

Rexroth Rho 4 PHG2000

1070072183
Edition 07

Software manual



Title Rexroth Rho 4
PHG2000

Type of Documentation Software Manual

Document Typecode DOK-RHO*4*-PHG2000****-PR07-EN-P

Purpose of Documentation The present manual informs about:

- handling and
- operation of the programming device PHG2000

Record of Revisions

Description	Release Date	Notes
DOK-RHO*4*-PHG2000****-PR06-EN-P	10.2003	Valid from VO07
DOK-RHO*4*-PHG2000****-PR07-EN-P	01.2005	Valid from VO08

Copyright © Bosch Rexroth AG, 1998 – 2005
Copying this document, giving it to others and the use or communication of the contents thereof without express authority, are forbidden. Offenders are liable for the payment of damages. All rights are reserved in the event of the grant of a patent or the registration of a utility model or design (DIN 34–1).

Validity The specified data is for product description purposes only and may not be deemed to be guaranteed unless expressly confirmed in the contract. All rights are reserved with respect to the content of this documentation and the availability of the product.

Published by Bosch Rexroth AG
Postfach 11 62
D-64701 Erbach
Berliner Straße 25
D-64711 Erbach
Tel.: +49 (0) 60 62/78-0
Fax: +49 (0) 60 62/78-4 28
Abt.: BRC/ESH (KW)

Overview of all manuals

Overview of all manuals

Manual	Contents
Connection conditions Rho 4.0	2 System overview
	3 Installation
	4 Electrical connection
	5 Interfaces
	6 LED display
	7 Maintenance and replacement
	8 Order numbers
System description Rho 4.0	2 System overview
	3 Structure of the rho4.0
	4 PCLrho4.0
	5 CAN-Bus peripheral unit
	6 SERCOS interface
	7 Software
	8 File management
Connection conditions Rho4.1, Rho 4.1/IPC300	2 System overview
	3 Security functions
	4 Installation
	5 Electrical connection
	6 Interfaces
	7 LED display
	8 Maintenance and replacement
	9 Software
	10 Order numbers
Connection conditions Rho 4.1/BT155, Rho 4.1/BT155T, Rho 4.1/BT205	2 System overview
	3 Security functions
	4 Installation
	5 Electrical Connections
	6 Interfaces
	7 Display and Operating Controls
	8 Maintenance and Replacemant
	9 Software
	10 Order numbers
System description Rho 4.1	2 Structure of the rho4.1
	3 PCL
	4 CAN-Bus peripheral unit
	5 SERCOS interface

Overview of all manuals

Manual	Contents
	6 Software
	7 File management
	8 Scope of the rho4.1 Software

Manual	Contents
Control functions	2 Survey of special functions
	3 Accurate position switching
	4 Setting the machine position
	5 Calling operating system functions
	6 Parameterization of the belt characteristic
	7 Selecting a point-file
	8 Mirroring
	9 Belt type
	10 System date and time
	11 System counter
	12 WC main range
	13 Setting the belt counter
	14 Recording of reference path
	15 Flying measurement (rho4.1 only)
	16 MOVE_FILE
	17 Setting the block preparation
	18 Exception-Handling
	19 Belt counter current value
	20 Automatic velocity adjustment for PTP movements
	21 Belt-synchronous working area belt kind 4
	22 Current belt speed
	23 Changing the belt simulation speed
	24 General functions
	25 Process-oriented functions
	26 BAPS3 keywords
	Machine parameters
3 Application of the machine parameters	
4 General system parameters	
5 Speeds	
6 Positions	
7 Kinematic parameters	

Overview of all manuals

Manual	Contents
	8 Measuring system parameters
	9 Belt parameters
	10 Drive parameters Servodyn-GC
	11 Drive parameter Servodyn-D
	12 Table of parameters
Manual	Contents
BAPS3 Programming manual	2 Program structure
	3 Constants
	4 Variables
	5 Program control
	6 Value assignments and combinations
	7 Functions
	8 Movement statement
	9 Write/read functions
	10 BAPS3 keywords
	BAPS3 Short description
3 Constants and variables	
4 Program structure	
5 Value assignments and combinations	
6 Standard functions	
7 Movements and speeds	
8 Belt synchronous	
9 Workspace limitation	
10 Write/read functions	
11 Special functions	
12 Library functions	
13 Fix files	
14 BAPS3 keywords	
Signal descriptions	
	3 Signal description of PCL inputs
	4 Signal description of PCL outputs
Status messages and warnings	2 rho4 status messages
	3 Warnings
	4 CANopen error codes
ROPS4/Online	2 General information
	3 Activation and functions of Online
	4 The function key box

