time. The opposite side must read the buffer before it can write to the Shared Memory itself.

Cardinally each record has to be acknowledged by the opposite side before the next record can be sent.

A CI program can communicate with multiple BCOMMIOs or there is a CI program for each active BCOMMIO.

Functions for use of the Shared Memory are available in two ways: first in a DLL (Dynamic Link Library) called BCWIN.DLL (see chapter 9.2 DLL functions), second by an ActiveX control (see chapter 9.4 The Active-X Control CI\_OCX) called CI\_OCX.OCX. Both are delivered with the B-COMM WIN32 package and are copied into the system directory of Windows during the installation of B-COMM WIN32.

---

## 9.2. DLL functions

The DLL contains the following functions, which can be used by the CI program (the syntax shows the function declarations for the programming language C):

### 9.2.1. SetCIHandle

this function has to be called by the CI program immediately after it is started in order to receive the Windows messages (see below) from BCOMMIO.

**void SetCIHandle (HANDLE CIHandle, int MsgID, int StreamNo);**

CIHandle	Handle (number) of the CI program, to which BCOMMIO should send the messages
MsgID	ID (number) of the message to send (default value is 513 dec.)
StreamNo	number of the corresponding stream

### 9.2.2. SetImmDataHandle

this function has to be called by the program immediately after it is started in order to receive the Windows messages IMM\_DATA\_ACK and IMM\_DATA\_NACK (see below) from BCOMMIO.

**void SetImmDataHandle (HANDLE ImmDataHandle, int MsgID, int StreamNo);**

ImmDataHandle	Handle (number) of the program, to which BCOMMIO should send the messages
MsgID	ID (number) of the message to send (default value is 513 dec.)
StreamNo	number of the corresponding stream

### 9.2.3. InitCISharedMem

This function has to be called by the CI programm to establish the connection to BCOMMIO. The function reads out the initialization string :STATUS:, which was written into Shared Memory by BCOMMIO. The functions waits a maximum of the CI timeout (default 60 seconds) for that record. After that, the CI program has to write :OK: into Shared Memory by use of the function *WriteCISharedMem* (see below).

**int InitCISharedMem (int StreamNo);**

StreamNo	number of the corresponding stream
Return value	Ø = :STATUS: received from BCOMMIO within the

---

CI timeout

- 1 = BCOMMIO not active or it's active but still initializing (this could last up to 90 seconds)
- 2 = an internal event could not be opened
- 3 = an internal event could not be opened
- 4 = BCOMMIO didn't write anything into Shared Memory during the timeout
- 5 = BCOMMIO has written something into Shared Memory, but not :STATUS:
- 6 = an internal event could not be set/reset

---

The function either returns immediately with one of the return values -1, -2, -3 or -6, or, after the connection to BCOMMIO was successful, with the return value 0, or after the CI timeout with one of the values -4, -5 or -6. Only when the return value is -1 (BCOMMIO not active, this function should be called again. When a return value of -2, -3, -4, -5 or -6 occurred, serious internal errors occurred which prevent a connection to BCOMMIO.

**Attention: the connection to BCOMMIO can last up to 90 seconds, when on a LAN stream a Beta is not available during startup of BCOMMIO. So it's possible, that the function *InitCISharedMem* will return -1 for up to 90 seconds.**

#### 9.2.4. ExitCISharedMem

This function has to be called by the CI program before it shuts down in order to release the Shared memory area.

**int ExitCISharedMem (int StreamNo);**

StreamNo                      number of the corresponding stream

Return value                always 0

**Attention:** You should not call any function of the DLL (except *SetCIHandle* and *InitCISharedMem*) after you called this function, because BCOMMIO has no more access to the Shared Memory.

#### 9.2.5. ReadCISharedMem

This function is used to read out data records or acknowledgements that BCOMMIO has written into Shared Memory.

**int ReadCISharedMem (char \*data, int StreamNo);**

\*data                      pointer to the buffer to which the received data should be copied. The buffer must be able to receive up to 256 bytes. When the received record is less than 256 bytes in length, the rest of the buffer is filled with NULL bytes.

StreamNo                number of the corresponding stream

Return value                0 = no data in Shared Memory  
-1 = error while reading from Shared Memory  
-2 = error occurred when resetting an internal event  
-3 = BCOMMIO is currently writing into Shared Memory, the function has to be called again later  
> 0 = number of the received bytes

If you receive an error -1 or -2, then you should shut down BCOMMIO

---

and call the function *ExitCISharedMem*. Then you can either end your program or call *InitCISharedMem*.



---

### 9.2.6. WriteCISharedMem

With this function the CI program can write data into Shared Memory.

**int WriteCISharedMem (char \*data, int StreamNo);**

*data	pointer to the buffer which contains the data to write. The buffer must not contain more than 255 bytes, it has to be terminated by a NULL byte. If the buffer contains more than 256 bytes, the rest is truncated.
StreamNo	number of the corresponding stream
Return value	Ø = data record written without error -1 = error while writing into Shared Memory -2 = data record was not written because the Shared Memory is occupied -3 = the buffer was empty

When receiving the error -1, you should shutdown BCOMMIO and call the function *ExitCISharedMem*. Then you can either end your program or call *InitCISharedMem*.

When receiving the error -3, then you passed an empty buffer to the function. Fill the buffer correctly and then call the function again. Receiving the error -2 can have two reasons:

- BCOMMIO has not yet read out the most recent data record from the CI program. You then should wait a little while (e.g. until you received the message SM\_READ) and then retry to write the data record into the Shared Memory.

- BCOMMIO has written a data record into Shared Memory. You should first read out that record by use of *ReadCISharedMem*, answer that record to BCOMMIO (either by :ACK: or an ONLINE answer) and then retry to write the desired data record.

### 9.2.7. GetBCOMMIOStatus

By use of this function, the CI program can check whether BCOMMIO is still active.

**BOOLEAN GetBCOMMIOStatus (int StreamNo);**

StreamNo	number of the corresponding stream
return value	FALSE = BCOMMIO is not active TRUE = BCOMMIO is active

---

### 9.2.8. ImmediateSTOPLINE

With this functions BCOMMIO can be terminated „immediately“ under some circumstances. Immediate means, that a currently executed order is not terminated but executed until it is ready. If BCOMMIO is currently waiting for the answer to a data record, it writes this data record into a file called CINOHOST.<x > and then shuts down. When BCOMMIO is restarted, the contents of that file is re-sent to the CI program, so no bookings are lost.

**void ImmediateSTOPLINE (int StreamNo);**

StreamNo                      number of the corresponding stream

### 9.2.9. SendImmediateData

With this function the CI program can send an “asynchronous” data record to BCOMMIO. BCOMMIO then sends the data record to the appropriate terminal.

**BOOLEAN SendImmediateData (char data, int StreamNo);**

\*data                          pointer to the data record to send  
StreamNo                      number of the corresponding stream

Return value                  TRUE if the data record could be copied into the internal buffer, else FALSE

As an answer to this function, BCOMMIO sends one of two possible Windows messages to the CI program: *IMM\_DATA\_ACK* or *IMM\_DATA\_NACK* (see chapter 9.5 Messages)

**Attention:** You will not receive an answer like :ACK: via the CI when sending a data record with this function! If you send more than one record by use of this function without waiting for the *IMM\_DATA\_ACK* message between the records, it could happen, that you overwrite a former record and that a record will never reach BCOMMIO.

---

### 9.3. Program flow in a CI program

In a CI application, the program flow could be realized as follows (please keep in mind that it's only an example and therefore is lacking some parts):

1. call of function *SetCIHandle()*
2. call of function *SetImmDataHandle*

this function has to be called by the program immediately after it is started in order to receive the Windows messages *IMM\_DATA\_ACK* and *IMM\_DATA\_NACK* (see below) from *BCOMMIO*.

**void SetImmDataHandle (HANDLE ImmDataHandle, int MsgID, int StreamNo);**

*ImmDataHandle*            Handle (number) of the program, to which  
                              *BCOMMIO* should send the messages  
*MsgID*                    ID (number) of the message to send  
                              (default value is 513 dec.)  
*StreamNo*                number of the corresponding stream

3. *InitCISharedMem()*
4. return value 0 => go to item 4, else back to item 2  
Attention: it could last up to 90 seconds until the return value 0 is reached when using a LAN stream where a BETA is not available
5. write :OK: into Shared Memory by use of function  
*WriteCISharedMem()*

now the CI program can run in a loop to receive data records from *BCOMMIO* respectively the terminals or to send data records or commands to *BCOMMIO* and the terminals

6. call of function *ReadCISharedMem()*
7. return value 0 => go to item 5  
return value > 0 => continue with item 7
8. interpretation of the data record and send according answer by use of function *WriteCISharedMem()* (:ACK: or Online answer or other answer)
9. check if a data record has to be sent to a terminal; if not => back to item 5
10. write data record into Shared Memory by a call of function  
*WriteCISharedMem()*
11. return value 0 => continue with item 11  
return value -2 => buffer is occupied by a data record of *BCOMMIO* or a terminal => back to item 5 to empty the buffer, then you can try again to send the record
12. call of function *ReadCISharedMem()*
13. return value 0 => back to item 11  
return value > 0 => answer received => continue with item 13
14. should the application end?  
no => back to item 5  
yes => call of function *ExitCISharedMem()* and exit

The example program flow uses the polling method, i.e. it checks periodically whether a data record from *BCOMMIO* or a terminal is present by calling

---

function *ReadCISharedMem()*. Another possibility is the usage of an event handler, which reacts to the messages sent by BCOMMIO (see chapter 9.5 Messages).

