



**ROHDE & SCHWARZ**

Test and Measurement  
Division

# Supplement

## Controller Function for ZVR

Printed in the Federal  
Republic of Germany

Microsoft, Windows NT, Windows, Excel, Visual C++ and Visual Basic are trademarks or registered trademarks of Microsoft Corporation.

# Contents

<b>1</b>	<b>Controller Function .....</b>	<b>5</b>
1.1	Connecting a Mouse .....	6
1.2	Connecting an External Keyboard .....	7
1.3	Connecting an External Monitor .....	8
1.4	Connecting an Output Device .....	10
1.5	Connecting a CD-ROM Drive .....	18
1.6	Firmware Update .....	20
1.7	Installing Windows NT Software .....	21
1.8	Options .....	22
1.8.1	Option FSE-B17 – Second IEC/IEEE Interface .....	22
1.8.2	Option FSE B16 – Ethernet Adapter .....	25
<b>2</b>	<b>Modifications of the <i>GENERAL SETUP</i> Menu .....</b>	<b>38</b>
<b>3</b>	<b>Modifications of the <i>COPY</i> Menu .....</b>	<b>39</b>
3.1	Printing Data – <i>COPY</i> Key .....	39
3.2	Configuration of the Output Device – <i>SETTINGS</i> Key .....	41
<b>4</b>	<b>Modifications of the IEC/IEEE-Bus Commands .....</b>	<b>43</b>
4.1	HCOPY-System .....	43
4.2	SYSTEM-System .....	44
<b>5</b>	<b>RSIB Interface .....</b>	<b>45</b>
5.1	Remote Control via RSIB Interface .....	45
5.2	RSIB Interface Functions .....	46
5.3	Programming via the RSIB Interface .....	53
5.3.1	Visual Basic .....	53
5.3.2	C / C++ .....	56
5.3.3	WinWord (Word Basic) .....	58
5.3.4	Excel .....	60



# Index

## A

Abort	
Print .....	39
<b>Administrator identification</b> .....	5
AUI connector .....	27

## B

<b>BNC connector</b> .....	25
----------------------------	----

## E

Ethernet Adapter .....	25
------------------------	----

## F

FTP	
operation .....	37

## H

Hardcopy	
abort .....	39
copy .....	39
format .....	42
output device .....	41

## K

Key	
COPY (hardcopy) .....	39
Keyboard, external	
connection .....	7

## L

<b>Login</b> .....	5
<b>Login (NT controller)</b> .....	5
<b>Logout</b> .....	5

## M

MICROSOFT NET	
operation .....	31
Monitor	
connection .....	8
Mouse	
connection .....	6

## N

network operating system	
FTP .....	37

MICROSOFT NET .....	31
NOVELL NETWARE .....	31
TCP/IP .....	36
NOVELL NETWARE	
operation .....	31
NT controller .....	5

## O

Option	
ZVR-B16 - Ethernet Adapter .....	25

## P

Password	
Windows NT .....	5
Plotter	
connection .....	10
Print	
abort .....	39
Printer	
connection .....	10

## R

Remote Control	
RS-232-C .....	45
<b>RJ45 (star topology)</b> .....	27

## S

Setup	
general .....	38
Soft key	
ENABLE DEV1 / DEV2 .....	42
GENERAL SETUP .....	38
HARDCOPY DEVICE .....	41
MONITOR CONNECTED .....	38
SETTINGS DEVICE 1/2 .....	41

## T

TCP/IP	
operation .....	36
Thick Ethernet .....	27
<b>Thin Ethernet</b> .....	25

## W

Windows NT .....	5
administrator .....	5
login .....	5
password .....	5



# 1 Controller Function

**Caution:**

- *The drivers used in the integrated controller function are adapted to the measuring instrument. Only the settings described below should be resorted to. The existing driver software should be modified only with the update software released by Rohde&Schwarz.*
- *Do not power down during booting. Such a switch-off may lead to corruption of the hard disk files.*

The instrument possesses an integrated Windows-NT controller. The user can switch between the measurement screen and the controller screen. When an external monitor is connected, the measurement function and the controller function can be displayed simultaneously (see section "Connecting an External Monitor"). The controller function is automatically booted on instrument power-up.

Windows NT operation is described in the supplied manual or in the online help of Windows NT.

**Login**

Windows NT requires a login window in which the user is asked to identify himself by entering his name and password. An autologin comes as standard in the instrument, ie the login is performed automatically in the background. The user name used for this is "instrument" and the password is also "instrument" (in small letters).

To login under another name, call up the logout window in the task bar with START - SHUT DOWN. Mark the item "Close all programs and log on as a different user?" in the window and click YES while holding the SHIFT key down until the login window is displayed to enter the user identification. The password should be entered in the correct syntax including low-case and upper-case characters.

**Administrator identification**

Some of the installations (eg CD-ROM drive) described below are possible only under the administrator login. This is indicated in the relevant window.

The administrator is an identification prescribed by Windows NT which in particular allows system settings which are disabled for the standard user.

The password for the administrator is 894129 for the instrument.

After an installation under the administrator identification, the "Service Pack X" of Windows NT is to be re-installed, see section "Installation of Windows NT Software".

On the power-up following an installation under the administrator identification, the NT login window is displayed (no autologin). The user name "Administrator" is indicated in the window. This entry is to be replaced by "instrument" and then "instrument" as password entered. An autologin is then possible.

**Changeover between measurement screen and controller screen**

To call up the controller screen use key combination <ALT><SYSREQ> (US keyboard).

To return to the measurement screen, activate the window "Rohde & Schwarz ZVR NT Interface" on the controller.

**Logout**

The instrument can be switched off (standby mode). Logging out of Windows NT is not necessary.

### 1.1 Connecting a Mouse



**Caution:**

*The mouse may only be connected when the instrument is switched off (STANDBY). Otherwise, correct operation of the mouse and instrument cannot be guaranteed.*

The instrument has the facility to simplify operation by connecting a mouse to the rear-panel PS/2-mouse connector (MOUSE) for .

MOUSE



During measurement operation, softkeys, tables and data input fields may also be controlled via the mouse. For computer function, the mouse has the usual functions.

The control functions for the mouse during measurement operation are described in Chapter 2 in the section "Mouse Operations". Appendix E contains a list in which the screen display elements for mouse control of the corresponding soft keys and push buttons of the instrument are described. Appendix A contains the interface description.

After connection the mouse is automatically recognized. Special settings such as mouse cursor speed etc., can be performed in the Windows NT menu START - SETTINGS - CONTROL PANEL - MOUSE.



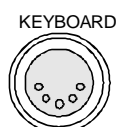
### 1.2 Connecting an External Keyboard



**Caution:**

*The keyboard may only be connected when the instrument is switched off (STANDBY). Otherwise, correct operation of the keyboard cannot be guaranteed.*

The instrument is fitted with a rear-panel 5-contact DIN connector (KEYBOARD) for the connection of an external PC keyboard.



During measurement operations, the keyboard simplifies the input of commentary text, filenames, etc. For computer function, the keyboard has the usual functions.

Appendix E contains a list which describes the assignment of the instrument front-panel key functions to the key codes of the external keyboard as well as special key combinations used for quick operations. Appendix A contains the interface description.

After connection the keyboard is automatically recognized. The default setting is for the US keyboard. Special settings such as repetitional rate etc. can be performed in Windows NT menu START - SETTINGS - CONTROL PANEL - KEYBOARD.

### 1.3 Connecting an External Monitor



**Caution:**

The monitor may only be connected when the instrument is switched off (STANDBY). Otherwise, the monitor may be damaged.

Do not modify the screen driver (display type) since this would disturb instrument operation.

**Notes:**

- When connecting the monitor to the PC MONITOR connector, the display of controller function can be adapted to the external screen (eg higher resolution) in the NT menu START-SETTING - CONTROL PANEL - DISPLAY PROPERTIES.
- CHIPS (setting = both) should not be modified since otherwise switching between the external monitor and the instrument screen is not possible.

The instrument is fitted with a rear-panel connector PC MONITOR or ANALYZER MONITOR for the connection of an external monitor.



With an external monitor it is possible to have an enlarged display of the measurement screen (ANALYZER MONITOR connector) or of the controller screen (PC MONITOR connector). The measuring instrument and the Windows NT controller can be operated in parallel. The mouse and the keyboard are allocated to only one mode.

#### Display of the measurement screen - Connection to ANALYZER MONITOR connector

**Connection**

After connecting the external monitor the measurement screen is displayed on both the external monitor and the instrument. Further settings are not necessary.

**Operation**

The instrument is operated as usual via its softkeys, the mouse and keyboard, etc..

**Toggle between measurement screen and controller screen**

Key combination <ALT><SYSREQ> is used to call up the controller. After call up, the mouse and keyboard are allocated to the controller function.

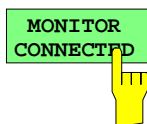
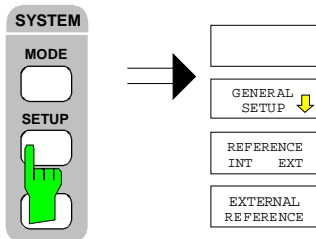
By activating the window " Rohde & Schwarz ZVR NT Interface " the user returns to the measurement screen and the mouse and keyboard are allocated to this screen.

### Display of the controller screen - Connection to PC MONITOR connector

#### Connection

After connecting the monitor external monitor operation should be selected.

Setting is performed in the *SYSTEM SETUP-GENERAL SETUP* menu (key group: *SYSTEM*, see in Chapter 2, the section "Presettings and Interface Configuration"):



---

#### Call *SETUP-GENERAL SETUP* menu

- Press the *SETUP* key in the *SYSTEM* field.

The *SETUP* menu is opened.

- 
- Press the *GENERAL SETUP* softkey.

The *GENERAL SETUP* submenu is opened and the current settings of the instrument parameters are displayed on the screen in the form of tables.

---

#### Selecting the external monitor mode

- Press the *MONITOR CONNECTED* softkey.

The softkey is in color to indicate that the external monitor mode is activated. The external monitor displays the controller screen.

#### Operation

The controller function is operated as usual with the mouse and keyboard. The measuring instrument (displayed on the instrument screen) can be operated with both the softkeys and keys of the instrument.

#### Toggle between screens

By activating (clicking) the window "R&S Analyzer Interface" at the controller the mouse and keyboard are allocated to the measurement screen. The mouse and keyboard are allocated to the controller when the window is deactivated.

### 1.4 Connecting an Output Device



**Caution:**

The output device may be connected only when the instrument is switched off (STANDBY)

- Note:**
- The installation of some printer drivers is possible only under the administrator identification (see section "Controller Function").
  - After the installation, the "Service Pack X" of Windows NT is to be re-installed, see "New Installation of Windows NT Software".
  - To ensure that the instrument performs an autologin, the user identification is to be reset to "instrument" after the next power-up, see section "Controller Function".

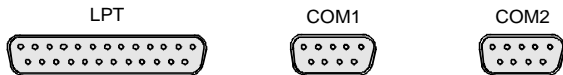
The instrument provides an option for connecting up to three different output devices to allow printing a hard copy of the display screen.

The interfaces can also be used for printing in the controller mode. The output formats "WMF" (Windows Metafile) and "Clipboard" are preset. A large number of output devices can be connected under Windows NT after installing suitable printer drivers.

The output devices supported by the instrument can be found in the selection box *DEVICE/LANGUAGE* in the *COPY -SETTINGS DEVICE1/2* menu (see in Chapter 2, the section "Documentation of Measurement Results"). To print out via the COM interface, the latter must be assigned the controller function in the *SETUP - GENERAL SETUP* menu (owner = OS).

Appendix A contains the interface description of the connectors.

The interface connectors are located on the rear panel:



After connection of the output device to the appropriate interface connector, the configuration of the interface as well as the output device must be entered and the output device must be assigned to the interface.

#### 1. Configuration of the interface

LPT1 Interface LPT1 cannot be configured.

**Note:** An external CM-ROM drive can be connected to this interface. If this interface is allocated, one of the serial interfaces can be used for the printout.

COM1/COM2 The configuration of serial interfaces COM1 and COM2 can be performed in the Windows NT menu *START - SETTINGS - CONTROL PANEL - PORTS*. The parameters *COM PORTS BAUDRATE, BITS, PARITY, STOPBITS, HW/SW-HANDSHAKE* determine the transmission parameters of the interface. They must agree with the specifications of the output device (see the operating manual for the output device). To print out via the COM interface, the latter must first be assigned the controller function in the *SETUP - GENERAL SETUP* menu (owner = OS).[bd1]

**Note:** Allocation of the instrument (Owner = Instrument) in the menu *SETUP - GENERAL SETUP* makes the interfaces available for remote control. The settings made in the menu *SETUP - GENERAL SETUP* overwrite the settings in the NT menu.

However, settings in the Windows NT menu do not overwrite those of the *SETUP* menu. This means that the settings are only valid as long as the interfaces are allocated to the controller.

**2. Selection and installation of the printer driver**

The selection and installation of the printer driver, the assignment to the interface and the setting of most of the printer-specific parameters (eg paper size) is performed under Windows NT in the START - SETTINGS - PRINTER menu.

## Connecting an Output Device

---

### 3. Configuration of the connected output device

The configuration of the connected output device and the assignment to the interface takes place in the *HARDCOPY DEVICE-SETTINGS DEVICE1/2* menu (key group: *COPY*, see in Chapter 2, the section "Selection and Configuration of the Output Device "). The configuration of up to two output devices (*DEVICE1* and *DEVICE2*) may be entered, of which, at least one must be activated for printing.

- The parameter *DEVICE/LANGUAGE* determines which output device is to be used.
- The parameter *PRINT TO FILE* determines if the output is in the form of a file.
- The parameter *ORIENTATION* sets the page format to horizontal or vertical (portrait).