Overview of all manuals

Manual	Contents
	5 Function key assignment
	6 The marker box
	7 File ROPS4WIN.ini
	8 Selection of a file
	9 TCP/IP settings for ROPS4
Manual	Contents
DLL library	2 Library functions
	3 Calling library functions in BAPS
	4 Block structure of the rho4.1
	5 Library server
	6 Application development
	7 rho4 library functions
	8 Variable access per DLL
	PHG2000
3 PHG2000 system variables	
4 Selection of PHG functions	
5 Info function of the PHG	
6 Controlling the PHG2000 output	
7 Define/Teach	
8 SRCAN functions	
9 File and User Memory Functions	
10 File list	
11 Process info	
12 Restoring the PGH display	
13 Variable assignment of PHG keys	
14 Select point file and point name	
15 BDT editor	
Connection conditions Rho 4.1/IPC 40.2	
	3 Security Functions
	4 Installation
	5 Eelectrical Connections
	6 Interface Ports & Connectors
	7 Display- and Operating Components
	8 Maintenance and Replacement
	9 Software
	10 Ordering Informations

Overview of all manuals

Manual	Contents
DDE-Server	2 Introduction
	3 Hardware and Software
	4 Operation
	5 Items of Server 4
	6 Scope of function

Overview of all manuals

Notes:

Contents

Contents

	Page
1	Safety Instructions 1–1
1.1	Intended use 1–1
1.2	Qualified personnel 1–2
1.3	Safety markings on products 1–3
1.4	Safety instructions in this manual 1–4
1.5	Safety instructions for the described product 1–5
1.6	Documentation, software release and trademarks 1–7
2	Hand-held programming unit PHG2000 2–1
2.1	Operating modes 2–3
2.1.1	PHG3 compatible mode 2–3
2.1.2	Transparent mode with a transfer rate of 9600 baud 2–4
2.1.3	Transparent mode with a transfer rate of 19200 baud 2–4
2.2	Keyboard layout 2–5
2.2.1	Mask-specific keyboard layout 2–5
2.3	BDT editor 2–6
2.4	Configuration and test 2–7
3	PHG2000 system variables 3–1
4	Selection of PHG functions 4–1
4.1	Direct selection 4–1
4.1.1	Direct return 4–1
4.1.2	Direct change of operating levels 4–2
4.2	Selection via interface 4–3
4.2.1	Functional description 4–3
4.2.2	Allocation file 4–3
4.2.3	Abort, return to basic level 4–4
4.2.4	Interface assignment 4–5
5	Info function of the PHG 5–1
5.1	Availability of the info function 5–1
5.2	Operation 5–2
5.3	Possible info texts 5–3
5.4	Output at PHG 5–6
6	Controlling the PHG2000 output 6–1
6.1	General remarks 6–1
6.1.1	Cursor positioning for different character sets 6–1
6.2	BAPS3 declaration part 6–3
6.3	Subroutine call in main program part 6–5
6.4	Outputs to the PHG2000 in subroutine technique 6–7

Contents

7	Define/Teach	7-1
7.1	Display layout	7-1
7.1.1	Display structure of Define	7-1
7.1.2	Display structure of Teach	7-2
7.2	Operation	7-4
7.2.1	Operation of Define	7-4
7.2.2	Operation of Teach	7-5
8	SRCAN functions	8-1
8.1	Electric type plate	8-1
8.1.1	Read serial number	8-1
8.1.2	Write serial number	8-1
8.1.3	Read machine parameter set	8-2
8.1.4	Write machine parameter set	8-2
8.1.5	Selection of the kinematic	8-2
8.2	Supervisor password	8-2
9	File and User Memory Functions	9-1
9.1	Changing the attributes of the files	9-1
9.2	User memory functions	9-1
10	File list	10-1
10.1	Creation of the objects in the BDT editor	10-2
10.1.1	Headline 'Dateiliste' on the PHG2000	10-2
10.1.2	List box for the file list	10-3
10.2	Display of the memory allocation	10-4
10.3	PHG2000 display	10-5
11	Process info	11-1
11.1	Mask in the BDT editor	11-1
11.2	Creation of the objects in the BDT editor	11-2
11.2.1	List box for the process list	11-3
11.2.2	Global process information	11-4
11.2.3	Information on selected process	11-4
11.3	PHG2000 display	11-5
11.4	Stopping processes	11-6
12	Restoring the PHG display	12-1
13	Variable assignment of PHG keys	13-1
14	Select point file and point name	14-1
14.1	Declaration	14-1