## 9.4. The Active-X Control CI\_OCX

This ActiveX control was designed to make the use of the Shared Memory interface between BCOMMIO and a Call Interface program as easy as possible. It also gives the programmer some possibilities to display the status of BCOMMIO and the connected terminals.

It internally uses the functionality of the dynamic link library BCWIN.DLL (see chapter 9.2 DLL functions) which a software developer can also use to transfer data between BCOMMIO and the application program.

Following you can see a snapshot of the demo project OCXDEMO (for Visual Basic 6. 0):

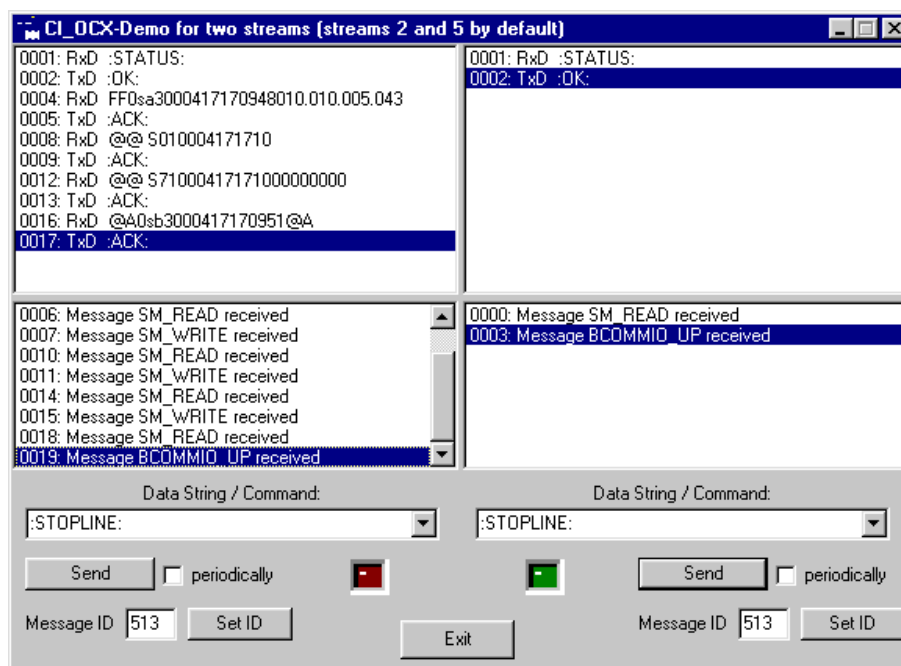


Figure 9-1: Demo program OCXDEMO

In the lower left corner of the window you can see two little LEDs which, in this case, represents the OCX (see property [ShowAsLED](#)). The left stream currently has one terminal, which is not available, the right stream is running without errors.

---

## 9.4.1. Properties

### 9.4.1.1 AutoAcknowledge

**AutoAcknowledge** (*True* or *False*, default = *False*)

*True*: the ActiveX-control automatically sends an `:ACK:` to BCOMMIO when a data record of record type "X", ">", "Y", "S" or "O" was received (independent of the operating mode). The property OnlineBooking is set to *False* in this case.

All other record types are not auto-acknowledged. If the operating mode of the data record is Online, the property OnlineBooking is set to *True*, otherwise it is set to *False*. So the programmer of the CI program has to decide whether an Online answer has to be sent to the Terminal or the record just has to be acknowledged with `:ACK:`.

*False*: the ActiveX control does not send any acknowledge record to BCOMMIO by itself. The programmer of the CI program has to send all responds to BCOMMIO.

### 9.4.1.2 AutoDisconnect

**AutoDisconnect** (*True* or *False*, default = *False*)

*True*: the ActiveX control automatically calls the method DisconnectFromBCOMMIO (which itself calls the DLL function ExitCISharedMem) when it receives the `BCOMMIO_DOWN` message.

*False*: the ActiveX control does not disconnect from BCOMMIO by itself. The programmer of the CI program has to call the method DisconnectFromBCOMMIO when the event evBCOMMIODown occurs.

### 9.4.1.3 BlinkOnError

**BlinkOnError** (*True* or *False*, default = *True*)

*True*: when the properties ShowAsLED and ShowMissingTerminal also have the value *True*, then the LED blinks when at least one terminal reports an error

*False*: the does not blink if a terminal reports an error

---

#### 9.4.1.4 BlinkInterval

**BlinkInterval** (integer value, default = 1000)

Specifies the interval in milliseconds in which the LED blinks when the property BlinkOnError has the value *True*.

#### 9.4.1.5 BCOMMIOStatus

**BCMMIOStatus** (*True* or *False*, readonly)

This readonly property represents the state of BCOMMIO, where *True* means that BCOMMIO is active (see also method ReadBCMMIOStatus).

#### 9.4.1.6 CLError

**CErrror** (integer value, readonly)

In case one of the methods ConnectToBCMMIO or SendDataToBCMMIO returns *False*, you can get the error number by reading this readonly property.

#### 9.4.1.7 ConnectInterval

**ConnectInterval** (integer value, default = 1000)

Specifies the interval in milliseconds in which the ActiveX control tries a connection to BCOMMIO (see also property ConnectRetries).

#### 9.4.1.8 ConnectRetries

**ConnectRetries** (integer value, default = 12)

Specifies the number of retries with which the ActiveX control tries to connect to BCOMMIO, which means to read out *:STATUS:* from the Shared Memory (see also property ConnectInterval).

---

#### 9.4.1.9 DebugLevel

**DebugLevel** (integer value from 0 to 9, default = 2)

Specifies the debug level of BCOMMIO. The higher this value is, the more information is output in the window of BCOMMIO. A value higher than 2 is usually only necessary when problems occur during the communication. A value from 6 to 9 writes (besides the debug files) a special file called BCWINDLL.<x>, which grows rapidly.

This value is passed to BCOMMIO as a commandline parameter when BCOMMIO is started by the ActiveX control (see property [StartBCOMMIO](#)).

#### 9.4.1.10 LEDShape

**LEDSHAPE** (values „as Circle“ and „as Square“, default = „as Square“)

When the property [ShowAsLED](#) is set to *True*, the shape of the LED can be set by this property.

#### 9.4.1.11 MessageID

**MessageID** (integer value, default = 513)

This property defines the message ID that BCOMMIO uses to send its messages (see description of DLL function *SetCIHandle* in the manual or in the Online Help of B-COMM WIN32).

#### 9.4.1.12 OnlineBooking

**OnlineBooking** (*True* or *False*, readonly)

This readonly property is set to *True* by the ActiveX control when it received an Online booking from BCOMMIO, otherwise it is set to *False*.



---

#### 9.4.1.13 SendInterval

This property exists only for compatibility reasons to former versions of the control, but it has no functionality.

#### 9.4.1.14 SendRetries

This property exists only for compatibility reasons to former versions of the control, but it has no functionality.

#### 9.4.1.15 ShowAsLED

**ShowAsLED** (*True* or *False*, default = *True*)

When this property is set to *True*, the ActiveX control is displayed as an LED during design time and runtime of the CI program. The LED represents the state of BCOMMIO with three different colors:

Yellow: BCOMMIO is not active

Green: BCOMMIO is initialized and running without any terminal error

Red: during initialization of BCOMMIO and (if the property ShowMissingTerminal is set to *True*) when any terminal connected to the Stream is not available.

Additionally, when receiving or sending a data record, the color changes from dark green/red to light green/red for a short time.

The shape of the LED can be set with the property LEDShape.

When the property is set to *False*, the ActiveX control is displayed with the B-COMM Icon during design time and is invisible during runtime.

#### 9.4.1.16 ShowMissingTerminal

**ShowMissingTerminal** (*True* or *False*, default = *False*)

When this property is set to *True* and the ActiveX control is displayed as a LED (see property ShowAsLED), then the control periodically checks the state of all connected terminals. When at least one of the terminals is not available, the color of the LED changes from green to red. The interval can be set by the property UpdateInterval. Additionally you can specify whether the LED should blink in case of an error or not and, if yes, in which interval, by use of the properties BlinkOnError and BlinkInterval.

#### 9.4.1.17 ShowTrafficlight

---

**ShowTrafficlight** (*True* or *False*, default = *False*)

When the window size of BCOMMIO (see property WindowSize) is set to *Hidden*, BCOMMIO can be represented by a traffic light in the system tray of Windows when this property is set to *True*.

#### 9.4.1.18 StartBCOMMIO

**StartBCOMMIO** (*True* or *False*, default = *True*)

When this property is set to *True*, the method ConnectToBCOMMIO automatically starts BCOMMIO with commandline parameters given by the properties Stream, DebugLevel, WindowSize and ShowTrafficlight.