Selecting the type of printer automatically sets the parameters *PRINT TO FILE* and *ORIENTATION* to values which correspond to a standard mode with this output device. Other printer-dependent parameters such as *FORMFEED*, *PAPERFEED* etc., can be modified under Windows NT in the printer properties window (*START/SETTINGS/PRINTER/SETTINGS/....*).

Table 1.4-1 shows the standard factory settings for the two output devices.

The factory settings of *DEVICE 1* correspond to output format "WMF" (Windows Metafile), printing is performed in a file. WMF is a common format which is used for the import of hardcopies (eg measurement windows) to other Windows applications that support this format (eg WinWord).

The factory setting of *DEVICE 2* is "Clipboard". In this setting the printout is copied to the Windows NT clipboard. Most of Windows applications support the clipboard. The clipboard content can be directly inserted in a document via *EDIT - PASTE*.

Table 1.4-1 Factory settings for *DEVICE 1* and *DEVICE 2* in the *HARDCOPY-DEVICE SETTINGS* menu.

Parameter	Parameter Name	<i>DEVICE 1 Settings</i>	<i>DEVICE 2 Settings</i>
Output Device	DEVICE/LANGUAGE	WINDOWS METAFILE	CLIPBOARD
Output	PRINT TO FILE	YES	---
Paper Format	ORIENTATION	---	---

## Connecting an Output Device

In the following example, a HP DeskJet 660C printer is connected to interface LPT1 and configured as *DEVICE2* of the instrument to output hardcopies of the measurement screen.

---

**Switch off device.**

---

**Connect printer to interface LPT1.**

---

**Switch on device.**

---

### Select printer driver under Windows NT

- Press key combination <ALT> <SYSREQ>  
The Windows NT screen is displayed.

- In the Start menu press first "Setting" and then "Printers".

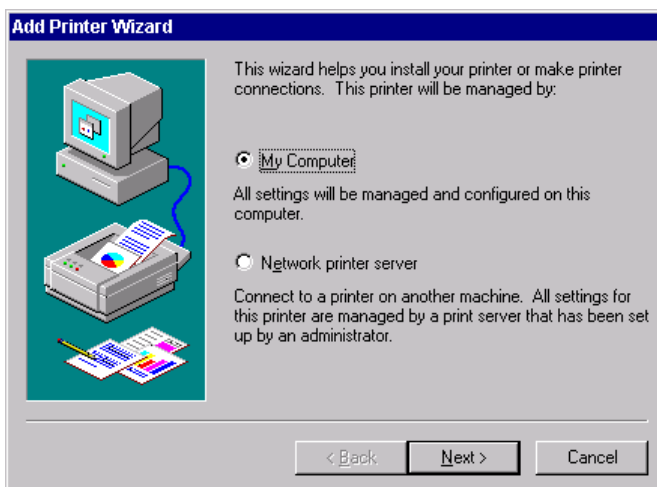
The printer window is opened.

- Double-click symbol "Add Printer".

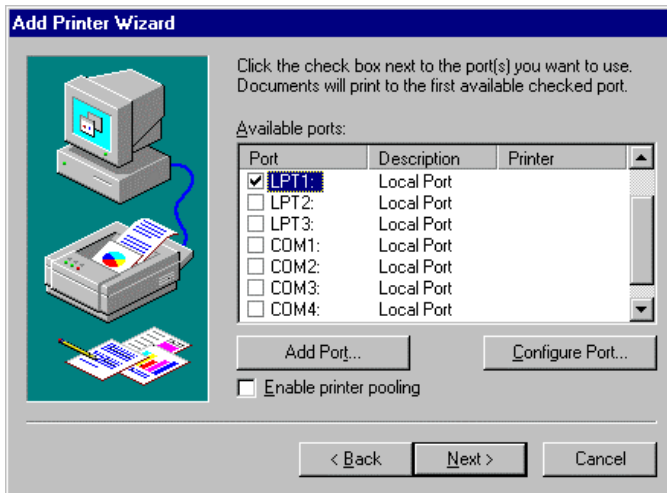
The "Add Printer Wizard" window is opened. This window leads through the following printer driver installation.

- Click first "My computer" and then "Next".

The available ports are displayed.

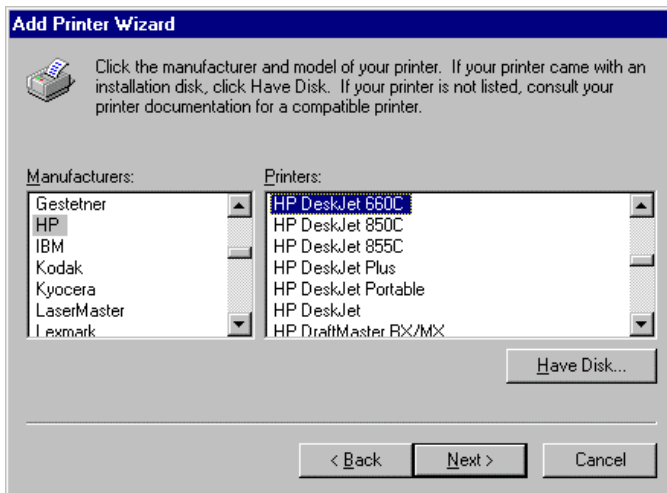


## Connecting an Output Device



- Select LPT1 port.  
The selection is marked by a tick.

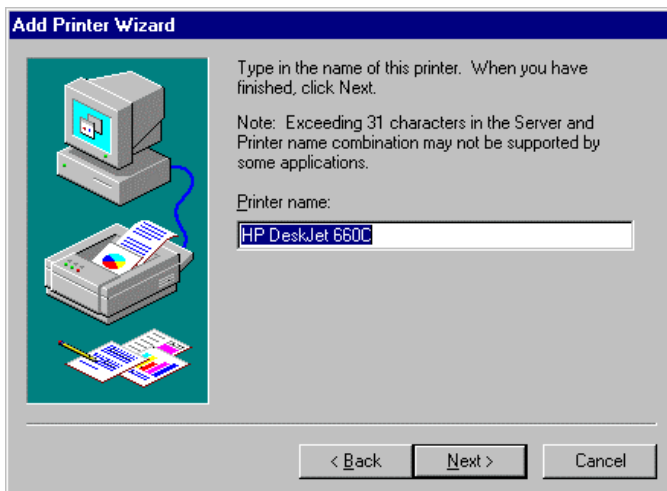
- Click "Next".  
The available printer drivers are displayed. The left-hand selection table indicates the manufacturers and the right-hand one the available printer drivers.



- Mark "HP" in selection table "Manufacturers" and "HP DeskJet 660C" in selection table "Printers".

**Note:** *If the desired type of output device is not shown in this list, it means that it is not supported by Windows NT.*

- Click "Next".  
The entry field for the printer name is displayed.

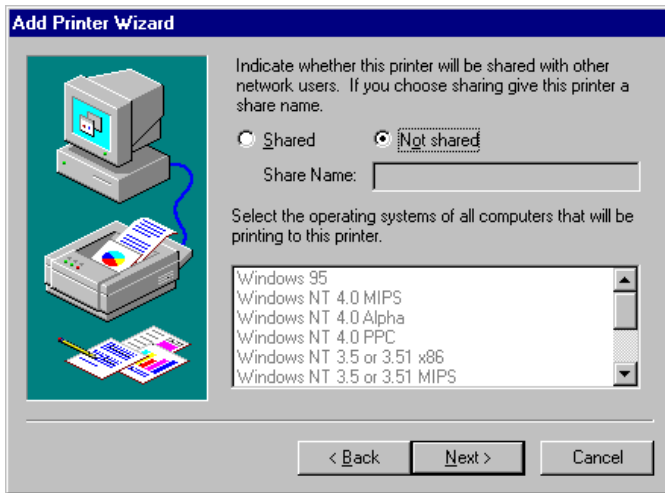


- The printer name can be modified in the entry field "Printer name" (max. 60 characters).

If one or more printers are already installed, a query is displayed in this window to ask if the printer last installed as default printer should be selected for the Windows NT applications (Do you want your Windows-based programs to use this printer as default printer?) "No" is preset.



## Connecting an Output Device



- Click "Next".

A query is displayed for providing the printer in the network. This query is irrelevant when installing a local printer. The answer "Not shared" is preset.

- Click "Next".

The window for starting a test page is displayed. The test page is for checking if the installation was successful.



- Switch on printer.
- Click Yes (recommended)".
- Click "Finish".

A test page is printed out if the installation was successful.

If the test page is not printed out or not completely, the Windows NT online help offers troubleshooting instructions under the entry "Printer Trouble Shooting".

### **Note:**

*If after clicking "Finish" the user is asked to specify the path for the printer driver, this printer installation should be performed under the administrator identification (see section "Controller Function").*

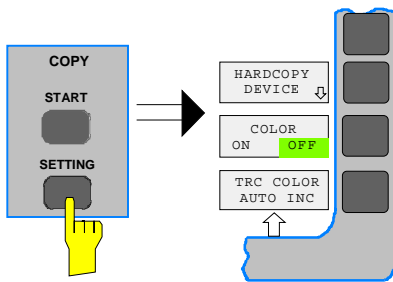
The instrument has to be configured with this printer for the printout of the measurement screen.

### **Configuring HP DeskJet 660C.**

- Click button "Rohde & Schwarz ZVR NT Interface".

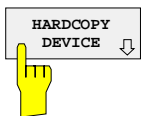
The measurement screen is displayed.

## Connecting an Output Device



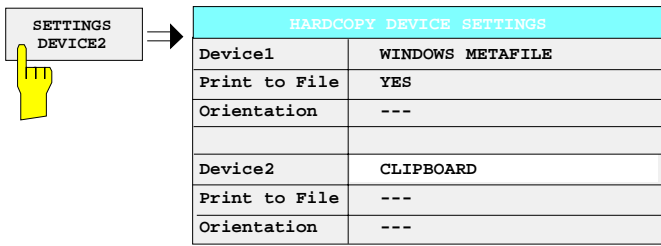
- Press the *SETTINGS* key in the *COPY* field.

The *SETTING* menu is opened.



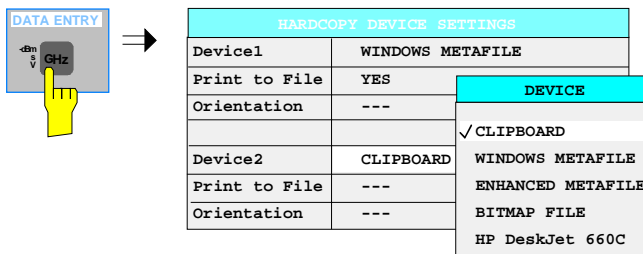
- Press the *HARDCOPY DEVICE* softkey.

The *HARDCOPY DEVICE* submenu is opened and the current settings of the two possible output devices are displayed on the screen in the form of tables.



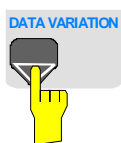
- Press *SETTING DEVICE2*.


Line *DEVICE2* is marked with the selection bar.



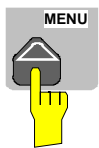
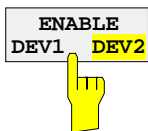
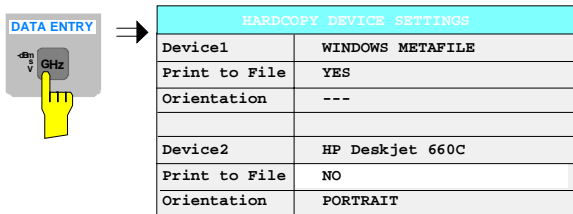
- Press one of the unit keys.

The selection box *DEVICE* is displayed on the screen. The current selection is marked by a tick and highlighted by the selection bar.



- Press cursor key  until the entry *HP DeskJet 600C* is highlighted by the selection bar.

## Connecting an Output Device



- Press one of the unit keys.

The selection box *DEVICE* is closed and *HP DeskJet 660C* is entered in line *DEVICE2*.

### Note:

Selecting the type of printer automatically sets the parameters *PRINT TO FILE* and *ORIENTATION* to values which correspond to a standard mode with this output device. Other printer-dependent parameters such as *PAPERSIZE*, can be modified under Windows NT in the printer properties window (*START/SETTINGS/PRINTER/SETTINGS*).

### Switch on printer.

- Press *ENABLE* softkey until *DEV2* is marked on the second softkey line.

Printing can then be started with the *START* key in the *HARDCOPY* menu.

### Return to main menu

- Press the menu key several times.

**Note:** After the installation, the "Service Pack X" of Windows NT is to be re-installed, see "New Installation of Windows NT Software".

### 1.5 Connecting a CD-ROM Drive



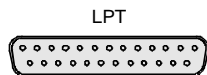
**Caution:**

The CD-ROM may only be connected when the instrument is switched off (STANDBY). If this is not observed correct operation of the CD-ROM and the instrument cannot be guaranteed.

**Notes:** The installation of a CD-ROM is possible only under the administrator identification (see section "Controller Function").

- After the installation, the "Service Pack X" of Windows NT is to be re-installed, see "New Installation of Windows NT Software".
- To ensure that the instrument performs an autologin, the user identification is to be reset to "instrument" after the next power-up, see section "Controller Function",

The instrument is fitted with a rear-panel interface LPT1 for the connection of a CD-ROM drive.



The following CD-ROM drives are supported:

- MICROSOLUTIONS BACKPACK External CD-ROM.
- FREECOM IQ DRIVE
- ADAPTEC Parallel SCSI Adapter + SCSI CD-ROM

After connection, the CD-ROM drive is to be installed under Windows NT.

---

**Switch off device.**

---

Connect CD-ROM drive to interface LPT1 of the instrument and to AC power source.

---

**Switch on device.**

---

**Administrator identification**

- Press key combination <ALT> <SYSREQ>  
The Windows NT screen is displayed.

- 
- Call the logout window with "Shut Down" in the "Start" menu.