Contents

15	BDT editor	15-1
15.1	Principles	15-1
15.1.1	Pre-requisites	15-1
15.1.2	Installation	15-2
15.1.3	Configuration workstation	15-3
15.1.4	Concept	15-3
15.1.5	Short overview	15-6
15.2	Menus and dialog windows	15-9
15.2.1	Menu Projekt (Project)	15-9
15.2.2	<Projekt: Projektname> (Project: Project name) Menu Project ... 15-12	
15.2.3	Menu Mask	15-14
15.2.4	<Project: Projectname> Menu Maske (Mask)	15-16
15.2.5	Menu Objekt (Object)	15-26
15.2.6	Menu Ansicht	15-34
15.2.7	Menu Extras	15-34
15.2.8	Menu Hilfe (Help)	15-37
15.2.9	KontextMenu	15-38
15.3	BDT loader	15-39
15.3.1	Dialog window	15-39
15.4	Technical data	15-41
15.5	Software, operator devices, accessories	15-41
15.6	Create pictograms with Paintbrush	15-42
15.6.1	Screen size and attributes	15-42
15.6.2	Paintbrush as Pixel editor	15-42
A	Appendix	A-1
A.1	Abbreviations	A-1
A.2	Index	A-2
A.3	PHG menu tree	A-4

Contents

Notes:

Safety Instructions

1 Safety Instructions

Please read this manual before you startup the rho4.
Store this manual in a place to which all users have access at any time.

1.1 Intended use


This instruction manual presents a comprehensive set of instructions and information required for the standard operation of the described products. The described products are used for the purpose of operating with a robot control rho4.

The products described

- have been developed, manufactured, tested and documented in compliance with the safety standards. These products normally pose no danger to persons or property if they are used in accordance with the handling stipulations and safety notes prescribed for their configuration, mounting, and proper operation.
- comply with the requirements of
 - the EMC Directives (89/336/EEC, 93/68/EEC and 93/44/EEC)
 - the Low-Voltage Directive (73/23/EEC)
 - the harmonized standards EN 50081-2 and EN 50082-2
- are designed for operation in industrial environments, i.e.
 - no direct connection to public low-voltage power supply,
 - connection to the medium- or high-voltage system via a transformer.

The following applies for application within a personal residence, in business areas, on retail premises or in a small-industry setting:

- Installation in a control cabinet or housing with high shield attenuation.
- Cables that exit the screened area must be provided with filtering or screening measures.
- The user will be required to obtain a single operating license issued by the appropriate national authority or approval body. In Germany, this is the Federal Institute for Posts and Telecommunications, and/or its local branch offices.

 **This is a Class A device. In a residential area, this device may cause radio interference. In such case, the user may be required to introduce suitable countermeasures, and to bear the cost of the same.**

The faultless, safe functioning of the product requires proper transport, storage, erection and installation as well as careful operation.

Safety Instructions

1.2 Qualified personnel

The requirements as to qualified personnel depend on the qualification profiles described by ZVEI (central association of the electrical industry) and VDMA (association of German machine and plant builders) in:

Weiterbildung in der Automatisierungstechnik
edited by: ZVEI and VDMA
MaschinenbauVerlag
Postfach 71 08 64
D-60498 Frankfurt.

The present manual is designed for RC technicians. They need special knowledge on handling and programming robots.

Interventions in the hardware and software of our products, unless described otherwise in this manual, are reserved to specialized Rexroth personnel.