#### 9.4.1.19 Stream

**Stream** (values "*Stream 0*" to "*Stream 29*", default = "*Stream 0*")

This property defines the stream number that BCOMMIO uses.

#### 9.4.1.20 UpdateInterval

**UpdateInterval** (integer value, default = *1000*)

Defines the interval in milliseconds, with which the ActiveX control checks the state of the connected terminals, when the properties ShowMissingTerminal and ShowAsLED are set to *True*.

#### 9.4.1.21 WindowSize

**WindowSize** (values „*Hidden*“, „*Standard with Focus*“ (default), „*Minimized with Focus*“, „*Standard without Focus*“ and „*Minimized without Focus*“)

Specifies the window size of BCOMMIO's window when it is started by the ActiveX control (see property StartBCOMMIO)

---

## 9.4.2. Methods

### 9.4.2.1 ConnectToBCOMMIO

Boolean **ConnectToBCOMMIO** (no parameters)

This method tries to establish a connection to BCOMMIO via the Shared Memory interface. It waits a given time (see properties ConnectInterval and ConnectRetries) for the `:STATUS:` string of BCOMMIO and then answers with `:OK:`).

When this method returns *False*, the property CLError contains one of the following error codes:

- |        |   |
|--------|---|
| -1     | BCOMMIO has not started   |
| -2, -3 | an internal error occurred in the Shared Memory so that the connection failed |
| -4     | BCOMMIO has not answered (written <code>:STATUS:</code> )                     |
| -5     | BCOMMIO has answered, but not with <code>:STATUS:</code>                      |
| -6     | an internal error occurred in the Shared Memory so that the connection failed |
| -7     | BCOMMIO has shut down during connection                                       |
| -8     | BCOMMIO could not be started  |

Internally this method calls the DLL function *InitCISharedMemory*.

### 9.4.2.2 DisconnectFromBCOMMIO

**DisconnectFromBCOMMIO** (no parameters)

This method internally calls the DLL function *ExitCISharedMem* and so releases the Shared Memory used by the former connection.

### 9.4.2.3 ReadBCOMMIOStatus

Boolean **ReadBCOMMIOStatus** (no parameters)

This method returns the state of BCOMMIO. The return value is *False* if BCOMMIO is not running and *True* when BCOMMIO is active (see also property BCOMMIOStatus). It internally calls the DLL function *GetBCOMMIOStatus*.

---

#### 9.4.2.4 SendDataToBCOMMIO

Boolean **SendDataToBCOMMIO** (parameter: data record)

This method sends a given data record to BCOMMIO by use of the DLL function *WriteCISharedMem*.

When this method returns *False*, the property CLError contains one of the following error codes:

- 1 an error occurred during sending the data record
- 2 the data record could not be sent because the Shared Memory was occupied
- 3 the given data record was empty

#### 9.4.2.5 ShutdownBCOMMIO

**ShutdownBCOMMIO** (no parameter)

This method calls the DLL function *ImmediateSTOPLINE*. You can use this method, when BCOMMIO has sent a data record and you are currently not able to answer. BCOMMIO then writes the data record into the file CINOHOST.<Stream> and shuts down. When BCOMMIO is restarted, it first sends the data record(s) contained in that file to the CI program.

#### 9.4.2.6 SendImmediateDataToBCOMMIO

Boolean **SendImmediateDataToBCOMMIO** (parameter: data record)

This method sends the given data record to BCOMMIO by calling the DLL function *SendImmediateData* (see chapter 9.2.9 SendImmediateData).

---

## 9.4.3. Events

### 9.4.3.1 evBCOMMIODown

**evBCOMMIODown** (no parameter)

This event occurs when BCOMMIO has sent the message *BCOMMIO\_DOWN* (see chapter 9.5.3 *BCOMMIO\_DOWN*). When the property *AutoDisconnect* is set to *True*, then the ActiveX control automatically calls the method *DisconnectFromBCOMMIO*, otherwise you should call that method in this event handler.

### 9.4.3.2 evBCOMMIORead

**evBCOMMIORead** (no parameter)

This event occurs when BCOMMIO has read out the data record, that the CI program has written into Shared Memory and has sent the message *SM\_READ* (see chapter 9.5.2 *SM\_READ*).

### 9.4.3.3 evBCOMMIOStatus

**evBCOMMIOStatus** (parameter: status)

This event occurs when BCOMMIO has sent the message *BCOMMIO\_STATUS* (see chapter 9.5.5 *BCOMMIO\_STATUS*) as an answer to a STATUS command. The parameter status can have one of the following values:

- 1 terminal is logged on but currently not available
- Ø terminal is not logged on
- 1 terminal is logged on and is available
- 2 BCOMMIO is active

### 9.4.3.4 evBCOMMIUp

**evBCOMMIUp** (no parameter)

This event occurs when BCOMMIO has started and initialized and then has sent the message *BCOMMIO\_UP* (see chapter 9.5.4 *BCOMMIO\_UP*).

---

#### 9.4.3.5 evDataRecordReceived

**evDataRecordReceived** (parameter: received data record)

This event occurs when BCOMMIO has sent the *SM\_WRITE* (see chapter 9.5.1 *SM\_WRITE*) message and the ActiveX control has read out the data record from the Shared Memory. The data record is passed to this event in its parameter.

In this event handler you can analyze the received data record and you have to answer to it either with *:ACK:* (see also property AutoAcknowledge) or with an Online answer (see also property OnlineBooking).

#### 9.4.3.6 evBCOMMIOOrderStart

**evBCOMMIOOrderStart** (parameter: order no.)

This event occurs when BCOMMIO has sent the message *ORDER\_START* (see chapter 9.5.8 *ORDER\_START*).

#### 9.4.3.7 evBCOMMIOOrderEnd

**evBCOMMIOOrderEnd** (parameter: order no.)

This event occurs when BCOMMIO has sent the message *ORDER\_END* (see chapter 9.5.9

---

ORDER\_END).

#### **9.4.3.8 evBCOMMIOOrderRestart**

**evBCOMMIOOrderRestart** (parameter: order no.)

This event occurs when BCOMMIO has sent the message *ORDER\_REP\_START* (see chapter 9.5.10 ORDER\_REP\_START).

---

#### **9.4.3.9 evImmediateDataAck**

**evImmediateDataAck** (no parameter)

This event occurs when BCOMMIO has sent the message *IMM\_DATA\_ACK* (see chapter 9.5.6 *IMM\_DATA\_ACK*).

#### **9.4.3.10 evImmediateDataNack**

**evImmediateDataNack** (no parameter)

This event occurs when BCOMMIO has sent the message *IMM\_DATA\_NACK* (see chapter 9.5.7



---

IMM\_DATA\_NACK).

---

## 9.5. Messages

As mentioned above, BCOMMIO sends various Windows messages, which can be received and analyzed by a CI program. These messages are designated to reduce the system load, because a CI program has not to call the function *ReadCISharedMem* periodically to check whether BCOMMIO has sent a data record. Instead, the CI program waits for the message SM\_WRITE and then calls the function *ReadCISharedMem* once to read out the data record.

Something general about Windows messages:

a Windows message consists of 4 parameters:

the handle of the destination, to which the message should be sent (this is just a number which is managed by the operating system)

the ID of the message; this is also a number, which the software developer can specify „freely“. The area 0 to 4000 hex is reserved for use by the operating system. BCOMMIO uses the message ID 513 (decimal) by default, so that the resulting ID is 4000hex + 513dec = 1537 dec.

parameters 3 and 4 are available for the software developer (32 bit each) The 1<sup>st</sup> of these parameters is used to identify the message. It contains a number, which represents the cause of the message (e.g. SM\_WRITE (1), when BCOMMIO has written something into Shared Memory). The 2<sup>nd</sup> parameter is always used for the stream number.

Such messages can be sent by use of two Windows API functions: *SendMessage/SendMessageTimeout* or *PostMessage*. The difference is, that *SendMessage/SendMessageTimeout* waits until a process has received and processed the message. BCOMMIO uses *SendMessageTimeout* (for an explanation of the various function parameters see the Windows API documentation). Because of that, in the event handler of the CI program no DLL functions should be called. Instead you can set flags in the event handler and react to these flags in a main loop.

---

The messages of BCOMMIO:

### 9.5.1. SM\_WRITE

**SendMessageTimeout (CI\_HWND, WM\_USER+xxx, SM\_WRITE, StreamNo, flags, timeout, NULL);**

CI_HWND	handle of the receiving CI program (this handle can be forwarded to BCOMMIO by the function <i>SetCIHandle</i> (see above)
WM_USER+xxx	ID of the message (xxx is 513 decimal by default). This value is forwarded to BCOMMIO also by <i>SetCIHandle</i>
SM_WRITE	1
StreamNo	number of the corresponding stream
flags	SMT0_ABORTIFHUNG   SMT0_NORMAL
timeout	the CI timeout
NULL	unused parameter

This message is sent by BCOMMIO (exactly: automatically in the DLL) when it has written a data record into Shared Memory. Only during the connection phase (when BCOMMIO writes *:STATUS:* into Shared Memory), the message is NOT generated.