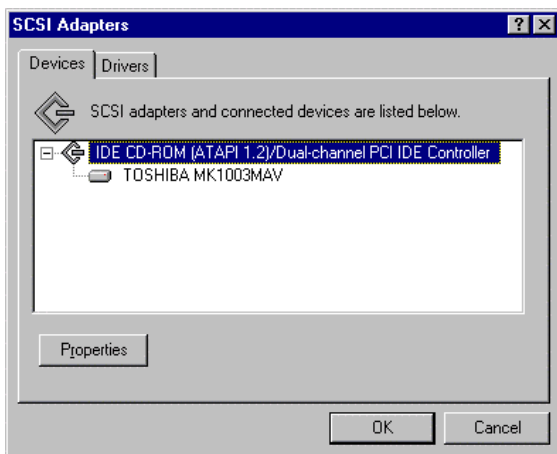
## Connecting a CD ROM Drive

- Mark entry "Shut down and log on as a different user".

- Press Shift key and click button "Yes" at the same time.

The login window is displayed.

- Enter "administrator" under "name" and "894129" under "password", confirm entry with "OK".



### Select driver under Windows NT

- In the Start menu press first "Setting" and then "Control Panel".

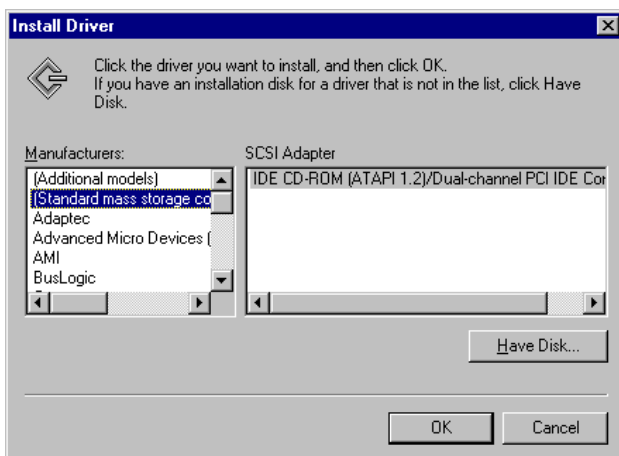
The system control window is opened.

- Double-click symbol "SCSI Adapters".

The "SCSI Adapters" window is opened.

- Click "Driver" index card and then button "Add".

The list of installed drivers is displayed.



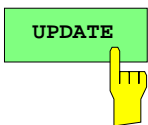
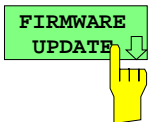
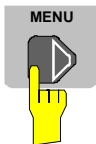
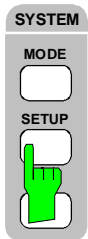
- Click "Have Disk".

This window leads through the following installation.

**Note:** After the installation, the "Service Pack X" of Windows NT is to be re-installed, see "New Installation of Windows NT Software".

### 1.6 Firmware Update

The installation of a new firmware version can be performed using the built-in diskette drive and does not require opening the spectrum analyzer. The firmware update kit contains several diskettes. The installation program is called up in the *SYSTEM - SETUP* menu.



The installation can be cancelled.



---

***Insert diskette 1 into the drive.***

---

***Call SETUP-GENERAL SETUP menu***

- Press the *SETUP* key in the *SYSTEM* field.  
The *SETUP* menu is opened.
- 

- Change to the right-hand menu using the menu key.
- 

- Press the *FIRMWARE UPDATE* softkey.  
The submenu is opened.
- 

- Press the *UPDATE* softkey.  
The installation program starts and leads the user through the remaining steps of the update.
- 

- Press the *RESTORE* softkey.  
The previous firmware version is restored.

## **1.7 Installing Windows NT Software**

The driver software and the system settings of Windows NT are exactly adapted to the measurement functions of the instrument. Correct operation of the instrument can therefore be guaranteed only if the software and hardware used is released or offered by Rohde & Schwarz.

The use of other software or hardware may cause malfunctioning or failures in the functions of the instrument.

A current list of released software can be obtained from your nearest Rohde&Schwarz agency (see list of addresses).

After each software installation requiring the administrator identification, it is necessary to re-install the "Service Pack X" of Windows NT (also with administrator identification; see section "Controller Function"):

---

### ***Re-installing Service Pack X***

- In the Start menu press first "Setting" and then "Run".

The entry window is opened.

- 
- Enter "C:\SPX\I386\UPDATE" into the command line and start installation with "OK".

The following window leads through the installation.

## Options

---

### 1.8 Options

#### 1.8.1 Option ZVR-B17 – Second IEC/IEEE Interface

- Notes:**
- The installation of option ZVR-B17 is possible only under the administrator identification (see section "Controller Function").
  - After the installation, the "Service Pack X" of Windows NT is to be re-installed, see "New Installation of Windows NT Software".
  - To ensure that the instrument performs an autologin, the user identification is to be reset to "instrument" after the next power-up, see section "Controller Function".

Besides the instrument external devices can also be controlled via the IEC/IEEE bus using the optional 3rd bus interface FSE-B17 and the computer function for instrument. The interface software permits IEC/IEEE-bus commands to be included in user programs. The installation instructions are enclosed with the option.

##### 1.8.1.1 Installing the software

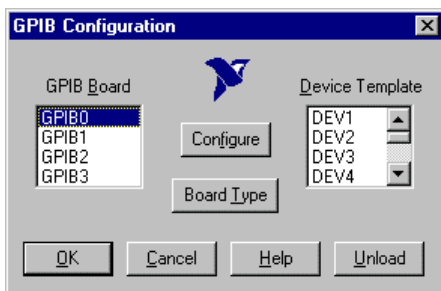
The operating software is already installed and need not be loaded from the driver diskettes which serve as backup diskettes.

The driver must be loaded on the start-up of Windows NT. To do this, enter the type of board, configure the board and enter the parameters for the connected equipment. If the option is factory fitted, all this has been done in the factory.

The following parameters may not be changed after selection on configuration of the board:

```
Board Type .....AT-GPIB/TNT
Base I/O Address .....02C0h
Interrupt Level .....3
DMA Channel .....5
Enable Auto Serial Polling ....No
```

For further parameters, refer to manual for the board.



---

#### Selecting the board type

- Click "Start" in the task bar.
- Click consecutively "Settings", "Control Panel" and "GPIB" in this sequence.

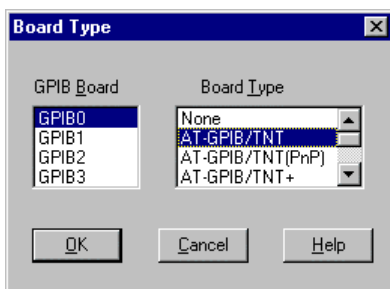
The "GPIB Configuration" menu for selecting the board type and configuring the board is opened.

- Click button "Board Type".

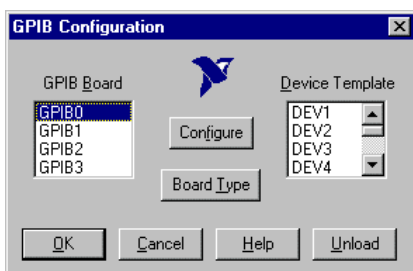
The "Board Type" menu for selecting the board type is opened.

---

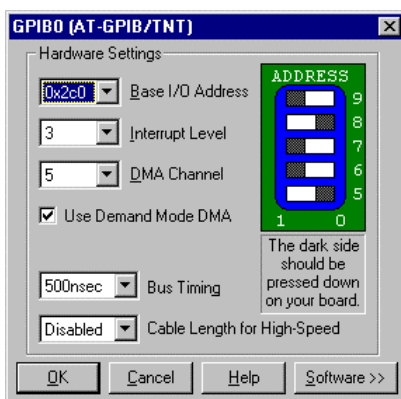




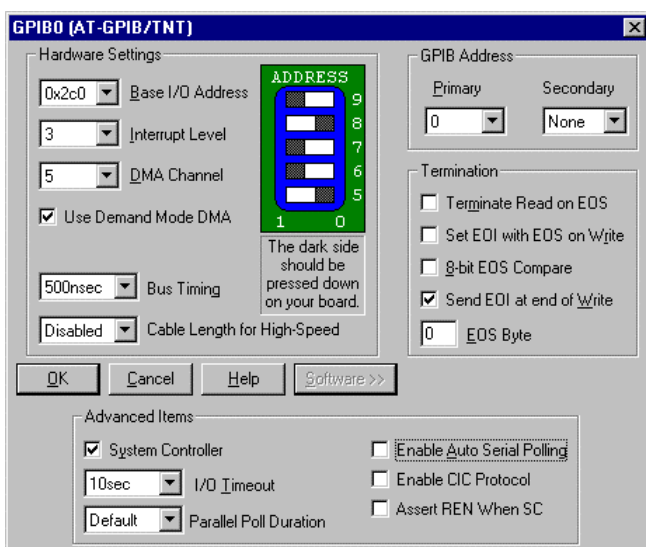
- Mark "GPIB0" in the "GPIB Board" list.
  - Mark "AT-GPIB/TNT" in the "Board Type" list.
  - Confirm the selection with "OK".
- The "GPIB Configuration" menu is displayed again.



- Click button "Configure".
- The "GPIB0 (AT-GPIB/TNT)" menu for configuring the board is opened.



- Set "3" in the "Interrupt Level" list.
  - Click button "Software".
- The menu is extended.



- Deactivate (= no tick) "Enable Auto Serial Polling" in the "Advanced Items" field.
- Quit the menu with "OK".

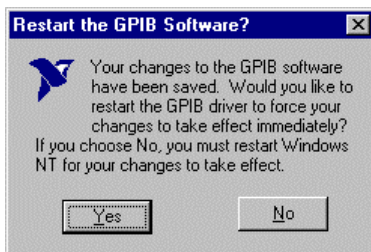
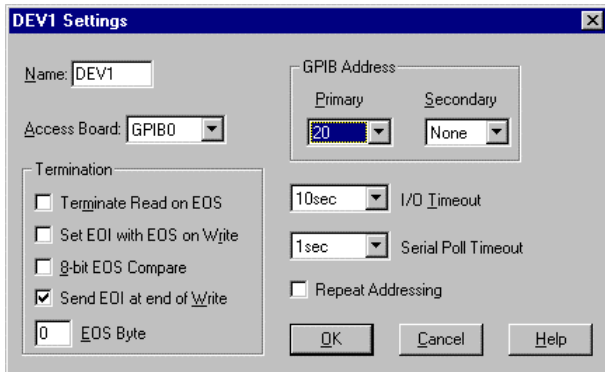
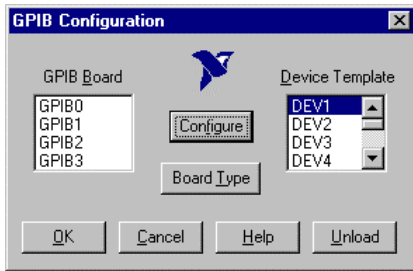
The "GPIB Configuration" menu is displayed again.

**Note:**

*The settings of the following parameters should not be no longer modified.*

```
Board Type           .. AT-GPIB/TNT
Base I/O Address     .. 02C0h
Interrupt Level      .. 3
DMA Channel          .. 5
Enable Auto Serial Polling .. No
```

## Options



### Setting the parameters for the connected equipment

- Mark the instrument in the "Device Template" list and confirm selection with "OK".

The "DEV.. Settings" menu is opened.

- Perform the settings for the selected unit in the "DEV.. Settings" menu.

The logic name for the instrument is preset with DEV1 and address 20. See board manual for further units.

**Note:** when assigning logic names to connected equipment note that these names do not correspond to the directory names under DOS.

- Terminate setting with "OK".

The query asking if the GPIB software should be re-started is displayed.

- Select "No".

- Re-start controller with Start-Restart in the task bar.

After the controller has been re-started, the settings for the GPIB interface are effective.

**Note:** After the installation, the "Service Pack X" of Windows NT is to be re-installed, see "New Installation of Windows NT Software".

## Use of DOS Programs

When using DOS programs, driver GPIB-NT.COM should be loaded. For this to take place, the line device=C:\PROGRA~1\NATION~1\GPIB\NI488\DosWin16\Gpib-nt.com has to be activated in the file C:\WINNT\SYSTEM32\CONFIG.NT. If the option is factory fitted, this line will have been entered in the factory.

### 1.8.1.2 Operation

The second IEC/IEEE-bus interface corresponds physically to that of the instrument (see appendix A). If the instrument is to be controlled via the IEC/IEEE bus, a bus cable must be plugged to both bus connectors.

The interface can be driven under DOS/WINDOWS3.1/95/NT by user-written software. The handling of IEC/IEEE-bus commands in user programs is described in the manual for the card.

The files are in directory C:\Program Files\National Instrument\GPIB\NI488.

## 1.8.2 Option ZVR B16 – Ethernet Adapter

With the option Ethernet Adapter FSE-B16, the device can be connected to an Ethernet-LAN (local area network). It is possible to transmit data via the network and to use the network printer. The adapter operates with a 10 MHz Ethernet in line with standards IEEE 802.3 10Base2 (Thin Ethernet, CheaperNet, BNC-Net) (B16 model 03) or 10Base5 (Thick Ethernet) (B16 model 02).

### 1.8.2.1 Hardware Installation



**Caution:**

*Prior to installation contact the network administrator, especially in case of complex LAN installations since mistakes in cabling might have influence the whole network.*

If the adapter is installed at the factory, it is preconfigured. In case of retrofitting, refer to the installation instructions. The hardware settings must not be modified since the functions of the device could otherwise be impaired.

The following parameters are factory-set:

I/O Addr. 300,      IRQ 5,      MEM D0000

The connection with the network depends on the connectors used in the network.

### BNC (Thin Ethernet, CheaperNet; ZVR-B16 Var. 03)

#### Connection

The device is looped into the LAN segment via rear-panel 2 BNC connectors.