Tampering with the hardware or software, ignoring warning signs attached to the components, or non-compliance with the warning notes given in this manual may result in serious bodily injury or damage to property.

Only electrotechnicians as recognized under IEC 60947-1 (modified) who are familiar with the contents of this manual may install and service the products described.

Such personnel are

- those who, being well trained and experienced in their field and familiar with the relevant norms, are able to analyze the jobs being carried out and recognize any hazards which may have arisen.
- those who have acquired the same amount of expert knowledge through years of experience that would normally be acquired through formal technical training.

With regard to the foregoing, please note our comprehensive range of training courses. Please visit our website at <http://www.boschrexroth.com> for the latest information concerning training courses, teachware and training systems. Personal information is available from our Didactic Center Erbach,
Telephone: (+49) (0) 60 62 78-600.

Safety Instructions

1.3 Safety markings on products

Warning of dangerous electrical voltage!



Warning of danger caused by batteries!



Electrostatically sensitive components!



Warning of hazardous light emissions
(optical fiber cable emissions)!



Disconnect mains power before opening!



Lug for connecting PE conductor only!



Functional earthing or low-noise earth only!



Connection of shield conductor only

Safety Instructions

1.4 Safety instructions in this manual



DANGEROUS ELECTRICAL VOLTAGE

This symbol is used to warn of a **dangerous electrical voltage**. The failure to observe the instructions in this manual in whole or in part may result in **personal injury**.



DANGER

This symbol is used wherever insufficient or lacking compliance with instructions may result in **personal injury**.



CAUTION

This symbol is used wherever insufficient or lacking compliance with instructions may result in **damage to equipment or data files**.

☞ This symbol is used to draw the user's attention to special circumstances.

★ This symbol is used if user activities are required.

Safety Instructions

1.5 Safety instructions for the described product**DANGER**

Danger of life through inadequate EMERGENCY-STOP devices! EMERGENCY-STOP devices must be active and within reach in all system modes. Releasing an EMERGENCY-STOP device must not result in an uncontrolled restart of the system! First check the EMERGENCY-STOP circuit, then switch the system on!

**DANGER**

**Danger for persons and equipment!
Test every new program before starting up a system!**

**DANGER**

**Retrofits or modifications may adversely affect the safety of the products described!
The consequences may include severe injury, damage to equipment, or environmental hazards. Possible retrofits or modifications to the system using third-party equipment therefore have to be approved by Rexroth.**

**DANGER**

Do not look directly into the LEDs in the optical fiber connection. Due to their high output, this may result in eye injuries. When the inverter is switched on, do not look into the LED or the open end of a short connected lead.

**DANGEROUS ELECTRICAL VOLTAGE**

Unless described otherwise, maintenance works must be performed on inactive systems! The system must be protected against unauthorized or accidental reclosing.

Measuring or test activities on the live system are reserved to qualified electrical personnel!

Safety Instructions



CAUTION

Danger to the module!

Do not insert or remove the module while the controller is switched ON! This may destroy the module. Prior to inserting or removing the module, switch OFF or remove the power supply module of the controller, external power supply and signal voltage!



CAUTION

use only spare parts approved by Rexroth!



CAUTION

Danger to the module!

All ESD protection measures must be observed when using the module! Prevent electrostatic discharges!

The following protective measures must be observed for modules and components sensitive to electrostatic discharge (ESD)!

- Personnel responsible for storage, transport, and handling must have training in ESD protection.
- ESD-sensitive components must be stored and transported in the prescribed protective packaging.
- ESD-sensitive components may only be handled at special ESD-workplaces.
- Personnel, working surfaces, as well as all equipment and tools which may come into contact with ESD-sensitive components must have the same potential (e.g. by grounding).
- Wear an approved grounding bracelet. The grounding bracelet must be connected with the working surface through a cable with an integrated 1 MΩ resistor.
- ESD-sensitive components may by no means come into contact with chargeable objects, including most plastic materials.
- When ESD-sensitive components are installed in or removed from equipment, the equipment must be de-energized.

Safety Instructions

1.6 Documentation, software release and trademarks

Documentation

The present manual provides information on the handling and operation of the programming device PHG2000.