### 9.5.2. SM\_READ

**SendMessageTimeout (CI\_HWND, WM\_USER+xxx, SM\_READ, StreamNo, flags, timeout, NULL);**

CI_HWND	handle of the receiving CI program (this handle can be forwarded to BCOMMIO by the function <i>SetCIHandle</i> (see above)
WM_USER+xxx	ID of the message (xxx is 513 decimal by default). This value is forwarded to BCOMMIO also by <i>SetCIHandle</i>
SM_READ	5
StreamNo	number of the corresponding stream
flags	SMT0_ABORTIFHUNG   SMT0_NORMAL
timeout	the CI timeout
NULL	unused parameter

This message is sent by BCOMMIO (exactly: automatically in the DLL) when it has read out a data record from Shared Memory.

---

### 9.5.3. BCOMMIO\_DOWN

**SendMessageTimeout (CI\_HWND, WM\_USER+xxx, BCOMMIO\_DOWN, StreamNo, flags, timeout, NULL);**

CI_HWND	handle of the receiving CI program (this handle can be forwarded to BCOMMIO by the function <i>SetCIHandle</i> (see above)
WM_USER+xxx	ID of the message (xxx is 513 decimal by default). This value is forwarded to BCOMMIO also by <i>SetCIHandle</i>
BCOMMIO_DOWN	2
StreamNo	number of the corresponding stream
flags	SMTO_ABORTIFHUNG   SMTO_NORMAL
timeout	the CI timeout
NULL	unused parameter

This message is sent by BCOMMIO just before it shuts down, when the CI programm is still running.

### 9.5.4. BCOMMIO\_UP

**SendMessageTimeout (CI\_HWND, WM\_USER+xxx, BCOMMIO\_UP, StreamNo, flags, timeout, NULL);**

CI_HWND	handle of the receiving CI program (this handle can be forwarded to BCOMMIO by the function <i>SetCIHandle</i> (see above)
WM_USER+xxx	ID of the message (xxx is 513 decimal by default). This value is forwarded to BCOMMIO also by <i>SetCIHandle</i>
BCOMMIO_UP	3
StreamNo	number of the corresponding stream
flags	SMTO_ABORTIFHUNG   SMTO_NORMAL
timeout	the CI timeout
NULL	unused parameter

This message is sent by BCOMMIO after it has started and initialized the corresponding interface (COM or LAN).

---

### 9.5.5. BCOMMIO\_STATUS

**SendMessageTimeout (CI\_HWND, WM\_USER+xxx, Param1, StreamNo, flags, timeout, NULL);**

CI_HWND	handle of the receiving CI program (this handle can be forwarded to BCOMMIO by the function <i>SetCIHandle</i> (see above)
WM_USER+xxx	ID of the message (xxx is 513 decimal by default). This value is forwarded to BCOMMIO also by <i>SetCIHandle</i>
Param1	in the lower 16 bits of the parameter: 4 (BCCOMMIO_STATUS) in the upper 16 bit: Ø = terminal is not logged on 1 = terminal is logged on and available -1 = terminal is logged on but not available
StreamNo	number of the corresponding stream
flags	SMTO_ABORTIFHUNG   SMTO_NORMAL
timeout	the CI timeout
NULL	unused parameter

This message is sent by BCOMMIO as an answer to the status command :*STATUS\_xy*..

### 9.5.6. IMM\_DATA\_ACK

**SendMessageTimeout (CI\_HWND, WM\_USER+xxx, IMM\_DATA\_ACK, StreamNo, flags, timeout, NULL);**

CI_HWND	handle of the receiving CI program (this handle can be forwarded to BCOMMIO by the function <i>SetCIHandle</i> (see above)
WM_USER+xxx	ID of the message (xxx is 513 decimal by default). This value is forwarded to BCOMMIO also by <i>SetCIHandle</i>
IMM_DATA_ACK	7
StreamNo	number of the corresponding stream
flags	SMTO_ABORTIFHUNG   SMTO_NORMAL
timeout	the CI timeout
NULL	unused parameter

This message is sent by BCOMMIO as an answer to the call of the function *SendImmediateData*, if the data record was sent successfully (see chapter 9.2.9 *SendImmediateData*)

---

### 9.5.7. IMM\_DATA\_NACK

**SendMessageTimeout (CI\_HWND, WM\_USER+xxx, IMM\_DATA\_NACK, StreamNo, flags, timeout, NULL);**

CI_HWND	handle of the receiving CI program (this handle can be forwarded to BCOMMIO by the function <i>SetCIHandle</i> (see above)
WM_USER+xxx	ID of the message (xxx is 513 decimal by default). This value is forwarded to BCOMMIO also by <i>SetCIHandle</i>
IMM_DATA_ACK	7
StreamNo	number of the corresponding stream
flags	SMTO_ABORTIFHUNG   SMTO_NORMAL
timeout	the CI timeout
NULL	unused parameter

This message is sent by BCOMMIO as an answer to the call of the function *SendImmediateData*, if the transmission of the data record failed (see chapter 9.2.9 *SendImmediateData*)

### 9.5.8. ORDER\_START

**SendMessageTimeout (CI\_HWND, WM\_USER+xxx, Param1, StreamNo, flags, timeout, NULL);**

CI_HWND	handle of the receiving CI program (this handle can be forwarded to BCOMMIO by the function <i>SetCIHandle</i> (see above)
WM_USER+xxx	ID of the message (xxx is 513 decimal by default). This value is forwarded to BCOMMIO also by <i>SetCIHandle</i>
Param1	in the lower 16 bit of this parameter: 9 ( <i>ORDER_START</i> ), in the upper 16 Bit record no. of the order within the order file, starting with Ø
StreamNo	number of the corresponding stream
flags	SMTO_ABORTIFHUNG   SMTO_NORMAL
timeout	the CI timeout
NULL	unused parameter

This message is sent by BCOMMIO when it starts executing an order.

---

### 9.5.9. ORDER\_END

**SendMessageTimeout (CI\_HWND, WM\_USER+xxx, Param1, StreamNo, flags, timeout, NULL);**

CI_HWND	handle of the receiving CI program (this handle can be forwarded to BCOMMIO by the function <i>SetCIHandle</i> (see above)
WM_USER+xxx	ID of the message (xxx is 513 decimal by default). This value is forwarded to BCOMMIO also by <i>SetCIHandle</i>
Param1	in the lower 16 bit of this parameter: 10 ( <i>ORDER_END</i> ), in the upper 16 Bit record no. of the order within the order file, starting with 0
StreamNo	number of the corresponding stream
flags	SMTO_ABORTIFHUNG   SMTO_NORMAL
timeout	the CI timeout
NULL	unused parameter

This message is sent by BCOMMIO when it stops executing an order.

### 9.5.10. ORDER\_REP\_START

**SendMessageTimeout (CI\_HWND, WM\_USER+xxx, Param1, StreamNo, flags, timeout, NULL);**

CI_HWND	handle of the receiving CI program (this handle can be forwarded to BCOMMIO by the function <i>SetCIHandle</i> (see above)
WM_USER+xxx	ID of the message (xxx is 513 decimal by default). This value is forwarded to BCOMMIO also by <i>SetCIHandle</i>
Param1	in the lower 16 bit of this parameter: 11 ( <i>ORDER_REP_START</i> ), in the upper 16 Bit record no. of the order within the order file, starting with 0
StreamNo	number of the corresponding stream
flags	SMTO_ABORTIFHUNG   SMTO_NORMAL
timeout	the CI timeout
NULL	unused parameter

This message is sent by BCOMMIO when it starts the repetition of an order. Therefore the switch **extended\_orders** in the section **[Others]** of the configuration file BCOMM32.INI has to be set to **Y** (see chapter 7.2 Configuration file BCOMM32.INI).

---

## 9.6. DDE

Additionally a converter program (DDE2SMEM.EXE) is supplied, which converts DDE communication to Shared Memory communication. For the use of DDE as the interface to CI programs another demo program is supplied with B-COMM WIN32 with its source code (BCCI\_DDE).

Attention: this kind of DDE support was implemented only to be compatible to B-COMM /WIN SMS. Existing 16-bit CI programs, which have been written for use with B-COMM /WIN SMS, can just be ported to 32-bit without changing the interface to B-COMM.

When you are developing new CI programs, you should prefer Shared Memory, because it's faster and more reliable than DDE.

## 9.7. Protocol

**EACH** record, whether sended from BCOMMIO to the CI program or sended from the CI program to BCOMMIO, **must** be acknowledged by the other side.

Data records from BCOMMIO to the CI program, must be acknowledged by the CI program either with **:ACK:** or (e.g. Online bookings) with a **data record** (e.g. R1 record). If not, or if the CI program answeres five times with **:NACK:**, BCOMMIO writes the record into a special file named **CINOHOST.<Stream>** and then shuts down (exception: the switch send\_enq in the configuration file BCOMM32.INI is set to Y. See also chapter 7.2 Configuration file BCOMM32.INI). During the next start BCOMMIO checks whether the file CINOHOST.<stream> existst and, if yes, sends the contents of the file to the CI program.

Data records, which the CI program sends to BCOMMIO, will also be acknowledged with **:ACK:** or another answer (see below).