If a cable is not connected to one of the BNC connectors, this BNC connector has to be terminated with 50 Ohm. BNC T connectors must not be used.

#### Network traffic



Note that the network traffic is disturbed if a segment is interrupted.

#### Requirements

Thin Ethernet segment requirements have to be complied with:

- maximum segment length of 185 m
- minimum distance between the connectors of 0.5 m
- maximum of 30 connectors per segment.

If components that comply with enhanced requirements are exclusively used (Ethernet Adapter FSE-B16 complies with the enhanced requirements):

## ***Options***

---

- maximum length of segment of 300 m
- maximum number of connectors of 100

With repeaters used:

- maximum total length of the network of 900 m containing a
- maximum of 3 segments
- maximum of two repeaters between two connectors

## **AUI (Thick Ethernet; ZVR- B16 Var. 02)**

<b>Connection</b>	The device is connected to the LAN segment using a transceiver cable (DB-15 AUI connector, not part of the equipment supplied) which is connected to the rear panel and to the transceiver.
<b>Network traffic</b>	The connection does not conflict with the network traffic. The device can also be disconnected from the network without any problems but make sure that data are not being transmitted.
<b>Requirements</b>	<p>The Thick Ethernet segment requirements have to be taken into account.</p> <ul style="list-style-type: none"><li>- The maximum segment length should be 500 m and the distance between the connectors should at least be 2.5 m.</li><li>- A maximum of 100 connectors may be used in a segment.</li></ul> <p>With repeaters used,</p> <ul style="list-style-type: none"><li>- the total length of the network must not be more than 2500 m containing</li><li>- a maximum of 3 segments.</li><li>- There should be no more than two repeaters between two connectors.</li></ul> <p>If other network components are used, these conditions may vary.</p>

## **RJ45 (UTP, 10BaseT, Western Connector)**

<b>Connection</b>	The device is connected to the LAN segment using a RJ45 cable (not part of the equipment supplied) which is connected to the rear panel and to the network hub of the LAN segment.
<b>Network traffic</b>	This connection does not conflict with the network traffic. The device can also be disconnected from the network without any problems but make sure that data are not being transmitted.
<b>Requirements</b>	<p>Since RJ45 is not a bus but a star topology no special requirements have to be taken into account for the connection.</p> <p>The LAN requirements should be considered in the installation.</p>

### **1.8.2.2 Software Installation**

Data transmission within the network is by means of data blocks, the so-called packets. Besides user data other information, the so-called protocol data (transmitter, receiver, type of data, order) are transmitted. The drivers corresponding to the protocol have to be installed to process protocol information. A network operating system is required for network services (data transmission, directory services, printing in the network) and thus has to be installed.

## Options

---

### Calling up Configuration Menu for Network Settings

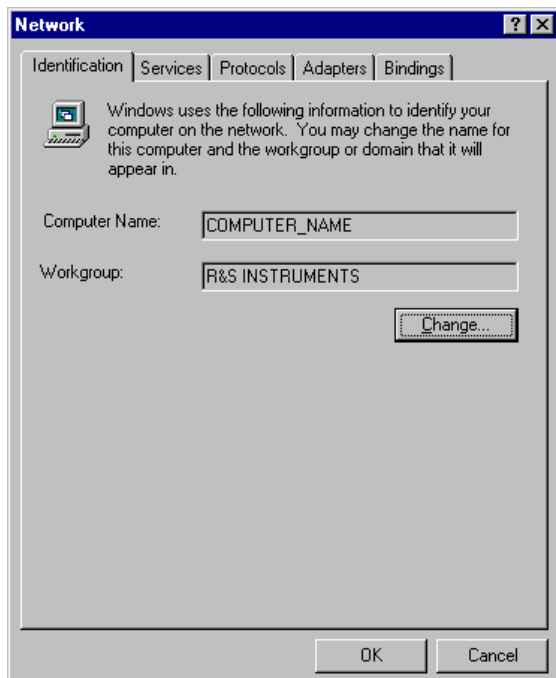
---

- Click "Start" in the task bar.
- Click consecutively "Settings", "Control Panel" and "Network".

The "Network" configuration menu for network settings is opened.

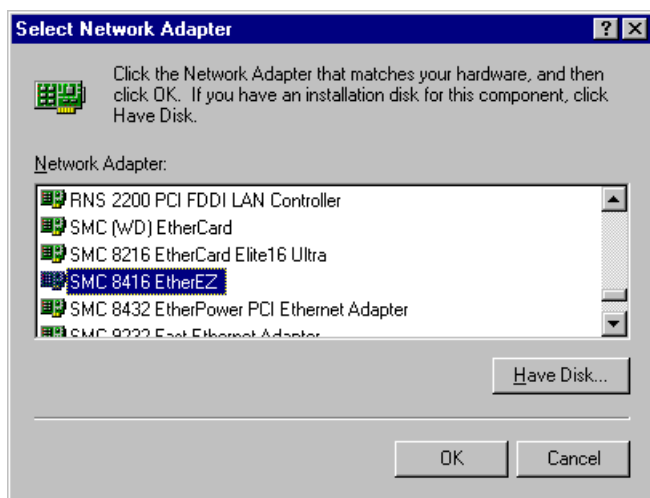
### Registering the Identification

**Note:** It is important for the computer name to be unique in the network..



- Select "Identification".
- Confirm computer and workgroup names with "OK" or enter new names in submenu "Change".

## Installation and Configuration of the Driver for the Network Adapter



- Select "Adapter".
- Click "Add" and mark network driver "SMC 8416 EtherEZ" and select with "OK".  
The query "Files.." is displayed.

- Answer it by clicking "Continue".  
The "SMCEthernet Card Setup" window is displayed.

- Close the window with "OK".  
Some files are copied and the network adapter is displayed under "Network Adapters".  
The entry "MS Loopback Adapter" refers to a driver which ensures instrument control and should not be modified.

**Note:** *The network adapter settings must not be modified since this may cause problems to the instrument.*

## Installation of Network Protocols

**Note:** *The network administrator knows which protocols are to be used..*

- Select "Protocol".
- Click "Add", mark the desired protocol and select with "OK".  
This operation has to be performed several times when several protocols are selected.

- Execute the installation by clicking "Continue".

**Note:** *If a protocol requires further settings, they can be performed with "Properties" after marking the corresponding entry. If further settings are not possible, this field is gray.*

## Options

---

### Installation of Network Services

To utilize the resources of the network it is necessary to install the corresponding services.

**Note:** *The network administrator knows which services are to be used.*

- 
- Select "Services".
  - Click "Add", mark the desired service and select with "OK".

This operation has to be performed several times when several services are selected.

Some services are pre-installed and can be cleared with "Remove" if they are not needed.

- 
- Execute the installation by clicking "Continue".

**Note:** *If a service requires further settings, they can be performed with "Properties" after marking the corresponding entry. If further settings are not possible, this field is gray.*

### Terminating the Installation

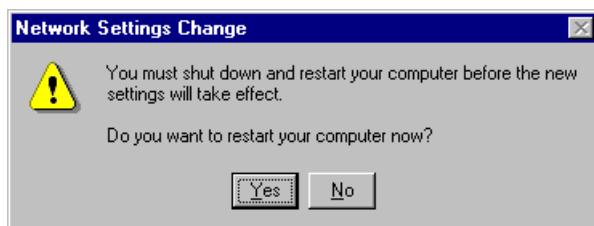
- 
- Quit the "Network" configuration menu for network settings with "OK".

The settings are checked and processed. Missing information is queried.

- 
- Answer the query "You must shutdown..." with "Yes".

The settings are valid after computer re-start.

**Note:** *After the installation, the "Service Pack X" of Windows NT is to be re-installed, see "New Installation of Windows NT Software".*





## Examples of Configurations

Network	Protocols	Services	Notes
NOVELL Netware	NWLink IPX/SPX Compatible Transport	Client Service for NetWare	The "Frame Type" used under "Protocols - Properties" should be set.
IP networks (FTP, TELNET, WWW, GOPHER, etc.)	TCP/IP Protocol	Simple TCP/IP Services	An "IP Address" unambiguous in the network should be set under "Protocols - Properties".
MICROSOFT network	NetBEUI Protocol or TCP/IP Protocol	Workstation Server	A name unambiguous in the network should be registered under "Identification - Computer Name".

### 1.8.2.3 Operation

After installing the network operating system it is possible to transfer data between the device and other computers and to use printers in the network. A precondition for network operation is the authorization to use network resources. Resources may be access to the file directories of other computers or the use of a central printer. The network or server administrator will grant the authorizations. The network name of the resource and the corresponding authorization are required. Passwords protect the resources against improper use. A user name is normally assigned to every authorized user. The user also has a password. Resources may then be assigned to the user. The type of access, i.e. whether data are only read or also written as well as a shared access to data has to be defined. Other types are possible depending on the network operating system.

## NOVELL

Operating system NETWARE from NOVELL is as server-supported system. Data transfer between the individual workstations is not possible. Data transfer is between workstation computers and a central computer, the server. This server provides storage capacity and the connection to the network printers. Like under DOS, the data on a server are organized in directories and are offered to the workstation as virtual drives. A virtual drive on a workstation is like a hard disk and data can be processed accordingly. This is called drive mapping. Also network printers can be addressed as normal printers.

Network operating system NOVELL is available in two forms: NETWARE 3 and NETWARE 4 NDS. In the previous version NETWARE 3, each server manages its own resources itself and is independent. A user has to be managed separately on each server. For NOVELL 4 NDS, all resources are managed in the NDS (NOVELL DIRECTORY SERVICE). The user only has to log in once and gains access to the resources released for him. The individual resources and the user are managed as objects in a hierarchical tree (NDS TREE). The position of the object in the tree is called CONTEXT for NETWARE and must be known in order to access the resources.

## MICROSOFT

For MICROSOFT, data can be transferred between workstations (peer-to-peer) but also between workstations and servers. The servers can provide access to individual files as well as the connection to network printers. Like under DOS, the data on a server are organized in directories and are offered to the workstation as virtual drives. A virtual drive on a workstation is like a hard disk and data can be processed accordingly. This is called drive mapping. Also network printers can be addressed as normal printers. A connection to DOS, WINDOWS FOR WORKGROUPS, WINDOWS95, WINDOWS NT is possible.

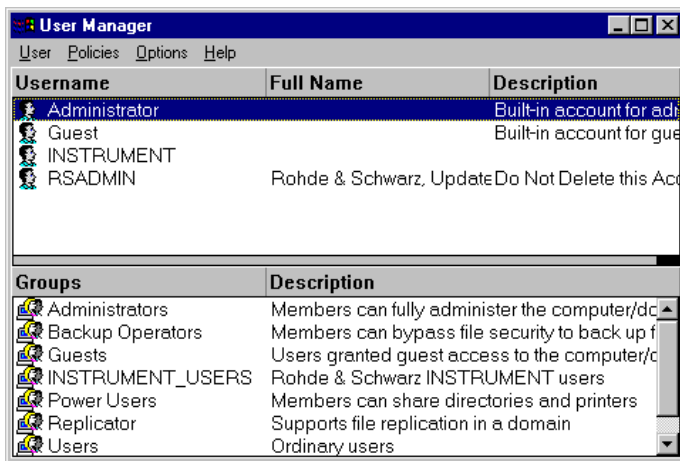
## Options

---

### Installing a user

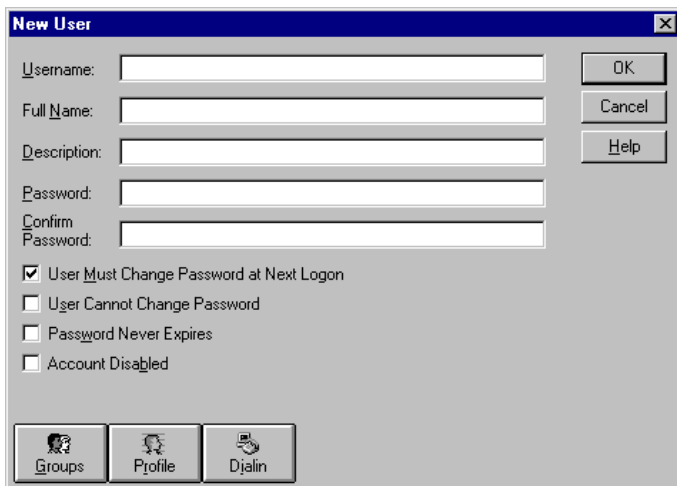
After the network software has been installed, the instrument logs with an error message during the next start-up since there is no user "Instrument" (= user identification for NT autologin) in the network. It is therefore necessary to install a user which should be the same for Windows NT and for the network. The network administrator is responsible for the installation of new users in the network.

**Note:** *The installation of new users is possible only under the administrator identification (see section "Controller Function").*



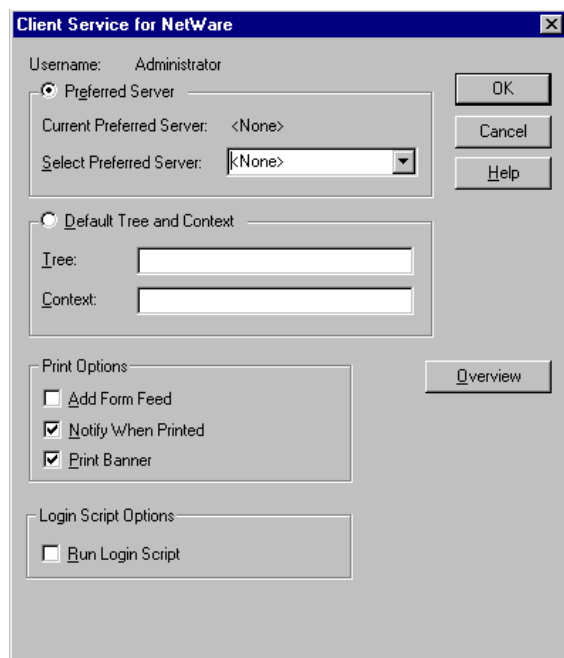
- Click "Start" in the task bar.
  - Click consecutively "Programs" "Administrative Tools (Common)" and "User Manager".
- The "User Manager" menu is opened.