Overview of available documentation	Part no.	
	German	English
Rho 4.0 Connectivity Manual	1070 072 364	1070 072 365
Rho 4.0 System description	1070 072 366	1070 072 367
Rho 4.1/IPC 40.2 Connectivity Manual	R911308219	R911308220
Rho 4.1/BT155, Rho 4.1/BT155T, Rho 4.1/BT205 Connectivity manual	1070 072 362	1070 072 363
Rho 4.1, Rho 4.1/IPC300 Connectivity manual	1070 072 360	1070 072 361
Control panels BF2xxT/BF3xxT, connection	1070 073 814	1070 073 824
Rho 4.1 System description	1070 072 434	1070 072 185
ROPS4/Online	1070 072 423	1070 072 180
BAPS plus	1070 072 422	1070 072 187
BAPS3 Short description	1070 072 412	1070 072 177
BAPS3 Programming manual	1070 072 413	1070 072 178
Control functions	1070 072 420	1070 072 179
Signal descriptions	1070 072 415	1070 072 182
Status messages and warnings	1070 072 417	1070 072 181
Machine parameters	1070 072 414	1070 072 175
PHG2000	1070 072 421	1070 072 183
DDE-Server 4	1070 072 433	1070 072 184
DLL-Library	1070 072 418	1070 072 176
Rho 4 available documentation on CD ROM	1070 086 145	1070 086 145

 **In this manual the floppy disk drive always uses drive letter A:, and the hard disk drive always uses drive letter C:.**

Special keys or key combinations are shown enclosed in pointed brackets:

- Named keys: e.g., <Enter>, <PgUp>,
- Key combinations (pressed simultaneously): e.g., <Ctrl> + <PgUp>

Safety Instructions

Release

 **This manual refers to the following versions:**

Hardware version: rho4

Software release: ROPS4

Trademarks

All trademarks of software installed on Rexroth products upon delivery are the property of the respective manufacturer.

Upon delivery, all installed software is copyright-protected. The software may only be reproduced with the approval of Rexroth or in accordance with the license agreement of the respective manufacturer.

MS-DOS® and Windows™ are registered trademarks of Microsoft Corporation.

PROFIBUS® is a registered trademark of the PROFIBUS Nutzerorganisation e.V. (user organization).

MOBY® is a registered trademark of Siemens AG.

AS-I® is a registered trademark of AS-International Association.

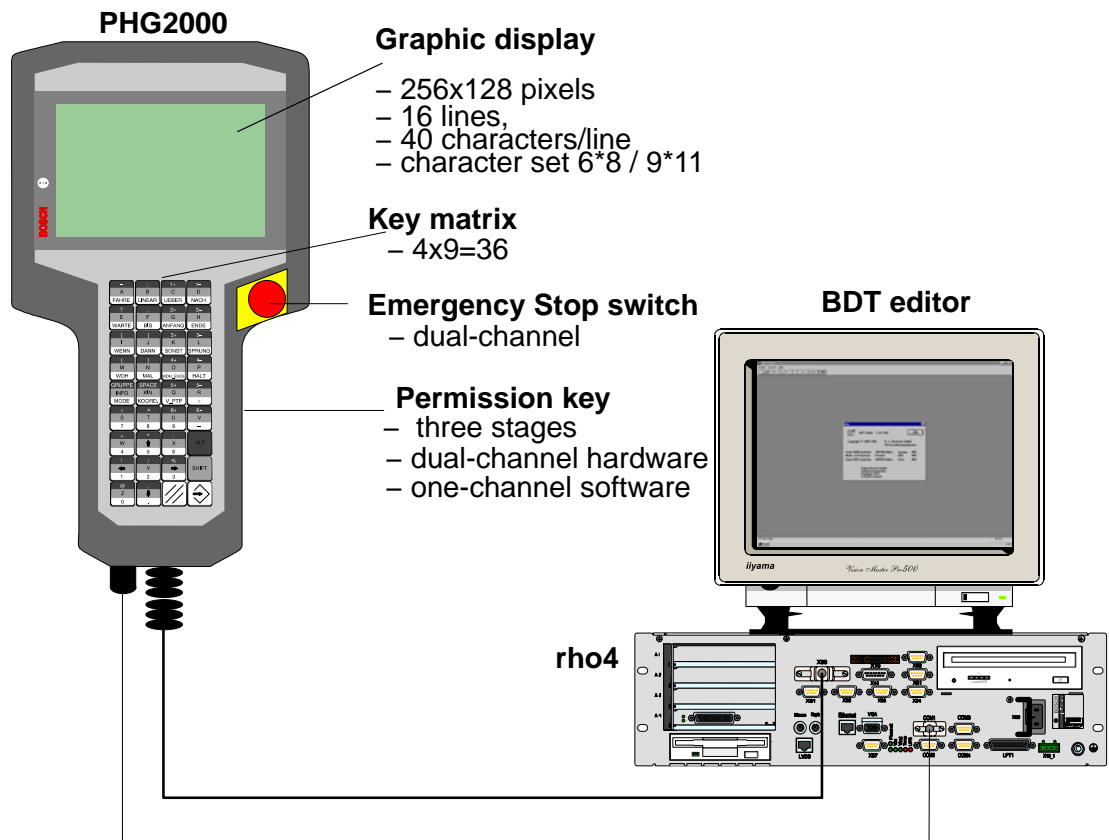
SERCOS interface™ is a registered trademark of Interessengemeinschaft SERCOS interface e.V. (Joint VDW/ZVEI Working Committee).