In general all data records (parameter records, display records and so on) could be sended by the CI program to BCOMMIO. BCOMMIO sends these records to the corresponding terminal. **An exception are terminals, which are connected via a modem. To such terminals only the data record for setting date/time to all terminals could be sended, e.g. \*\*0U1030122085000.**

To establish a connection to a terminal connected via modem, there must be an order in the order file (see also chapter 7.3 Order file ORDER.<Stream>). This order could be created either by the CI program itself or the CI program can send an order by use of the CI command (see below, command **:ORDER...:**)



---

## 9.8. Commands from the CI Program to BCOMMIO

Additionally to „normal“ data records, the CI program can send commands to BCOMMIO. These commands must have a colon at their start and end. The following commands will be recognized by BCOMMIO:

<b>:BPASTOP:</b>	The polling of Online- or Offline terminals is stopped. Therefore the terminals will no longer receive any data record sent by the CI program. This command is only useful with serial communication because in a LAN configuration the polling is done by the BETAs and could not be stopped.
<b>:BPASTART:</b>	The polling of Online- or Offline terminals is restarted so that the terminals will receive data records again. This is also only possible with serial communication.
<b>:CISTOP:</b>	Data records from the terminal(s) will be received and written into the booking file (when enabled in the BCOMM32.INI), but not sent to the CI program. Before sending this command all ONLINE terminals should be set to OFFLINE or AUTONOMOUS.
<b>:CISTART:</b>	Received data records are sent to the CI program again.
<b>:CISUSPEND:</b>	Data records are not sent to the CI program but written into a file named CISUSP.<Stream>. This file is sent to the CI program after reception of the command <b>:CIRESUME:</b> . Before sending this command all ONLINE terminals should be set to OFFLINE or AUTONOMOUS.
<b>:CIRESUME:</b>	If a file CISUSP.<Stream> exists, it is sent to the CI program and data records are sent to the CI program again as usual.
<b>:READTDF:</b>	The terminal configuration file TERMDEF.<Stream> is re-read by BCOMMIO. This is important, when an external program changed the file during an active communication.
<b>:STOPLINE:</b>	The same as the order STOPLINE, which means, that BCOMMIO is shut down.
<b>:STATUS:</b>	With this command the CI program can check whether BCOMMIO is still active. In this case, the command is acknowledged with <b>:ACK:</b> by BCOMMIO, otherwise of course no acknowledgment will be received. Additionally BCOMMIO sends the message BCOMMIO_STATUS.
<b>:ORDERSUSPEND:</b>	The execution of the order file is suspended. A currently active order will be finished.
<b>:ORDERRESUME:</b>	The execution of the order file is resumed.

---

**:ORDERDELETE:** Deletes the order file.

**:SET\_CITO\_xx:** Temporarily set the Call Interface timeout to xx seconds. The new value is only valid until the next start of BCOMMIO.

**:SET\_IO\_DBGLVL\_x:** Temporarily set a new value x for the debug level of BCOMMIO. This value is only valid until the next start of BCOMMIO.

**:SET\_DLL\_DBGLVL\_x:** Temporarily set a new value x for the debug level of the library BCWIN.DLL. This value again is only valid until the next start of BCOMMIO.

All the above commands are acknowledged with **:ACK:** by BCOMMIO (exception: the command **:ORDERDELETE:** is acknowledged with **:NACK:** in case the order file couldn't be deleted).

**After BCOMMIO received the command :STOPLINE: and has sent the answer :ACK:, it sends no more data records to the CI program.**

**:STATUS\_xy:** With this command the CI program can check the status of the terminal with the GID x and the DID y.

Possible answers to this command are:

**:TERM\_OKCO:** if the terminal is logged on and available

**:TERM\_ERRU:** if the terminal is logged on but currently not available

**:TERM\_ERCO:** if the terminal is not logged on

Additionally BCOMMIO sends the message BCOMMIO\_STATUS (see above).

**:STATUS\_ORDER:** With this command the CI program checks whether BCOMMIO is currently executing an order. If so, BCOMMIO answers with **:ACTIVE:**, otherwise with **:INACTIVE:**.

**:VERSION:** To this command BCOMMIO answers with its version number, e.g. **:VERSION\_3.3.0:**

**:VERSION\_IO:** same as **:VERSION:**

**:VERSION\_DLL:** To this command BCOMMIO answers with the version number of the BCWIN.DLL, e.g. **:VERSION\_3.3.0:**

---

Additionally there is the possibility to send whole orders to BCOMMIO via the Call Interface. For that you can use the following command:

`:ORDER_cmd_subcmd_lineno_gid_did_file_data_memo:`

where:

- cmd = command (e.g. WRPEAUTO)
- subcmd = subcommand (e.g. SYS1)
- lineno = line number of the terminal
- gid = GID of the terminal order \*
- did = DID of the terminal order \*
- file = entry in the field **File** of the order
- data = entry in the field **Data** of the order
- memo = entry of the field **External memo field** of the order

Entries, which should be empty, must be represented by <none>.

Examples:

```
:ORDER_WRPEAUTO_<none>_-1_@_@_BOOKING.DAT_MAST.REC_Ø:  
:ORDER_TIMEDAT_SYS1_Ø_*_*_BOOKING.DAT_<none>_Ø:  
:ORDER_MESSAGE_DØ_-1_@_A_<none>_Hello_Ø:
```

This command is also acknowledged with `:ACK:` by BCOMMIO.

**After BCOMMIO received the command `:ORDER_STOPLINE...` and has sent the answer `:ACK:`, it sends no more data records to the CI program.**

---

## 9.9. Additional Records from BCOMMIO to the CI Program

There are three additional records, which could be sent to the CI program by BCOMMIO:

*:SYSMESS\_xxx:*

This record is sent by BCOMMIO when the system message with the number xxx was written into the system message file SYSMESS.<Stream>. By default this feature is disabled, but can be enabled by setting the key **send\_sysmess\_to\_ci** in the section **[Others]** of the BCOMM32.INI file to Y.

*:ORDER\_xxxx\_START\_YYMMDDM\_HHMMSS\_cmd\_subcmd\_gid\_did\_file\_data\_status:*

This record is sent by BCOMMIO, when the order with the number xxxx of the order file ORDER.<Stream> is started.

*:ORDER\_xxxx\_END\_YYMMDDM\_HHMMSS\_cmd\_subcmd\_gid\_did\_file\_data\_status:*

This record is sent by BCOMMIO, when the order with the number xxxx of the order file ORDER.<Stream> has finished.

where:

- YYMMDD = start/end date of the order
- HHMMSS = start/end of the order
- cmd = command (e.g. WRPEAUTO)
- subcmd = subcommand (e.g. SYS1)
- gid = GID of the terminals (e.g. @)
- did = DID of the terminals (e.g. @)
- file = entry of the field **File** of the order
- data = entry of the field **Data** of the order
- status = execution status of the order (RUN when started, OKCO/ERRU/ERCO/OKME when finished)

When a field of the order file is empty, it is represented by <none> in the record.

These two data records are also disabled by default, but can be enabled by setting the switch **send\_orderlog\_to\_ci** in the section **[Others]** of the configuration file BCOMM32.INI to Y.

**After BCOMMIO received the command *:ORDER\_STOPLINE...* and has sent the answer *:ACK:*, it does NOT send the *:ORDER\_xxxx\_START...* and *:ORDER\_xxxx\_END...* data records to the CI program.**

The above three records have to be acknowledged by the CI program with *:ACK:*.

---

**:ENQ:**

This record can be send by BCOMMIO, when it gets no answer for a previously sent data record within the CI timeout. In case the CI program had not enough time to answer to the previous data record, but is still operational, it has three possibilities after receiving the **:ENQ:**

1<sup>st</sup>: the CI program sends no response to the **:ENQ:** within the CI timeout: BCOMMIO writes the previous data record into the CINoHost file and shuts down

2<sup>nd</sup>: the CI program sends **:ACK:** as the response to the **:ENQ:**, then BCOMMIO also writes the previous data record into the CINoHost file, but it does not shut down and the communication can go on as usual.

3<sup>rd</sup>: the CI program sends **:REQ:** as the response to the **:ENQ:**, then BCOMMIO sends the previous data record again. When it again gets no answer to that data record within the CI timeout, it writes the record to the CINoHost file and shuts down. When BCOMMIO receives the correct answer within the timeout, the communication can continue as usual.

To enable the sending of the **:ENQ:** record, you have to set the switch **send\_enq** in the section **[Others]** of the configuration file BCOMM32.INI to **Y**.

**:READ\_CISUSP\_START:**

This record is sent to the CI program after the BCOMMIO has received the **:CIRESUME:** command and begins to read the file CISUSP.<stream> and sends the data records from the file to the CI program.

**:READ\_CISUSP\_END:**

This record is sent to the CI program after the BCOMMIO has completely processed the file CISUSP.<stream>.

**Attention: between the above two records, no :CISUSPEND: and :CIRESUME: commands should be sent to the BCOMMIO by the CI program.**

## 9.10. Flow of Communication

For a CI program, there is the following sequence of commands during a communication via Shared Memory:

when starting, BCOMMIO writes the request **:STATUS:** into Shared Memory and then waits a specified timeout (key **cito** in the file BCOMM32.INI in the section **[Line<x>]**) for the answer **:OK:**. That means that the CI program has to read out the **:STATUS:** request from Shared Memory by use of the DLL function *InitCISharedMem* and then write **:OK:** into Shared Memory with the function *WriteCISharedMem*.