- Click "User" and select "New User".
- The menu "New User" for entering user data is opened.



- Fill in the lines "Username", "Password" and "Confirm Password" and confirm the entry with OK.
- The user data should correspond to the network settings.

### Only NOVELL network: Configure NOVELL Client



- Click "Start" in the task bar.
- Click consecutively "Settings", "Control Panel", "CSNW".

#### NOVELL 3.x

- Click "Preferred Server".
- Under "Select Preferred Server" select the NOVELL server for which the user has been installed.

#### NOVELL 4.x

- Click "Default Tree and Context".
- Enter the NDS Tree under "Tree" and, under "Context", the hierarchical path for which the user has been installed.

**Note:** *These specifications can be obtained from the network administrator.*

### Login in the Network

Network login is automatically performed with the operating system login. A prerequisite is that the user name and the password are the same under Windows NT and in the network.

### Use of Network Drives

- Click "Start" in the task bar.
- Click consecutively "Programs" and "Windows NT Explorer".
- Click "Network" line in the "All Directories" list.

A list of available network drives is displayed.

- Click "Tools" and "Map Network Drive".  
In the list "Shared Directories:" the network paths available in the network are displayed.
- Mark the desired network path.

## Options

---

- Select the drive under "Drive:"
- Activate "Reconnect at Logon:" if the link is to be automatically established at each unit start.
- Connect the network path to the selected drive with "OK".

The user name and the password are queried. The drive is then displayed in the "All Directories" list of Explorer.

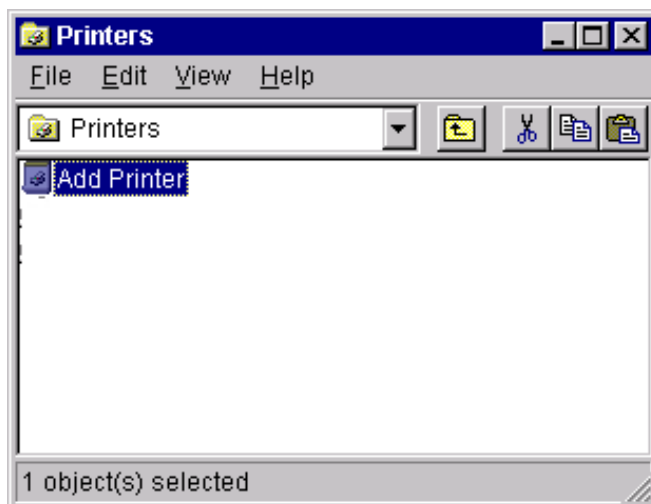
**Note:** *Only drives for which an authorization is available may be connected.*

---

Disconnect link:

- Click "Tools" and "Disconnect Network Drive" in Explorer.
- Select under "Drive:" the drive whose connection is to be removed.
- Disconnect link with "OK". The query should be answered with "Yes".

## Printing on a Network Printer

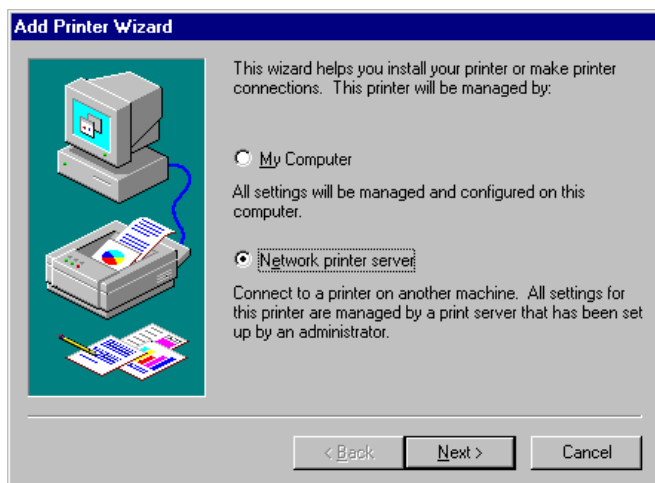


---

### Select printer driver under Windows NT

- Press key combination <ALT> <SYSREQ>  
The Windows NT screen is displayed.

- In the Start menu press first "Setting" and then "Printers".  
The printer window is opened.



- Double-click line "Add Printer".

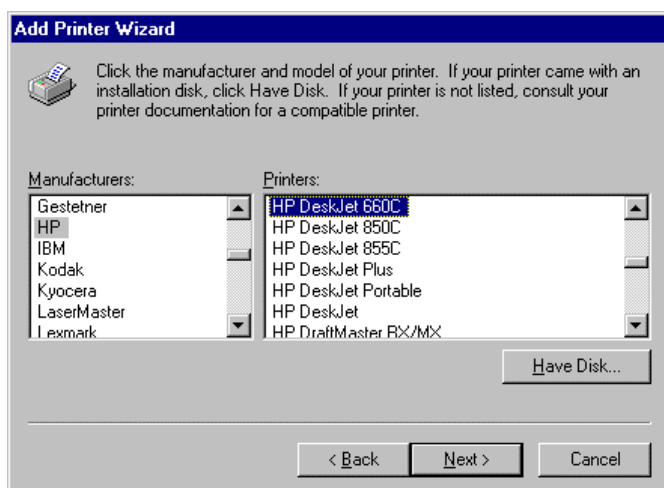
The "Add Printer Wizard" window is opened. This window leads through the following printer driver installation.

- Click first "Network Printer Source" and then "Next".

The list of available network printers is displayed.

- Mark the printer and select with "OK".

The available printer drivers are displayed. The left-hand selection table indicates the manufacturers and the right-hand one the available printer drivers.



- Mark the manufacturer in selection table "Manufacturers" and then the printer driver in selection table "Printers".

## Options

---



- Click "Next".

The window for starting a test page is displayed. The test print is for checking if the installation was successful.

- Switch on printer.
- Click Yes (recommended)".
- Click "Finish".

A test page is printed out if the installation was successful.

If the test page is not printed out or not completely, the Windows NT online help offers troubleshooting instructions under the entry "Printer Trouble Shooting".

---

The instrument has to be configured with this printer for the printout of the measurement screen. This configuration is described in this chapter in the section "Connecting an Output Device".

## Server Function

With the server function data can be provided on the instrument for use in other computers. This is possible only in the MICROSOFT network. The server function is released after network installation as standard. If this is not required, it should be deactivated, see "Installation of Network Services".

The availability of instrument data in the network is controlled by releases. The release is a property of a file or of a directory. To grant a release, the object is to be marked in "Windows NT Explorer" and pressed by the right-hand mouse key. The release is performed under Properties -> Sharing by selecting "Shared As". Other computers can then access these objects with the names allocated under "Share Name". The online help gives further information on the network operation.

## TCP/IP

The TCP/IP protocol allows files to be transmitted between different computer systems. A program running on both computers is required to control the data transfer. The same operating or file system need not be used by the two partners. A file transfer is possible between DOS/WINDOWS and UNIX, for example. One partner has to be configured as host the other as client or vice versa. The system performing several processes at the same time (UNIX) will normally be the host. The usual file transfer program used for TCP/IP is FTP (File Transfer Protocol). An FTP host is installed as standard on most of the UNIX systems.

After installing the TCP/IP services, a terminal link can be established with "Start" - "Programs" - "Accessories" - "Telnet" or a data transmission with FTP using "Start" - "Run" "ftp" - "OK". Thus, all controller systems can be accessed which support these universal protocols (UNIX, VMS, ...).

Further information is given in the NT online help which can be called up with "Help".

## **FTP**

For a complete description of the functions and commands see the FTP documentation.

### **Establishing a connection**

➤ Click "Start" and then "Run" in the task bar.

The program is started with the DOS command

```
FTP
```

The following command sets up the connection:

```
OPEN <xx.xx.xx.xx>
```

xx.xx.xx.xx = IP address, e.g. 89.0.0.13

---

### **File transmission**

To transmit a file to the target system, the following command is used:

```
PUT <file name>
```

file name = name of file e.g. DATA.TXT.

To call a file from the target system, the following command is used:

```
GET<file name>
```

file name = name of file e.g. SETTING.DAT.

```
TYPE B
```

allows the transmission of files in the BINARY format, no conversion is performed.

```
TYPE A
```

allows the transmission of files in the ASCII format. Thus, control characters are converted so that the text files can be read on the target system, too.

#### **Examples:**

```
PUT C:\AUTOEXEC.BAT
```

sends the AUTOEXEC.BAT file to the target system.

```
LCD DATA
```

changes to subdirectory DATA in the computer function.

```
CD SETTING
```

changes to the subdirectory SETTING on the target system.

---

### **Changing directories**

The command

```
LCD <path>
```

changes the directory as the corresponding DOS command.

```
LDIR
```

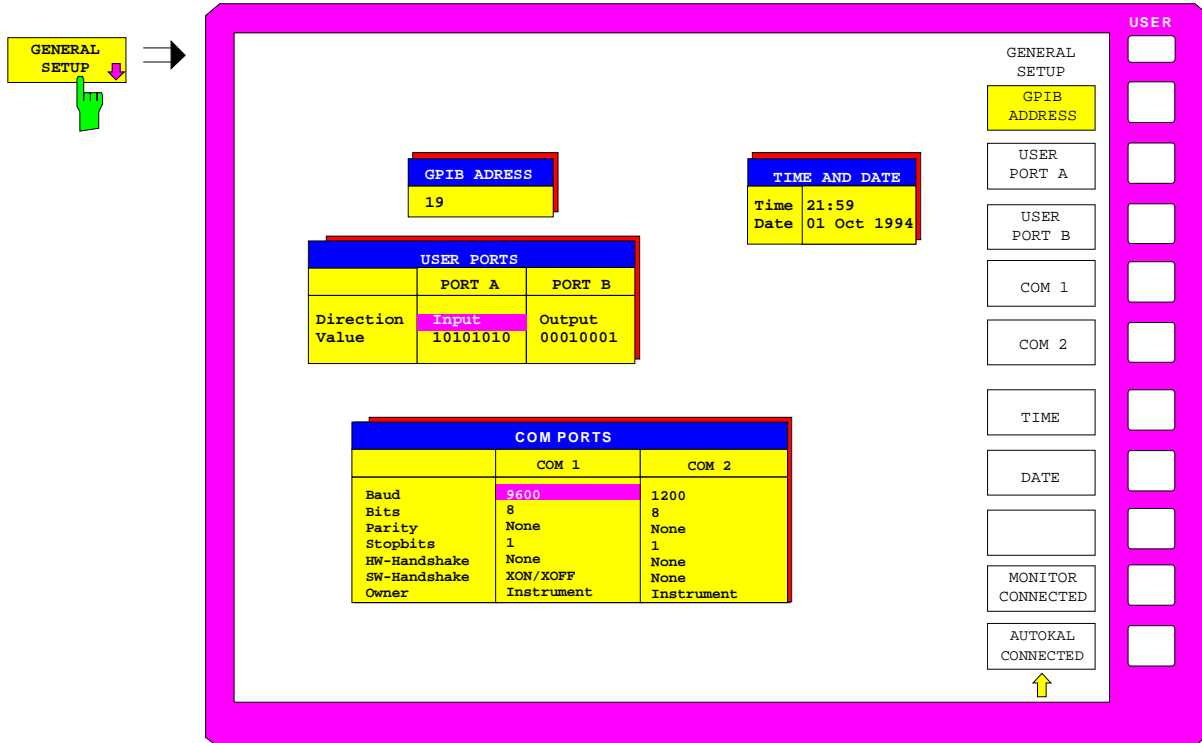
lists the directory.

These commands refer to the computer function of the instrument. If the 'L' preceding the commands is omitted, they apply to the target system.

## 2 Modifications of the *GENERAL SETUP* Menu

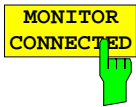
The *GENERAL SETUP* soft key branches to a sub-menu in which the basic setup of general instrument parameters may be performed. The current settings are displayed in tabular form on the display screen where they may then be edited.

*SYSTEM SETUP* menu:



### Connecting the External Monitor

Submenu *SYSTEM SETUP-GENERAL SETUP*:



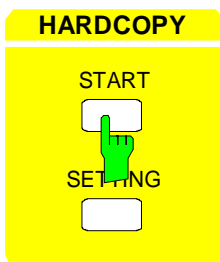
Softkey *MONITOR CONNECTED* informs the unit that an external monitor is available at PC MONITOR connector. The external monitor displays the Windows NT screen.



## 3 Modifications of the *COPY* Menu

### 3.1 Printing Data – *COPY* Key

The instrument uses the printer function of Windows NT to output hardcopies. Any printer supported by Windows NT can be used. In addition, the instrument permits data output in the usual data formats WMF, EWMF and BMP which enable hardcopies to be directly inserted into other documents. Network printers can be used if the instrument is connected to a network.



The *COPY* key initiates the printing of measurement/ instrument status data. The instrument can distinguish between two different output devices, each of which may be individually configured, e.g., a laser printer and a ink jet printer. All documents are printed on the output device which is currently active.

Pressing the *COPY* key initiates the print job. The printer parameters defined under Windows NT and in the *COPY SETTINGS* menu are used for setting up the printer configuration. After pressing the *COPY* key, all of the display items to be printed are written to the printer buffer of Windows NT. Since the printer runs in the background, the instrument may be operated immediately after pressing the *COPY* key.

With *COPY SCREEN* selected in menu *COPY SETTINGS*, all the diagrams with traces and status displays are printed as they occur on the screen. Softkeys, open tables and data entry fields are not printed out. Function *COPY TRACE* allows to print out individual traces. With *COPY TABLE* tables can be printed out.