INTERBUS-S® is a registered trade mark of Phoenix Contact.

DeviceNet® is a registered trade mark (TM) of ODVA (Open DeviceNet Vendor Association, Inc.).

Hand-held programming unit PHG2000

2 Hand-held programming unit PHG2000



The PHG2000 is designed for the operation of the rho4, the manual moving of axes and for user-specific operating surfaces.

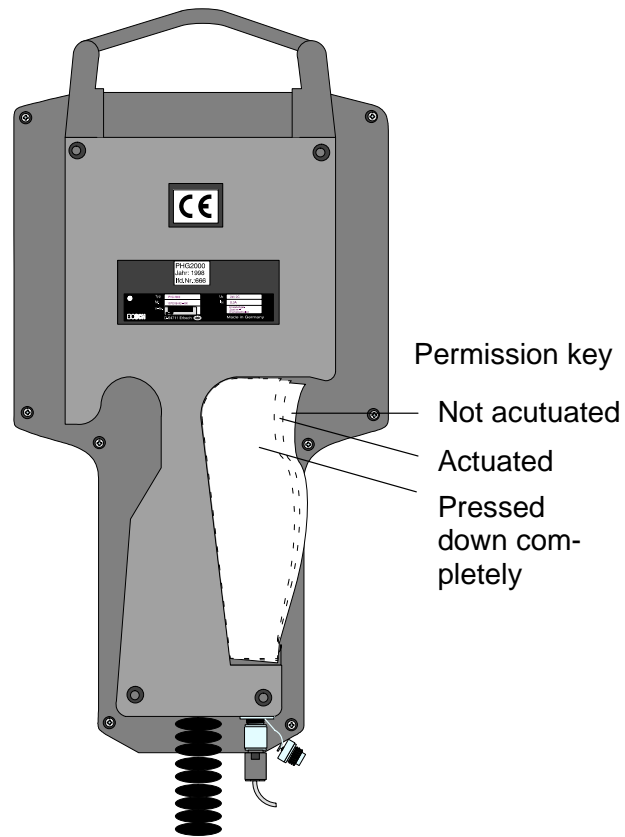
With the PHG2000 system variables can be accessed. Via the configuration tool BDT editor, a project with a menu structure of many branches can be set up which disposes of individual masks in each level. The PHG2000 is supplementary to the operating system with screen.

The Emergency Stop switch and the permission key of the PHG2000 are equipped with double contact sets. Both are to be wired in a dual-channel way in the Emergency Stop circuit. Additionally, the permission key is monitored by the software of the operating system in setting mode. It can furthermore be wired in the safety circuit e.g. for performance reduction.

Hand-held programming unit PHG2000

The permission key is designed in a way that three positions are possible

- not pressed (inactive)
- actuated (active)
- pressed down completely (inactive, panic position)



These three positions have been realized in a safe technique and offer additional safety for the operating personnel in case of danger as e.g. a possible clenching up of the human hand can lead to an Emergency Stop.