When BCOMMIO can read out **:OK:** from Shared Memory, the connection between CI program and BCOMMIO is established.

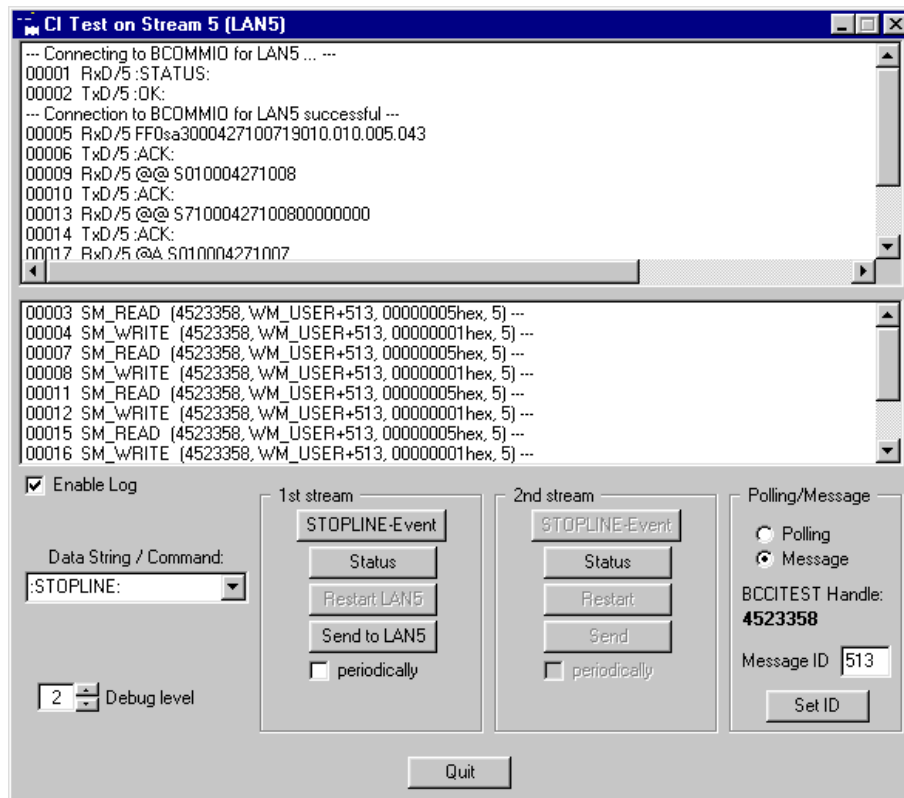
---

All data records, that BCOMMIO receives from the terminal(s), are sent to the CI program via the Shared Memory.

The receiving process has to read out these data records from the Shared Memory continuously.

For that it can either periodically call *ReadCISharedMem* (Polling) or wait for the Windows message *SM\_WRITE* (see above.) and then call *ReadCISharedMem* once (event driven).

The following screenshot shows the start of a communication between BCOMMIO and the demo program BCCITEST.EXE:



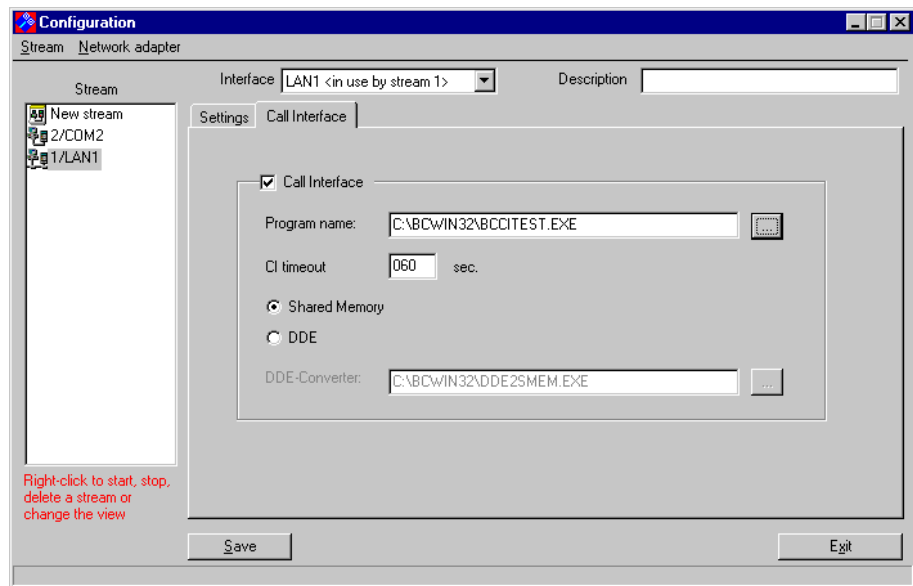
**Figure 9-2: Demo program BCCITEST**

The upper of the two windows shows the sented and received data records, the lower window shows the received messages.

Here are some explanations regarding the numbers at the beginning of each line:

- 00001: (upper) the demo program has received :STATUS: from BCOMMIO
- 00002: (upper) the demo program writes :ACK: an the answer
- 00003: (lower) BCOMMIO has read the record and sends the message SM\_READ
- 00004: (lower) BCOMMIO has written a record and send the message SM\_WRITE
- 00005: (upper) the demo program reads the record (FF0sa...)
- 00006: (upper) the demo program answeres with :ACK:
- 00007: (lower) BCOMMIO has read the answer and sends the message SM\_READ

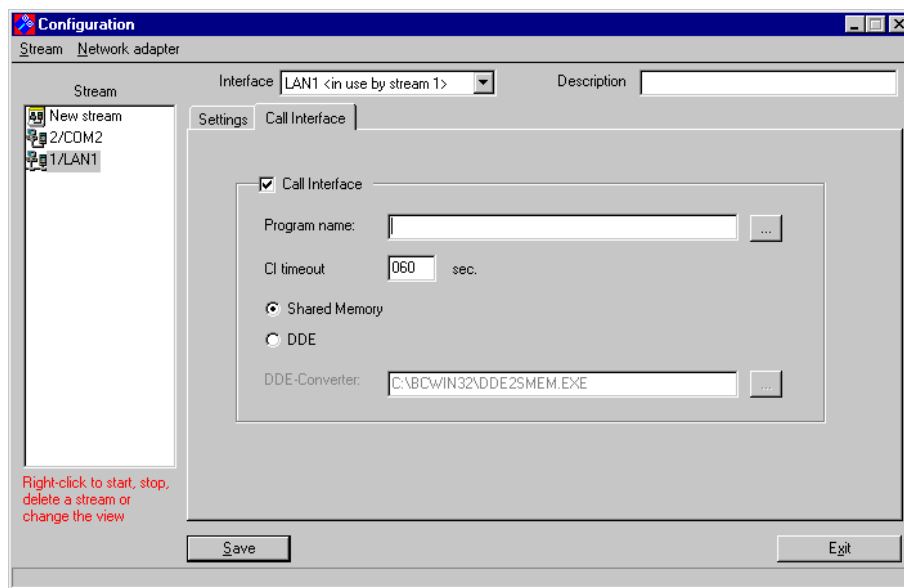
## 9.11. Configuration of the Call Interface



**Figure 9-3: Call Interface configuration 1**

In this example BCOMMIO starts the demo program C:\BCWIN32\BCCITEST.EXE automatically.

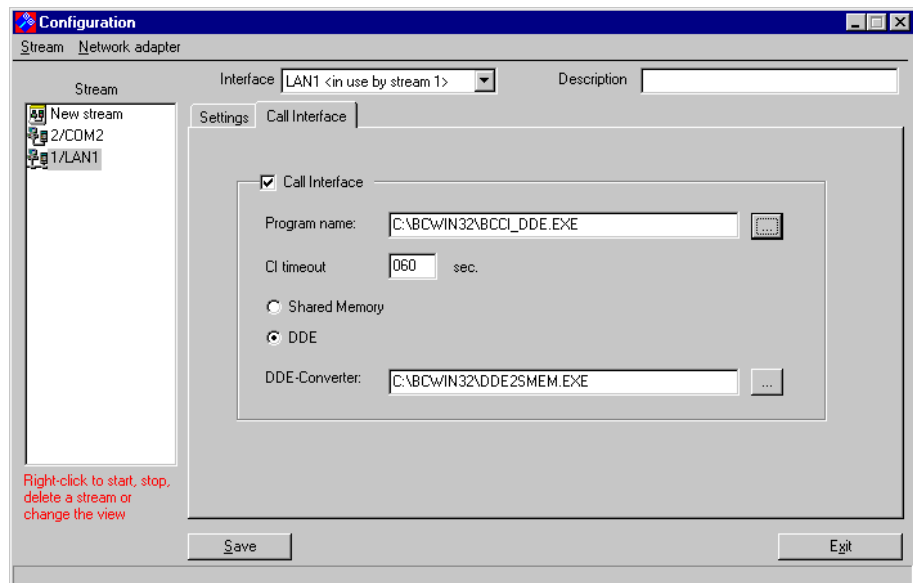




**Figure 9-4: Call Interface configuration 2**

In this example BCOMMIO must be started manually or by the CI program, because no program name for a CI program is configured.