If output device *CLIPBOARD* is active in menu *COPY SETTINGS*, submenu *COPY DEVICE*, the clipboard can be used to directly transfer hardcopies to Windows applications. The copy is written in the clipboard by pressing *COPY*. The user can then change to another Word document and insert the clipboard content into the document via the menu EDIT - PASTE by using the key combination CTRL+V.

If the *PRINT TO FILE* option in the *COPY DEVICE* submenu of the *COPY SETTINGS* menu is selected then, upon pressing *COPY*, the file name to which the output data are to be written is requested. For this an entry field is opened for entering the file name. If no external keyboard is connected, the help line editor is activated.

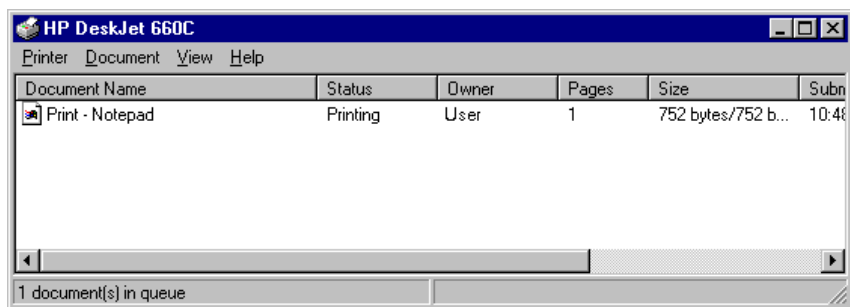
If the *COPY* key is pressed again during an active print job, a second output can be released which can also be joined to the printer queue. Any number of print jobs can be released consecutively.

Current print jobs can be aborted only by canceling the entries in the Windows NT printer queue. After starting the print a printer symbol is displayed in the task bar near the time indication.

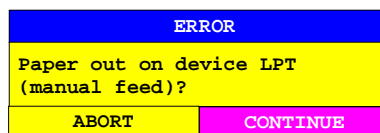


A double-click on this symbol opens a window containing the entries of the printer queue. The relevant print order can be cancelled by marking it with the mouse and pressing the DEL key.

## Menu **HARDCOPY**



While a print job is in progress, problems may occur in the output device. If, while printing, the output device issues a PAPER OUT message, i.e., no more paper is available, the user will be prompted by the following message



to load paper into the output device. The print job will then be either continued (CONTINUE selected) or aborted (ABORT selected).

Switchover between b/w and colored printouts is possible with softkey *COLOR ON/OFF* provided that the printer connected is able to output such prints. The colors of the printout correspond exactly to those of the screen, ie a red trace will be output in red.

To change the colors of the objects on the printout, the screen colors have to be changed correspondingly in menu *RESPONSE DISPLAY*, submenu *CONFIG DISPLAY*. One exception is the color of the background and the color of the diagrams. The output background is always white irrespective of the screen color and the diagrams are always black.

If several traces are to be output one after the other on the same sheet different colors can be chosen for each trace with the aid of softkey *TRC COLOR AUTO INC* (Trace Color Auto Increment).

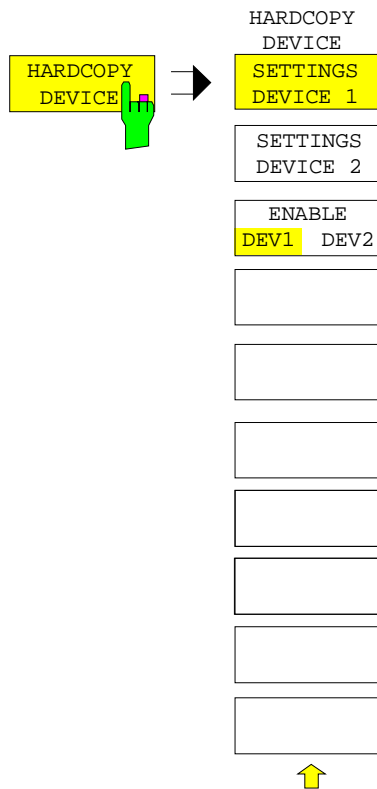
On most b/w printers, a better printout of the color screen is obtained on hardcopies if the color information is converted into gray shades. For this, the color output in menu *COPY SETTINGS* is to be activated (COLOR ON).

### 3.2 Configuration of the Output Device – SETTING Key

The instrument permits two different output devices to be configured. One of the devices is defined as the active device and can be used for hardcopies.

The installation and configuration of these output devices is mainly done under Windows NT and is valid for all Windows applications (see Chapter 1, section "Connecting an Output Device"). The active device and the settings which concern only the output of hardcopies are selected in menu *COPY DEVICE*.

*COPY SETTING* submenu:



The *HARDCOPY DEVICE* soft key switches to the sub-menu which is used for the selection/configuration of the two output devices.

When the sub-menu is called, the corresponding table is simultaneously displayed. The *SETTINGS DEVICE 1* soft key is active and the selection bar is located at the *DEVICE1* line in the corresponding column of the table.

HARDCOPY DEVICE SETTINGS	
Device1	WINDOWS METAFILE
Print to File	YES
Orientation	---
Device2	CLIPBOARD
Print to File	---
Orientation	---



After calling the menu, the *SETTINGS DEVICE 1* soft key is active and permits the selection and configuration of the output device *DEVICE 1*. The *SETTINGS DEVICE 2* soft key is used to configure *DEVICE 2*.



The actual selection of the active output device takes place with the *ENABLE DEV1 DEV2* soft key in the *HARDCOPY DEVICE* sub-menu.

## Menu **HARDCOPY**

### Device

The selection of the output device/language for DEVICE 1 and DEVICE 2 is made in this line.

HARDCOPY DEVICE SETTINGS	
Device1	WINDOWS METAFILE
Print to File	YES
Orientation	---
Device2	CLIPBOARD
Print to File	---
Orientation	---

DEVICE
✓ CLIPBOARD
WINDOWS METAFILE
ENHANCED METAFILE
BITMAP FILE
HP DeskJet 660C

Three file formats and the Windows NT clipboard are always available, even if a printer has not yet been installed under Windows NT. All installed printers are listed below in alphabetic order.

The printer installation is described in Chapter 1, section 1.6.

**CLIPBOARD** When "Clipboard" is selected the hardcopies are copied to the Windows NT clipboard. This permits to obtain a printout of high quality which can be directly inserted into other Windows applications (menu EDIT | PASTE or key combination CTRL+V). The lines 'Print to File', 'Orientation' and 'GPIB Address' are deactivated.

### WINDOWS METAFILE and ENHANCED METAFILE

WMF and EWMF are vector graphics formats which can be imported by most graphics and editing programs. EMF is recommended for recent Windows32 applications.

**BITMAP FILE** BMP is a bitmap format which can also be imported by most programs.

When WMF, EWMF and BMP are selected, the line "Print to File" is automatically set to ON and line "Orientation" deactivated.

### Print to File

With "Print to File" ON, the printout is directed to a file. In this case the user is prompted to enter a file name on calling up *COPY*.

*Note: This setting is coupled to the corresponding setting under Windows NT.*

### Orientation

In this line, the print format of the output page is set to either vertical (= PORTRAIT) or horizontal (= LANDSCAPE).



The *ENABLE DEV1 / DEV2* soft key determines the active output device. The default output device is DEVICE 1, i.e., all output takes place on DEVICE 1.

## 4 Modifications of the IEC/IEEE-Bus Commands

### 4.1 HCOPy-System

The commands `HCOPy:DEVIce:PRESet` and `HCOPy:DEVIce:RESolution` are omitted.

#### `HCOPy:DESTination<1|2>` <string>

This command selects the device for outputting the hardcopy.

**Parameter:** <string> ::= 'MMEM' | 'SYST:COMM:PRIN' | 'SYST:COMM:CLIP'

**Example:** "HCOP:DEST2 'SYST:COMM:SER2' "

**Features:** \*RST value: -  
SCPI: conforming

This command is an event which is why it is not assigned an \*RST value and has no query.

'MMEM' creates a file for the hardcopy output. Command `MMEM:NAME` <file\_name> defines the file name. All formats can be selected with `HCOPy:DEVIce:LANGUage`.

'SYST:COMM:PRIN' directs the hardcopy to the printer. The printer is selected with command `SYSTEM:COMMunicate:PRINter:SElect`. GDI should be selected for `HCOPy:DEVIce:LANGUage`.

'SYST:COMM:CLIP' directs the hardcopy to the clipboard. EWMF should be selected for `HCOPy:DEVIce:LANGUage`.

#### `HCOPy:DEVIce:LANGUage<1|2>` WMF|GDI|EWMF|BMP

This command determines the data format of the printout. The selection depends on the setting under `HCOPy:DESTination`.

**Example:** "HCOP:DEV:LANG WMF"

**Features:** \*RST value: -  
SCPI: conforming

WMF and EWMF (WINDOWS Metafile and Enhanced Metafile Format)  
Data formats for output files which can at a later time be integrated in corresponding programs for documentation purposes.

BMP (Bitmap) Data format for output files.

GDI (Graphics Device Interface) Default format for the output to a printer configured under Windows. For the output file the printer driver configured under Windows is used and thus a printer-specific format is generated.

## 4.2 SYSTEM-System

### SYSTEM:COMMunicate:PRINter<1|2>:ENUMerate:FIRSt?

This command queries the name of the first printer (in the list of printers) under Windows NT. The numeric suffix in `PRINter<1|2>` is not significant.

The names of other installed printers can be queried with command `SYSTEM:COMMunicate:PRINter:ENUMerate:NEXT?`.

If no printer is configured an empty string is output.

**Example:**        "`SYST:COMM:PRIN:ENUM:FIRS?`"

**Features:**     \*RST value:    -  
                  SCPI:           device-specific

### SYSTEM:COMMunicate:PRINter<1|2>:ENUMerate:NEXT?

This command queries the name of the next printer installed under Windows NT. The numeric suffix in `PRINter<1|2>` is not significant.

This command can only be sent after command `SYSTEM:COMMunicate:PRINter:ENUMerate:FIRSt?`.

An empty string is output after all printer names have been output.

**Example:**        "`SYST:COMM:PRIN:ENUM:NEXT?`"

**Features:**     \*RST value:    -  
                  SCPI:           device-specific

### SYSTEM:COMMunicate:PRINter<1|2>:SELEct <printer\_name>

This command selects one of the printers installed under Windows NT. The numeric suffix in `PRINter<1|2>` selects the device.

The name of the first printer is queried with `FIRSt?`. After that the names of other installed printers can be queried with `NEXT?`.

**Parameter:**    <printer\_name> ::= string which has been queried with commands `SYSTEM:COMMunicate:PRINter:ENUMerate:FIRSt?` and `NEXT?`.

**Example:**        "`SYST:COMM:PRIN:SEL 'HP_DESKJET660'`"

**Features:**     \*RST value:    -  
                  SCPI:           device-specific

## 5 RSIB Interface

The instrument is equipped with an RSIB interface that allows instrument control by Windows applications WinWord and Excel or by Visual C++ and Visual Basic programs. Control applications may run locally on the measuring instrument as well as on a remote controller in a network.

### 5.1 Remote Control via RSIB Interface

To access the measuring instruments via the RSIB interface the DLLs should be installed in the corresponding directories:

- RSIB.DLL in Windows NT *system* directory or control application directory.
- RSIB32.DLL in Windows NT *system32* directory or control application directory.

The DLLs are already installed in the corresponding directories on the measuring instruments.

For the different programming languages, files exist containing the declarations for the DLL functions and the error code definitions.

Visual Basic (16 bit):	'RSIB.BAS'	(C:/R_S/Instr\RSIB)
Visual Basic (32 bit):	'RSIB32.BAS'	(C:/R_S/Instr\RSIB)
C:	"RSIBC.H"	(C:\R_S\Instr\RSIB)
WinWord:	"RSIBWB.BAS"	(C:\R_S\Instr\RSIB)

The RSIB directory also contains the program RSIBCNTR.EXE with which SCPI commands can be sent to the instrument via the RSIB interface. This program can be used to test the function of the interface. Required for the test is delay module VBRUN300.DLL in the path or in the Windows directory.

The control is performed via one of the Windows applications WinWord or Excel or with Visual C++ or Visual Basic programs. The local link to the internal controller is established with the name '@local'. If a remote controller is used, the instrument IP address is to be indicated here.

<b>Via VisualBasic:</b>	internal controller:	ud = RSDLLibfind("@local", ibsta, iberr, ibcntl)
	remote controller:	ud = RSDLLibfind("82.1.1.200", ibsta, iberr, ibcntl)

The return to manual operation can be performed via the front panel (*LOCAL* key) or the RSIB interface.

<b>Via RSIB:</b>	...
	ud = RSDLLibloc (ud, ibsta, iberr, ibcntl);
	...

## 5.2 RSIB Interface Functions

This section describes all functions of DLL "RSIB.DLL" or "RSIB32.DLL" by means of which control applications can be written.

### 5.2.1.1 Variables `ibsta`, `iberr`, `ibcntl`

Same as with the National Instrument interface, successful execution of a command can be checked by means of the variables `ibsta`, `iberr` and `ibcntl`. To this end, references to the three variables are transferred to all RSIB functions. In addition, the status word `ibsta` is returned as a function value by all functions.

#### Status word `ibsta`

All functions send back a status word that provides information on the status of the RSIB interface. The following bits are defined:

Bit name	Bit	Hex code	Description
ERR	15	8000	This bit is set if an error occurs during a function call. If this bit is set, <code>iberr</code> contains an error code which specifies the error.
TIMO	14	4000	This bit is set if a timeout occurs during a function call. A timeout may occur in the following situations: <ul style="list-style-type: none"> <li>while waiting for an SRQ with the function <code>RSDLLWaitSrq()</code>.</li> <li>if no acknowledgment is received for data sent to an instrument with <code>RSDLLibwrt()</code> or <code>RSDLLilwrt()</code>.</li> <li>if no response from server to a data request with function <code>RSDLLibrd()</code> or <code>RSDLLilrd()</code>.</li> </ul>
CMPL	8	0100	This bit is set if the reply of the IEC/IEEE-bus parser is completely read. If a reply of the parser is read with the function <code>RSDLLilrd()</code> and the buffer length is not sufficient, the bit is cleared.