In the PHG3 compatible mode, the PHG2000 is behaving just like its ancestor PHG3, the digits, however, have the double size. In the enlarged mode, an intelligent dialog takes place between rho4 and PHG2000 due to the additional microprocessor. The communication between the partners is effected on the basis of objects, i.e. of specific commands, comprehensive macro procedures etc. The data exchange uses commands and transfer parameters.

The PHG2000 is equipped with a display fully supporting a graphic representation with 256 x 128 pixels. At the moment, two character sets with 6*8 and 9*11 pixels are available. For the PHG2000 individual display masks can be created and called.

Hand-held programming unit PHG2000

The user can build up a branched menu structure with masks and he can create his own operating surface adapted to his needs.

Into these masks, icons can be inserted for the operator prompting. The allocation of the icon key, K1 to K36, to the mask enables the selection of the individual menu levels.

In the masks, texts, text lists and variables are edited. Together with the description in the texts, variables from the control are displayed, controlled by the PHG2000. Variables are written into the control, effected by actuation of the key. Via the return value of the variable text list, a response is effected, dependent on the cursor position.

The assignment of the PHG keys can be adapted to client-specific needs by loading a changed keyboard layout file, Tasten.bnr in the rho4 or Layout.lay in the PHG2000, see chapter 13.

In a PHG2000 project, the assignment of the PHG keys can be adapted to client-specific needs through a layout-file (*.lay) in the PHG2000. A layout file is an ASCII file which can be changed with a normal editor. The file PHG2000.LAY is located in the directory of the BDT editor with a description of the structure of this file and the standard keyboard assignment. A layout file can be adopted with the BDT editor via the menu item [Extras][Import][Tastaturbelegung] in a PHG2000 project.

2.1 Operating modes

There are two operating modes for the PHG, the PHG3 compatible mode and the transparent mode. In the transparent mode it is gone back to the existing menu structure, or a user-oriented menu structure is defined by the operator himself. The operating mode is adjusted via machine parameter P35, PHG mode.

2.1.1 PHG3 compatible mode

The machine parameter P35 PHG mode is set to 1. The rho4 controls the data exchange

- Menu structure and operation via mode key as for PHG3
- Display as for PHG3, 4 lines of 20 characters each with a larger size
- Write PHG/read PHG as for PHG3
- Define/teach as for PHG3
- Redefine keys with the file Tasten.bnr

Hand-held programming unit PHG2000

2.1.2 Transparent mode with a transfer rate of 9600 baud

The machine parameter P35 PHG mode is set to 0 (presetting at delivery)

- Define (Mode 4.1) with new display and operation.
- Teach (Mode 4.2) with new display and operation.
- Redefine keys with the file Tasten.bnr.
- Write PHG/read PHG as for PHG3 via PHG interface or V24x with protocol 6, display: 4 lines à 20 characters (larger size)
- Write V24x/read V24x via V24x interface, setting V24x: interface 2 with protocol 2 (with echo) resp. protocol 5 (without echo). New functions for 256 x 128 pixels (in character set 6*8 or 9*11), see chapter 6.

Transparent mode with existing menu structure

The PHG keys and the PHG display are not disabled in the PCL. The rho4 controls the data exchange.

- Menu structure and operation via mode key as for PHG3.
- Display as for PHG3, 4 lines of 20 characters each with a larger size.

Transparent mode with user-oriented menu structure

The PHG display and possibly the PHG keys are disabled in the PCL. The PHG2000 controls the data exchange, the rho4 only supplies the data.

- User masks are created and loaded via the BDT editor. The PHG now has a menu structure of its own and individually defined key functions.