Settings to use the example CI program BCCI\_DDE with DDE:



**Figure 9-5: Call Interface configuration 3**

BCOMMIO first starts the converter C:\BCWIN32\DDE2SMEM.EXE and then the CI demo program C:\BCWIN32\BCCI\_DDE.EXE

---

## 10. Table of Figures

Figure 2-1: Info mask.....	15
Figure 2-2: Main window of BCWIN .....	16
Figure 2-3: Context menu of the PC symbol .....	18
Figure 2-4: Context menu of a stream symbol .....	19
Figure 2-5: Context menu of a LAN adapter symbol .....	22
Figure 2-6: Context menu of a terminal symbol.....	23
Figure 2-7: Terminal info .....	23
Figure 2-8: Maintain a terminal.....	24
Figure 2-9: Context menu of a file symbol.....	24
Figure 2-10: Drag and Drop symbols 1.....	25
Figure 2-11: Drag and Drop symbols 2.....	26
Figure 2-12: The status bar .....	28
Figure 3-1: Menu File.....	29
Figure 3-2: Print system messages .....	30
Figure 3-3: Display system messages.....	31
Figure 3-4: Menu Functions.....	33
Figure 3-5: Transmit master record file .....	36
Figure 3-6: Select master record .....	37
Figure 3-7: Maintain master record .....	38
Figure 3-8: Automatic master record download.....	41
Figure 3-9: Master record upload .....	47
Figure 3-10: Request bookings .....	49
Figure 3-11: Control buffer memory .....	51
Figure 3-12: Set date/time .....	53
Figure 3-13: Transmit display record.....	55
Figure 3-14: Transmit transparent data .....	57
Figure 3-15: Summer/Wintertime setup .....	59
Figure 3-16: Example for Summer/Wintertime setup.....	60
Figure 3-17: Seconds/User language in data record.....	62
Figure 3-18: Menu Configuration .....	64
Figure 3-19: Client management .....	65
Figure 3-20: Example of a terminal configuration.....	67
Figure 3-21: Terminal logon .....	68
Figure 3-22: File selection .....	71
Figure 3-23: Logged-on terminal .....	71
Figure 3-24: Update terminal program number .....	73
Figure 3-25: Set operating mode.....	74
Figure 3-26: Add new terminal type.....	76
Figure 3-27: New stream .....	77
Figure 3-28: Add communication stream .....	77
Figure 3-29: Serial stream .....	78
Figure 3-30: Network stream .....	79
Figure 3-31: Add network adapter .....	80
Figure 3-32: Call Interface .....	81
Figure 3-33: Context menu of a stream.....	82
Figure 3-34: Context menu of a network adapter.....	83
Figure 3-35: Large/Small icons.....	83
Figure 3-36: Start options for LAN and COM streams.....	84
Figure 3-37: Start options for FTP stream.....	85
Figure 3-38: Directories .....	86
Figure 3-39: Printer log .....	88
Figure 3-40: Global file definitions .....	89
Figure 3-41: Specific file definitions .....	90

---

Figure 3-42: FTP Transfer configuration .....	92
Figure 3-43: Change password .....	93
Figure 3-44: Backup .....	94
Figure 3-45: Restore.....	96
Figure 3-46: Scheduler .....	98
Figure 3-47: Services .....	100
Figure 3-48: Change language .....	102
Figure 3-49: Set PC date/time .....	103
Figure 3-50: Menu System setup .....	104
Figure 3-51: AT modem .....	105
Figure 3-52: Create modem parameter file .....	107
Figure 3-53: Device interface 1 .....	108
Figure 3-54: Device interface 2 .....	109
Figure 3-55: Device interface 3 .....	110
Figure 3-56: Device settings.....	111
Figure 3-57: Software download.....	112
Figure 3-58: Parameter setting.....	114
Figure 3-59: Parameter upload.....	116
Figure 3-60: Parameter download .....	118
Figure 3-61: BETA setup by Telnet 1 .....	120
Figure 3-62: BETA setup by Telnet 2 .....	121
Figure 3-63: BETA setup by ARP and Telnet 1 .....	122
Figure 3-64: BETA setup by ARP and Telnet 2.....	123
Figure 3-65: BETA setup by ARP and Telnet 3.....	124
Figure 3-66: Menu Order .....	125
Figure 3-67: Display orders .....	126
Figure 3-68: BCOMMIO window.....	131
Figure 3-69: Menu Info .....	133
Figure 3-70: System information .....	134
Figure 3-71: Version informations .....	135
Figure 3-72: Licence registration .....	136
Figure 3-73: Select options.....	137
Figure 3-74: Menu Help.....	138
Figure 3-75: Tooltip help.....	139
Figure 4-1: Select debug file.....	141
Figure 4-2: Main window of Debug File Viewer .....	142
Figure 5-1: TermStat configuration.....	145
Figure 5-2: Main window of TermStat.....	146
Figure 5-3: Streams and Terminals in TermStat .....	147
Figure 5-4: Logged-on terminals in TermStat.....	148
Figure 6-1: File system of B-COMM WIN32.....	154
Figure 9-1: Demo program OCXDEMO .....	213
Figure 9-2: Demo program BCCITEST .....	239
Figure 9-3: Call Interface configuration 1 .....	240
Figure 9-4: Call Interface configuration 2 .....	241
Figure 9-5: Call Interface configuration 3 .....	242



---

## 11. Index

:
---

:ACK:	231
:BPASTART:	232
:BPASTOP:	232
:CIRESUME:	232
:CISTART:	232
:CISTOP:	232
:CISUSPEND:	232
:ENQ:	236
:OK:	236
:ORDER_...:	234
:ORDER_xxxx_END_...:	235
:ORDER_xxxx_START_...:	235
:ORDERRESUME:	232
:ORDERSUSPEND:	232
:READ_CISUSP_END:	236
:READ_CISUSP_START:	236
:READTDF:	232
:SET_CITO_xx:	233
:SET_DLL_DBGLVL_x:	233
:SET_IO_DBGLVL_x:	233
:STATUS:	232, 236
:STATUS_ORDER:	233
:STOPLINE:	232
:SYSMESS_xxx:	235
:VERSION:	233
:VERSION_DLL:	233
:VERSION_IO:	233

A
---

About B-COMM	133
Activate scheduler	97
Add new terminal type	75
Additional Records from BCOMMIO to the CI Program	235
ANALYSE	126, 172, 174
AT modem	104
AutoAcknowledge	213

AutoDisconnect.....	213
Autom. Master record download.....	40
Autostart.....	21, 81, 154, 163

## B

Backup configuration.....	93
BCOMM32.INI.....	163
BCOMMIO.....	154
BCOMMIO_DOWN.....	227
BCOMMIO_STATUS.....	228
BCOMMIO_UP.....	227
BCOMMIOSatus.....	214
BETA setup by ARP and Telnet.....	121
BETA setup by Telnet.....	119
BlinkInterval.....	214
BlinkOnError.....	213
Booking file BOOKING.DAT.....	200
BOOKING.DAT.....	200

## C

Call Interface.....	156, 203
Calling up BCOMMIO.....	154
Call-Interface.....	79
Change password.....	92
CI command :BPASTART:.....	232
CI command :BPASTOP:.....	232
CI command :CIRESUME:.....	232
CI command :CISTART:.....	232
CI command :CISTOP:.....	232
CI command :CISUSPEND:.....	232
CI command :ORDER_...:.....	234
CI command :ORDERRESUME:.....	232
CI command :ORDERSUSPEND:.....	232
CI command :READTDF:.....	232
CI command :SET_CITO_xx:.....	233
CI command :SET_DLL_DBGLVL_x:.....	233
CI command :SET_IO_DBGLVL_x:.....	233
CI command :STATUS:.....	232
CI command :STATUS_ORDER:.....	233
CI command :STOPLINE:.....	232
CI command :VERSION:.....	233
CI command :VERSION_DLL:.....	233
CI command :VERSION_IO:.....	233
CI Protocol.....	231
CI record : READ_CISUSP_END:.....	236
CI record : READ_CISUSP_START:.....	236
CI record :ACK:.....	231
CI record :ENQ:.....	236
CI record :OK:.....	236
CI record :ORDER_xxxx_END_...:.....	235
CI record :ORDER_xxxx_START_...:.....	235
CI record :STATUS:.....	236

CI record : <i>SYSMESS_</i> xxx:	235
CI_OCX	212
CIError	214
CINOHST.<Stream>	231
Commands from the CI Program to BCOMMIO	232
Communication between BCOMMIO and TERMSTAT	152
Concept	11
Configuration	64
Configuration file BCOMM32.INI	163
Configuration of the Call Interface	239
ConnectInterval	214
ConnectRetries	214
ConnectToBCOMMIO	218
Control buffer memory	51
Control CI_OCX	212
Create modem parameter file	106