#### Error variable `iberr`

If the ERR bit (8000h) is set in the status word, `iberr` contains an error code that specifies the error. The RSIB has error codes of its own independent of the National Instrument interface.

Error	Error code	Description
IBERR_DEVICE_REGISTER	1	RSIB.DLL cannot register any new device.
IBERR_CONNECT	2	Link to the device has failed.
IBERR_NO_DEVICE	3	An interface function was called with an invalid device handle.
IBERR_MEM	4	No free memory available.
IBERR_TIMEOUT	5	Timeout has occurred.
IBERR_BUSY	6	The RSIB interface is blocked by a function not yet completed. Windows is not blocked, for example, by function <code>RSDLLibrd()</code> if data are still to be transmitted in response to this function. In this case a new call is possible. Further calls are however rejected by RSIB.DLL with error code <code>IBERR_BUSY</code> .
IBERR_FILE	7	Error in reading from or writing to a file.

#### Count variable - `ibcntl`

The variable `ibcntl` is updated with the number of bytes transmitted on every read and write function call.



**5.2.1.2 List of Interface Functions**

The DLL functions are matched to the interface functions for IEC/IEEE-bus programming from National Instruments. Functions supported by the DLLs are listed in the following table.

<b>Function</b>	<b>Description</b>
RSDLLibfind()	Provides a handle for accessing a device.
RSDLLibwrt()	Sends a string terminated with a null to a device.
RSDLLilwrt()	Sends a specific number of bytes to a device.
RSDLLilwrtf()	Sends the contents of a file to a device.
RSDLLibrd()	Reads data from a device into a string.
RSDLLilrd()	Reads a specific number of bytes from a device.
RSDLLilrdf()	Reads data from a device into a file.
RSDLLibtmo()	Sets timeout period for RSIB functions.
RSDLLibsre()	Switches a device to local or remote.
RSDLLibloc()	Switches a device temporarily to local.
RSDLLibeot()	Enables/disables END message in write operations.
RSDLLibrsp()	Starts a serial poll and reads the status byte.
RSDLLibonl()	Switches the instrument online/offline.
RSDLLTestSrqr()	Checks if a device has generated an SRQ.
RSDLLWaitSrqr()	Waits until a device generates an SRQ.

### 5.2.1.3 Description of Interface Functions

#### RSDLLibfind()

The function provides a handle for accessing the instrument with the name `udName`.

**VB format:** Function RSDLLibfind (ByVal udName\$, ibsta%, iberr%, ibcntl&) As Integer

**C format:** short FAR PASCAL RSDLLibfind( char far \*udName, short far \*ibsta, short far \*iberr, unsigned long far \*ibcntl)

**Parameter:** udName Instrument name

**Example:** ud = RSDLLibfind ("@local", ibsta, iberr, ibcntl)

The function must be called before all other interface functions.

The function supplies a handle as a return value. The handle must be specified in all functions for accessing the instrument. If no instrument with the name `udName` is found, the handle will take on a negative value.

The local link on the instrument is established with the name "@local". For setting up a link via a network, on the other hand, the IP address of the instrument must be given (eg "89.1.1.200").

#### RSDLLibwrt

The function sends data to the instrument with the handle `ud`.

**VB format:** Function RSDLLibwrt (ByVal ud%, ByVal Wrt\$, ibsta%, iberr%, ibcntl&) As Integer

**C format:** short FAR PASCAL RSDLLibwrt( short ud, char far \*Wrt, short far \*ibsta, short far \*iberr, unsigned long far \*ibcntl )

**Parameter:** ud Device handle  
Wrt String sent to the instrument.

**Example:** RSDLLibwrt(ud, "SENS:FREQ:STAR?", ibsta, iberr, ibcntl)

The function allows setting and query commands to be sent to the measuring instruments. The function `RSDLLibeot()` is used for defining whether the data are interpreted as a complete command.

#### RSDLLilwrt

The function sends `Cnt` bytes to the instrument with the handle `ud`.

**VB format:** Function RSDLLilwrt (ByVal ud%, ByVal Wrt\$, ByVal Cnt&, ibsta%, iberr%, ibcntl&) As Integer

**C format:** short FAR PASCAL RSDLLilwrt( short ud, char far \*Wrt, unsigned long Cnt, short far \*ibsta, short far \*iberr, unsigned long far \*ibcntl)

**Parameter:** ud Device handle  
Wrt String sent to the IEC/IEEE-bus parser.  
Cnt Number of bytes sent to the instrument.

**Example:** RSDLLilwrt (ud, '.....', 100, ibsta, iberr, ibcntl)

This function sends data to an instrument same as function `RSDLLibwrt()` but with the difference that binary data, too, can be sent. The length of the data is determined not by a zero-terminated string but by the definition of `Cnt` bytes. If the data are to be terminated with EOS (0Ah), the EOS byte must be appended to the string.

## RSDLLibwrtf

The function sends the contents of a file to the instrument with the handle ud.

**VB format:** Function RSDLLibwrtf (ByVal ud%, ByVal file\$, ibsta%, iberr%, ibcntl&) As Integer

**C format:** short FAR PASCAL RSDLLibwrtf( short ud, char far \*Wrt, short far \*ibsta, short far \*iberr, unsigned long far \*ibcntl )

**Parameter:** ud Device handle  
file File whose contents are sent to the instrument.

**Example:** RSDLLibrdf(ud, "C:\db.sav", ibsta, iberr, ibcntl)

The function allows setting and query commands to be sent to the measuring instruments. The function RSDLLibeot() is used for defining whether the data are interpreted as a complete command.

## RSDLLibrd()

The function reads data from the instrument into the string Rd with the handle ud.

**VB format:** Function RSDLLibrd (ByVal ud%, ByVal Rd\$, ibsta%, iberr%, ibcntl&) As Integer

**C format:** short FAR PASCAL RSDLLibrd( short ud, char far \*Rd, short far \*ibsta, short far \*iberr, unsigned long far \*ibcntl )

**Parameter:** ud Device handle  
Rd String into which the read data are copied.

**Example:** RSDLLibrd (ud, Rd, ibsta, iberr, ibcntl)

This function fetches replies of the IEC/IEEE-bus parser in response to a query command.

For this, a string of sufficient length must be generated when programming in Visual Basic. This can be accomplished in the definition of the string or with the command Space\$().

Generation of a string of the length 100: - Dim Rd as String \* 100

- Dim Rd as String

Rd = Space\$(100)

## RSDLLilrd

The function reads Cnt bytes from the instrument with the handle ud.

**VB format:** Function RSDLLilrd (ByVal ud%, ByVal Rd\$, ByVal Cnt&, ibsta%, iberr%, ibcntl&) As Integer

**C format:** short FAR PASCAL RSDLLilrd( short ud, char far \*Rd, unsigned long Cnt, short far \*ibsta, short far \*iberr, unsigned long far \*ibcntl )

**Parameter:** ud Device handle  
cnt Maximum number of bytes copied to target string Rd by the DLL.

**Example:** RSDLLilrd (ud, Rd, 100, ibsta, iberr, ibcntl)

The function reads data from an instrument same as function RSDLLibrd() but with the difference that here the maximum number of bytes is allowed to be copied to target string Rd to be defined with Cnt.

Writing beyond a string can thus be prevented with this function. The number of bytes cut off is lost.

### RSDLLibrdf()

Reads data from the instrument into the file `file` with the handle `ud`.

**VB format:** Function RSDLLibrdf (ByVal ud%, ByVal file\$, ibsta%, iberr%, ibcntl&) As Integer

**C format:** short FAR PASCAL RSDLLibrdf( short ud, char far \*file, short far \*ibsta, short far \*iberr, unsigned long far \*ibcntl )

**Parameter:** ud Device handle  
file File into which the read data are written.

**Example:** RSDLLibrdf (ud, "c:\db.sav", ibsta, iberr, ibcntl)

This function serves to read replies of the IEC/IEEE-bus parser that are larger than 64 KB. The file name may include the drive and path.

### RSDLLibtmo

The function defines the timeout for an instrument. The default value for the timeout is 5 seconds.

**VB format:** Function RSDLLibtmo (ByVal ud%, ByVal tmo%, ibsta%, iberr%, ibcntl&) As Integer

**C format:** void FAR PASCAL RSDLLibtmo( short ud, short tmo, short far \*ibsta, short far \*iberr, unsigned long far \*ibcntl )

**Parameter:** ud Device handle  
tmo Time in seconds

**Example:** RSDLLibtmo (ud, 10, ibsta, iberr, ibcntl)

Timeout can occur in the following cases:

- Waiting for an SRQ with the function RSDLLWaitSrq().
- Waiting for an acknowledgment for data sent to an instrument with RSDLLibwrt() or RSDLLilwrt().
- Waiting for a reply to a data request made with function RSDLLibrd() or RSDLLilrd().

### RSDLLibsre

The function switches the instrument to LOCAL or REMOTE mode.

**VB format:** Function RSDLLibsre (ByVal ud%, ByVal v%, ibsta%, iberr%, ibcntl&) As Integer

**C format:** void FAR PASCAL RSDLLibsre( short ud, short v, short far \*ibsta, short far \*iberr, unsigned long far \*ibcntl )

**Parameter:** ud Device handle  
v Instrument status  
0 - local  
1 - remote

**Example:** RSDLLibsre (ud, 0, ibsta, iberr, ibcntl)

## RSDLLibloc

The function switches the instrument temporarily to LOCAL mode.

**VB format:** Function RSDLLibloc (ByVal ud%, ibsta%, iberr%, ibcntl%) As Integer

**C format:** void FAR PASCAL RSDLLibloc( short ud, short far \*ibsta, short far \*iberr, unsigned long far \*ibcntl)

**Parameter:** ud Device handle

**Example:** RSDLLibloc (ud, ibsta, iberr, ibcntl)

After switchover the instrument can be manually operated from the front panel. On the next access to the instrument by means of one of the functions of the RSIB.DLL, the instrument is switched again to the REMOTE state.

## RSDLLibeot

The function enables the END message after write operations or disables it.

**VB format:** Function RSDLLibeot (ByVal ud%, ByVal v%, ibsta%, iberr%, ibcntl%) As Integer

**C format:** void FAR PASCAL RSDLLibeot( short ud, short v, short far \*ibsta, short far \*iberr, unsigned long far \*ibcntl)

**Parameter:** ud Device handle.

v 0 - no END message  
1 - send END message

**Example:** RSDLLibeot (ud, 1, ibsta, iberr, ibcntl)

If the END message is disabled, the data of a command can be sent by means of several successive write function calls. The END message must be enabled prior to the last data block.

## RSDLLibrsp

The function carries out a Serial Poll and outputs the instrument status.

**VB format:** Function RSDLLibrsp(ByVal ud%, spr%, ibsta%, iberr%, ibcntl%) As Integer

**C format:** void FAR PASCAL RSDLLibrsp( short ud, char far\* spr, short far \*ibsta, short far \*iberr, unsigned long far \*ibcntl)

**Parameter:** ud Device handle.

spr Reference to status byte

**Example:** RSDLLibrsp(ud, spr, ibsta, iberr, ibcntl)

## RSIB Interface

---

### RSDLLibonl

The function switches the instrument to the 'online' or 'offline' state. When switching to 'offline' the interface is enabled and the device handle made invalid. The next call of RSDLLibfind sets up the communication again.

**VB format:** Function RSDLLibonl (ByVal ud%, ByVal v%, ibsta%, iberr%, ibcntl&) As Integer

**C format:** void FAR PASCAL RSDLLibonl( short ud, short v, short far \*ibsta, short far \*iberr, unsigned long far \*ibcntl)

**Parameter:**

ud	Device handle.
v	Status of instrument
	0 - local
	1 - remote

**Example:** RSDLLibonl(ud, 0, ibsta, iberr, ibcntl)

### RSDLLTestSRQ

The function checks the status of the SRQ bit.

**VB format:** Function RSDLLTestSrq (ByVal ud%, Result%, ibsta%, iberr%, ibcntl&) As Integer

**C format:** void FAR PASCAL RSDLLTestSrq( short ud, short far \*result, short far \*ibsta, short far \*iberr, unsigned long far \*ibcntl)

**Parameter:**

ud	Device handle
result	Reference to an integer value in which the library returns the status of the SRQ bit.
	0 - no SRQ
	1 - SRQ active, instrument has output a service request

**Example:** RSDLLTestSrq (ud, result%, ibsta, iberr, ibcntl)

The function corresponds to the function RSDLLWaitSrq but with the difference that RSDLLTestSRQ returns immediately the current status of the SRQ bit whereas RSDLLWaitSrq waits for an SRQ to occur.

### RSDLLWaitSrq

The function waits until the instrument triggers an SRQ with the handle ud.

**VB format:** Function RSDLLWaitSrq (ByVal ud%, Result%, ibsta%, iberr%, ibcntl&) As Integer

**C format:** void FAR PASCAL RSDLLWaitSrq( short ud, short far \*result, short far \*ibsta, short far \*iberr, unsigned long far \*ibcntl)

**Parameter:**

ud	Device handle
result	Reference to an integer value in which the library returns the status of the SRQ bit.
	0 - no SRQ occurred within the timeout
	1 - SRQ occurred within the timeout

**Parameter:** RSDLLWaitSrq( ud, result, ibsta, iberr, ibcntl );

The function waits until one of the two following events occurs:

- The instrument triggers an SRQ.
- No SRQ occurs during the timeout defined with RSDLLibtmo( ).

## 5.3 Programming via the RSIB Interface

The following hints apply to both the 16-bit and the 32-bit DLL versions (RSIB.DLL and RSIB32.DLL) unless an explicit distinction is made.