## D

Data records from BCOMMIO to TERMSTAT	150
Data records from TERMSTAT to BCOMMIO	149
Data Transfer Kernel BCOMMIO	154
DDE	231
DDE2SMEM.EXE	231
Debug file(s)	32
Debug level	83
DebugLevel	215
Delete all orders	128
Delete system messages	31
Deutsch/English GB/English US	101
Device address	66
Device interface	107
DID	66
Directories	85
DisconnectFromBCOMMIO	218
Display orders	125
Display system messages	31
Dll function <i>ExitCISharedMem</i>	206
DLL function <i>GetBCOMMIOSStatus</i>	208
DLL function <i>ImmediateSTOPLINE</i>	209
DLL function <i>InitCISharedMem</i>	204
DLL function <i>ReadCISharedMem</i>	206
DLL function <i>SendImmediateData</i>	209
DLL function <i>SetCIHandle</i>	204
DLL function <i>SetImmDataHandle</i>	204, 210
DLL function <i>WriteCISharedMem</i>	208
DLL functions	204

## E

ERCO	127, 173
error codes	186
ERRU	127, 173
evBCOMMIODown	220



evBCOMMIOOrderEnd.....	221
evBCOMMIOOrderRestart.....	222
evBCOMMIOOrderStart.....	221
evBCOMMIORead.....	220
evBCOMMIOStatus.....	220
evBCOMMIUp.....	220
evDataRecordReceived.....	221
Event <i>evBCOMMIODown</i> .....	220
Event <i>evBCOMMIOOrderEnd</i> .....	221
Event <i>evBCOMMIOOrderRestart</i> .....	222
Event <i>evBCOMMIOOrderStart</i> .....	221
Event <i>evBCOMMIORead</i> .....	220
Event <i>evBCOMMIOStatus</i> .....	220
Event <i>evBCOMMIUp</i> .....	220
Event <i>evDataRecordReceived</i> .....	221
Event <i>evImmediateDataAck</i> .....	223
Event <i>evImmediateDataNack</i> .....	223
evImmediateDataAck.....	223
evImmediateDataNack.....	223
Execution status.....	173
Execution Status.....	127
Exit.....	32
ExitCISharedMem.....	206

## F

Father-son-procedure.....	43
File.....	29
File definitions.....	88
File system.....	153
Flow of Communication.....	236
FTP Transfer configuration.....	91
Functions.....	33, 204

## G

GetBCOMMIOSStatus.....	208
GID.....	66
Group address.....	66

## H

Help.....	137
HOLD.....	127, 173

## I

IMM_DATA_ACK.....	228
IMM_DATA_NACK.....	229
ImmediateSTOPLINE.....	209
Info.....	132
InitCISharedMem.....	204
Initialize communication stream.....	76

Initialize interface.....	76
Installation.....	14
Introduction.....	10

## L

Language.....	101
Languages .....	201
LEDShape .....	215
Licence registration.....	135
Line no. ....	160
Line No. ....	71
Logon a terminal .....	65
LOWPAR.....	172

## M

Manual .....	137
Master record upload .....	47
MESSAGE.....	126, 172, 174
Message <i>BCOMMIO_DOWN</i> .....	227
Message <i>BCOMMIO_STATUS</i> .....	228
Message <i>BCOMMIO_UP</i> .....	227
Message <i>IMM_DATA_ACK</i> .....	228
Message <i>IMM_DATA_NACK</i> .....	229
Message <i>ORDER_END</i> .....	230
Message <i>ORDER_REP_START</i> .....	230
Message <i>ORDER_START</i> .....	229
Message <i>SM_READ</i> .....	226
Message <i>SM_WRITE</i> .....	226
MessageID .....	215
Messages .....	225
Method <i>ConnectToBCOMMIO</i> .....	218
Method <i>DisconnectFromBCOMMIO</i> .....	218
Method <i>ReadBCOMMIOStatus</i> .....	218
Method <i>SendDataToBCOMMIO</i> .....	219
Method <i>SendImmediateDataToBCOMMIO</i> .....	219
Method <i>ShutdownBCOMMIO</i> .....	219
Modem .....	104, 106
MODEMPAR.....	126, 172, 175
MRUPLD .....	126, 172, 175

## N

Notes to time-controlled orders.....	98
--------------------------------------	----

## O

OKCO .....	127, 173
OKME.....	127, 173
OnlineBooking.....	215
Operation.....	29
Order .....	124

Order ANALYSE.....	174
Order file ORDER.<Stream>.....	169
Order MESSAGE.....	174
Order MODEMPAR .....	175
Order MRUPLD.....	175
Order READCFG.....	176
Order READTDF .....	176
Order REDAAUTO .....	177
Order REDAOFF .....	177
Order REPARA.....	178
Order SETMODE .....	178
Order STATUS .....	179
Order STOPLINE .....	179
Order SWMEM.....	180
Order TIMEDAT .....	181
Order UPDPRGNO.....	182
Order WRDATRA .....	182
Order WRPARA .....	183
Order WRPEAUTO .....	183
Order WRPEHAAU.....	184
ORDER.<Stream> .....	169
ORDER_END.....	230
ORDER_REP_START .....	230
ORDER_START .....	229
Other Languages .....	201

## P

Parameter download.....	117
Parameter File PARAM.DAT .....	199
Parameter setting .....	113
Parameter upload.....	115
PGMDNLD .....	172
Print system messages .....	30
Printer log.....	87
Program Download .....	111
Program flow in a CI program .....	210
Program Operation.....	29
Property <i>AutoAcknowledge</i> .....	213
Property <i>AutoDisconnect</i> .....	213
Property <i>BCOMMIOStatus</i> .....	214
Property <i>BlinkInterval</i> .....	214
Property <i>BlinkOnError</i> .....	213
Property <i>CIError</i> .....	214
Property <i>ConnectInterval</i> .....	214
Property <i>ConnectRetries</i> .....	214
Property <i>DebugLevel</i> .....	215
Property <i>LEDShape</i> .....	215
Property <i>MessageID</i> .....	215
Property <i>OnlineBooking</i> .....	215
Property <i>ShowAsLED</i> .....	216
Property <i>ShowMissingTerminal</i> .....	216
Property <i>ShowTrafficlight</i> .....	217
Property <i>StartBCOMMIO</i> .....	217

Property <i>Stream</i> .....	217
Property <i>UpdateInterval</i> .....	217
Property <i>WindowSize</i> .....	217
Protocol.....	231

## R

ReadBCOMMIOStatus .....	218
READCFG .....	126, 172, 176
ReadCISharedMem .....	206
READTDF .....	126, 172, 176
REDAAUTO.....	126, 172, 177
REDAOFF .....	126, 172, 177
REPARA.....	126, 172, 178
Request bookings .....	49
Restore configuration.....	95
RUN .....	127, 173

## S

SCHDUL.....	172
Scheduler.....	97
Seconds/User language in data record.....	62
SendDataToBCOMMIO .....	219
SendImmediateData .....	209
SendImmediateDataToBCOMMIO .....	219
Service.....	11, 16, 17, 21, 81, 163
Services .....	99
Set date/time.....	26, 53, 163
Set operating mode.....	73
Set PC date/time.....	102
Set time/date.....	26, 53, 163
SetCIHandle .....	204
SetImmDataHandle .....	204, 210
SETMODE.....	126, 172, 178
Setting a network interface.....	78
Setting a serial interface .....	78
Setup .....	14
Shared Memory .....	203
ShowAsLED .....	216
SHOWDBG .....	140
ShowMissingTerminal .....	216
ShowTrafficlight .....	217
ShutdownBCOMMIO .....	219
SM_READ .....	226
SM_WRITE.....	226
Software download.....	111
Start communication .....	129
Start options .....	83
StartBCOMMIO.....	217
STATUS .....	126, 172, 179
Stop communication.....	131
STOPLINE.....	126, 172, 179
Stream .....	217

Summer/Wintertime setup.....	59
SWMEM.....	126, 172, 180
SYSMESS.<Stream>.....	185
System information .....	133
System Message File SYSMESS.<Stream>.....	185
System requirements .....	14
System setup.....	103

## T

TERMDEF.<Stream> .....	160
Terminal configuration file TERMDEF.<Stream> .....	160
Terminal logon.....	65
Terminal status .....	135
TERMMODE.....	172
TERMSTAT .....	144
The Active-X Control CI_OCX .....	212
The application SHOWDBG.....	140
The application TERMSTAT.....	144
The B-COMM WIN32 file system.....	153
Time-controlled orders.....	98
TIMEDAT .....	126, 172, 181
Tooltip help.....	137
Translation into other Languages .....	201
Transmit display record.....	55
Transmit master record file .....	34
Transmit transparent data.....	57

## U

Update terminal program number .....	72
UpdateInterval.....	217
UPDPRGNO .....	182
Using the Program.....	15

## V

Version informations.....	134
---------------------------	-----

## W

WindowSize.....	217
Winter/Summertime setup.....	59
WRDATRA .....	126, 172, 182
WriteCISharedMem .....	208
WRPARA.....	126, 172, 183
WRPEAUTO .....	126, 172, 183
WRPEHAAU .....	126, 172, 184



Kaba Benzing GmbH  
Postfach 3980  
78028 Villingen-Schwenningen  
Telefon (0 77 20) 603-0  
Telefax (0 77 20) 60 31 02