The RSIB interface supports links to max. 16 measuring instruments at the same time.

### Debugging Hints:

The RSIB interface uses a "Heartbeat" mechanism for monitoring the connection. This means that check information is exchanged every 5 second and is to be answered, otherwise the connection to the firmware is aborted. During debugging the heartbeat cannot be sent. To prevent an automatic call clear-down during debugging, the following registry entry has to be generated in the instrument.

```
REGEDIT4
[HKEY_LOCAL_MACHINE\SOFTWARE\Rohde&Schwarz\RSIB]
"HeartBeat"=dword:00000000
```

The entry can be manually set up with registry editor "Regedit". Another possibility is to write the above 3 lines in an ASCII file with the extension ".reg". A subsequent double click in the Windows NT Explorer registers this entry.

If the entry is not available (normal status) or "Heartbeat" differs from 0, the "Heartbeat" is activated. The entry should be removed after debugging.

### 5.3.1 Visual Basic

#### Programming Hints:

- Access to RSIB.DLL functions

To generate Visual Basic control applications, the file RSIB.BAS for 16-bit applications or RSIB32.BAS for 32-bit applications (C:\R\_S\INSTR\RSIB) is added to a project to enable call-up of the RSIB.DLL/RSIB32.BAS functions.

- Declaration of DLL functions as procedures

All DLL functions return an integer value. All functions in the file RSIB.BAS are therefore declared as follows:

```
Declare Function RSDLLxxx Lib "rsib.dll" ( ... ) As Integer
```

The function value with the status variable `ibsta` must be assigned a variable for every call. Since this value is also returned via a reference parameter of the functions, it is possible to declare the functions as procedures in the following way:

```
Declare Sub RSDLLxxx Lib "rsib.dll" ( ... )
```

- Generation of reply buffer

Since the DLL returns zero-terminated strings as replies, a string of sufficient length must be generated before functions `RSDLLibrd()` and `RSDLLilrd()` are called because Visual Basic assigns the strings a length value which is not updated by the DLL. The string length data can be generated with one of the following:

```
-Dim Rd as String * 100
-Dim Rd as String
Rd = Space$(100)
```

- Reading trace data in real format

## ***RSIB Interface***

---

When the function declarations of the file RSIB.BAS are used, replies from the instrument can be assigned to only one string. Trace data can be read faster and processing is simpler, however, if data are read in real format into float arrays. Data can be assigned to an array with float values as follows:

Function declaration of RSDLLibrd() in RSIB.BAS:

```
Declare Function RSDLLibrd Lib "rsib.dll" (ByVal ud%, ByVal Rd$,  
ibsta%, iberr%, ibcntl&) As Integer
```

To read data directly into an array with real numbers, the string variable must be replaced by a suitable structure which can be defined as follows:



```
Type TRACEREAL
  len As String * 6      ' Header of real data "#42000"
  Points(500) As Single  ' Float-Array
End Type
```

**Note:** *the structure must be defined in a code module.*

To be able to transfer the structure as a reference parameter to the DLL, a special function declaration must be created.

```
eg Declare Function RSDLLibrdTraceReal Lib "rsib.dll" Alias "RSDLLibrd"
  (ByVal ud%, rd as Any, ibsta%, iberr%, ibcntl&) As Integer
```

Using this function, trace data can be read into a reply buffer of the type TRACEREAL. This mechanism is operational only for 16-bit Basic versions. With 32-bit Basic programs the header and useful data must be read by means of two separate function calls.

### Programming examples:

- In the following example, the start frequency of the local instrument is queried.

```
Dim ibsta As Integer      ' Status variable
Dim iberr As Integer     ' Error variable
Dim ibcntl As Long      ' Count variable
Dim ud As Integer        ' Unit descriptor (handle) for measuring
                          ' instrument
Dim Cmd As String        ' Command string

' Setting up link to measuring instrument
ud = RSDLLibfind("@local", ibsta, iberr, ibcntl)
If (ud < 0) Then
  ' error handling
End If

' Sending query command to instrument
Cmd = "SENS:FREQ:STAR?"
If (RSDLLibwrt(ud, Cmd, ibsta, iberr, ibcntl) And IBSTA_ERR) Then
  ' error handling
End If

' Making space for response
Response = Space$(100)

' Querying response of instrument
If (RSDLLibrd(ud, Response, ibsta, iberr, ibcntl) And IBSTA_ERR) Then
  ' error handling
End If
```

### **5.3.2 C / C++**

#### **Programming Hint:**

Access to RSIB.DLL functions

The functions of RSIB.DLL are declared in the header file RSIBC.H. The DLL functions can be incorporated in a C/C++ program in three different ways:

1. By generating the import library RSIB.LIB by means of IMPLIB.EXE from RSIB.DLL and adding the library to the project.
2. By specifying the functions of RSIB.DLL in the module definition file (\*.def) under IMPORTS.
3. By loading the library with the LoadLibrary() function while the program is running and determining the pointers of the DLL functions with GetProcAddress(). Prior to program end, RSIB.DLL must be freed with function FreeLibrary().

For the first two ways of incorporating a DLL, the latter will be automatically loaded immediately before the start of the application. At the end of the program, the DLL will be freed provided it is not used by other applications.

**Programming example:**

In the following C program, a single sweep is started on the instrument with the IP address 89.1.1.200 and a marker is set to the maximum level. Before the maximum level can be determined, the sweep must be terminated. Synchronization to the end of the sweep is made by triggering a service request at the end of the sweep with command "\*OPC" (operation complete). The control program waits for the SRQ with the function RSDLLWaitSrqr(). Then the maximum level is determined ("CALC:MARK:MAX") and the level is read out ("Y?"). Prior to readout a check is made with Serial Poll whether data are available (MAV bit in status register set).

```
#define MAX_RESP_LEN 100

short          ibsta, iberr;
unsigned long  ibcntl;
short         ud;
short         srq;
char          Maxlevel[MAX_RESP_LEN];
char          spr;

// Determining the handle for the instrument
ud = RSDLLibfind( "89.1.1.200", &ibsta, &iberr, &ibcntl );

// If instrument exists
if ( ud >= 0 ) {

    // Setting timeout for RSDLLWaitSrqr() to 10 seconds
    RSDLLibtmo( ud, 10, &ibsta, &iberr, &ibcntl );

    // Activate SRQ generation through event status register (ESR)
    // and enable ESB bit of SRE register
    RSDLLibwrt( ud, "*ESE 1;*SRE 32", &ibsta, &iberr, &ibcntl );

    // Set single sweep, trigger sweep
    // and generate SRQ at the end of the sweep with "*OPC"
    RSDLLibwrt( ud, "INIT:CONT off;INIT;*OPC", &ibsta, &iberr, &ibcntl );

    // Wait for SRQ (end of sweep)
    RSDLLWaitSrqr( ud, &srq, &ibsta, &iberr, &ibcntl );

    // If sweep is terminated
    if (srq) {

        // Set marker to first maximum and query level
        RSDLLibwrt( ud, "CALC:MARK:MAX;Y?", &ibsta, &iberr, &ibcntl );
        // Check if data are available (MAV bit in status register set)
        RSDLLibrsp( ud, &spr, &ibsta, &iberr, &ibcntl );
        if (spr & 0x10) {
            // then read out data
            RSDLLilrd( ud, Maxlevel, MAX_RESP_LEN, &ibsta, &iberr, &ibcntl );
        }
    }
    // end connection to instrument
    RSDLLibonl( ud, 0, &ibsta, &iberr, &ibcntl );
}

else {
    ; // Error - instrument not found
}
```

### 5.3.3 WinWord (Word Basic)

#### Programming Hints:

- Access to RSIB.DLL functions

The functions of DLL `RSIB.DLL` can be utilized from a WinWord macro (WinWord 2.0/6.0) with certain restrictions. The function declarations for WinWord are defined in the module `rsibwb.bas` and must be copied into the macro when a WinWord macro is created.

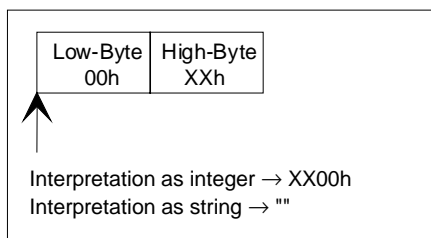
- Examples of macros in the file RSIB.DOT

The following programming examples show examples of the macros `SetStartFreq` and `GetStartFreq`, by which setting commands are sent to the instrument or instrument settings can be queried. The two macros are contained in the document pattern `rsib.dot`.

- Restrictions for WinWord macros

Parameters of the integer type can be defined only as value parameters in the declaration of the DLL functions with the statement `Declare`. However, the functions of DLL `RSIB.DLL` expect, for the variables `ibsta`, `iberr` and `ibcntl`, references via which the DLL can return values to the caller. To be able to specify variable references for these parameters, the functions are to be declared with the string parameter type.

By declaring the variables `ibsta`, `iberr` and `ibcntl` as strings, attempts to access undefined memory ranges are avoided. This method, however, makes the querying of parameters more difficult. For example, the error code is returned in the event of an error in the variable `iberr`. In this process, the two bytes of the integer value are copied into the string. WinWord however does not use the length information to handle strings but expects a zero at the end of the string. Therefore, for an integer value whose low-byte is equal to zero, the high-byte will not be queried since the low-byte with the zero is regarded as the end of the string.



Problem of variable declaration in WinWord

Querying error variables with values other than zero is nevertheless possible since error codes with the first character being less than 100h can be queried (`iberr=ASC(iberr$)`).

Querying the status variable, on the other hand, is not possible in the manner described since the low-byte of the status variable is always zero. The status byte is however returned also by the DLL function value so that an error check can be made.

For the use of the DLL `RSIB.DLL` from WinWord, it would be expedient to generate a DLL whose interface functions are adapted to the capabilities of WinWord.

**Programming examples:**

- Sending a setting command

The macro `SetStartFreq` is used as an example for setting the start frequency of an instrument with the IP address 89.1.1.200. The macro first reads the desired value from an input box. The value is then set with the DLL function `RSDLLibwrt()` via an SCPI command.

```
' Declaration from file "rsibwb.bas":
  Declare Function RSDLLibfind Lib "rsib.dll"(udName$, ibsta$, iberr$,
  ibcntl$) As Integer
  Declare Function RSDLLibwrt Lib "rsib.dll"(ud As Integer, Wrt$, ibsta$,
  iberr$, ibcntl$) As Integer

Sub MAIN

  Dim ud, status
  Dim cmd$, value$
  Dim ibsta$, iberr$, ibcntl$

  ' Query the frequency to be set:
  value$ = InputBox$("Enter the start frequency:", "R&S-IECBUS-
  Interface")

  ' Generate SCPI command
  cmd$ = "SENS:FREQ:STAR" + value$

  // Determine the handle for the instrument
  ud = RSDLLibfind("89.1.1.200", ibsta$, iberr$, ibcntl$)
  If(ud < 0) Then
    MsgBox "Error in function RSDLLibfind" + Chr$(10) + "Errorcode:
    " + Str$(Asc(iberr$)), 'error', 48
    Goto End
  End If

  ' Send command to instrument parser
  status = RSDLLibwrt(ud, cmd$, ibsta$, iberr$, ibcntl$)
  If(status <> 0) Then
    MsgBox "Error in function RSDLLibwrt" + Chr$(10) + "Errorcode:
    " + Str$(Asc(iberr$)), 'error', 48
    Goto End
  End If

End:

End Sub
```

- Querying instrument setups

The macro `GetStartFreq` is used as an example for querying the start frequency of the local instrument and reading it into the current document. First, the SCPI query command is sent to the instrument with the function `RSDLLibwrt()`, then the value is fetched with the function `RSDLLibrd()`. A string of sufficient length is not required for reading the output buffer since WinWord, like language C, processes the string length by using the terminating zero rather and not a prefixed length information as in Visual Basic.

## RSIB Interface

---

```
' Declaration from file "rsibwb.bas"
Declare Function RSDLLibfind Lib "rsib.dll"(udName$, ibsta$, iberr$,
ibcntl$) As Integer
Declare Function RSDLLibwrt Lib "rsib.dll"(ud As Integer, Wrt$, ibsta$,
iberr$, ibcntl$) As Integer
Declare Function RSDLLilrd Lib "rsib.dll"(ud As Integer, Rd$, cnt As
Long, ibsta$, iberr$, ibcntl$) As Integer

Sub MAIN

    Dim ud, status
    Dim buffer$, cmd$
    Dim ibsta$, iberr$, ibcntl$

    // Determine the handle for the instrument
    ud = RSDLLibfind("@local", ibsta$, iberr$, ibcntl$)
    If(ud < 0) Then
        MsgBox "Error in function ibfind" + Chr$(10) + "Errorcode: " +
Str$(Asc(iberr$)), 'error', 48
        Goto End
    End If

    ' Generate SCPI command for querying the start frequency
    cmd$ = "SENS:FREQ:STAR?"

    ' Send command to instrument parser
    status = RSDLLibwrt(ud, cmd$, ibsta$, iberr$, ibcntl$)
    If(status <> 0) Then
        MsgBox "Error in function ibwrt" + Chr$(10) + "Errorcode: " +
Str$(Asc(iberr$)), 'error', 48
        Goto End
    End If

    ' Read reply from parser
    status = RSDLLilrd(ud, buffer$, 10, ibsta$, iberr$, ibcntl$)
    If(status <> 0) Then
        MsgBox "Error in function ibrd" + Chr$(10) + "Errorcode: " +
Str$(Asc(iberr$)), 'error', 48
        Goto End
    End If

    ' Insert reply into document
    Insert buffer$ + "Hz"

End:
End Sub
```

### 5.3.4 Excel

The use of DLL functions in the macro language of Excel 4.0 has not been examined. Excel 5.0 uses VBA (Visual Basic for Applications) as macro language, which can create Excel macros same as Visual Basic programs.