2.1.3 Transparent mode with a transfer rate of 19200 baud

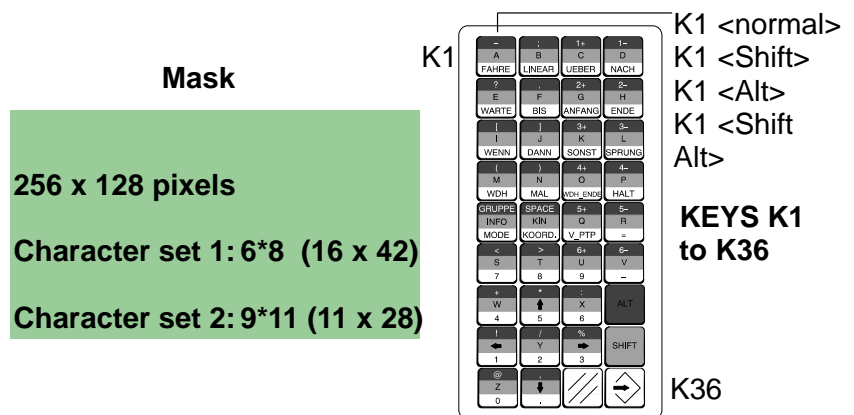
The machine parameter P35 PHG mode is set to 2. This enables a data transfer with 19200 baud. The further behaviour is like the one described under 2.1.2.

Hand-held programming unit PHG2000

2.2 Keyboard layout

The standard keyboard layout is valid as long as no new run-up of the control with the file Tasten.bnr has been carried out and so a new keyboard layout has been activated. It is also valid if in the transparent mode with user-oriented menu structure, the PHG keys and the PHG display have been disabled and consequently the mask-associated keyboard layout according to Layout.lay is activated.

2.2.1 Mask-specific keyboard layout



Each mask is strictly allocated an individual, freely definable keyboard layout. The 36 keys K1 to K36 can be assigned 4fold

- <Key>
- <Shift> + <Key>
- <Alt> + <Key>
- <Shift> + <Alt> + <Key>

It is possible to allocate to each key combination the selection of another mask or the writing of variables. In this way a user menu structure is created.

For user masks and variables max. 128 kBytes Flash-EPROM-Memory is available.

Hand-held programming unit PHG2000

2.3 BDT editor

With the tool BDT editor the PHG2000 masks and the linkage of masks via keys is defined.

The following mask elements are available:

- Constant texts
- Display of variables, e.g. AUTO_HAND_RCI, position axis 3, see manual of signal descriptions
- Display of constant texts with variable sections, e.g. selection 47 pieces
- Display of variables as texts, e.g. OFF, ON, black, brown, red
- Display/scrolling of lists with constant texts. Selection can be written onto rho4-variables (return value)
- Display/scrolling of lists with variable texts. Selection can be written onto rho4-variables (return value)
- Writing of a variable when calling a mask
- Writing on rho4 variables by pressing/releasing a key, e.g. F1 gripper=1, F5 gripper=0
- Editing of displayed rho4 variables
- Call of other masks by pressing a key
- Call of other masks by pressing a key dependent on variables, e.g. key switch

Hand-held programming unit PHG2000

2.4 Configuration and test

Only possible if there is no active communication with the rho4. Therefore, run down rho4, pull off mains plug resp. 24-V-supply at the rho4, so that only the 24-V-supply of the PHG exists.

With the key combination



the contrast setting of the PHG2000 is called.



- With the displayed key assignment Plus (K1) and Minus (K2) the contrast value can be set between 0 and 127.
- With the key <?> (K3) the version statuses of all PHG drivers are displayed.
- The key combination <Shift> + <?> calls the test mode in which the keys of the PHG2000 can be checked.

Pressed key(s)	Display
<Key>	1
<Shift> + <Key>	2
<Alt> + <Key>	3
<Shift> + <Alt> + <Key>	4

Hand-held programming unit PHG2000

Notes:

PHG2000 system variables

3 PHG2000 system variables

The data exchange between rho4 and PHG2000 is carried out via system variables. These system variables are defined in the BDT editor. They can access the rho4 variables by reading and in several cases by writing (not onto the user variables defined in the program).

When defining the system variables, the following definitions must be determined

Variable name	max. 30 characters
Data type	Bit Byte byte (unsigned), Word word (unsigned), Double double (unsigned), String string (variable text list), Real (IEEE format)
Variable number	See tables on next pages.
Array	If the variable number does not refer definitely to one variable only, but to a variable field, Array must be selected. With that, an entry into the input field Index is automatically required.
Variable index	Serial number, only with Array, see tables. Coded index, for a multidimensional field.

The index of the PCL signals is derived from the standard signal assignment of the PCL interface, see manual of signal description rho4.

The screenshot shows a dialog box titled "Variable bearbeiten" with the following fields and options:

- Name: RCA
- Datentyp: Bit
- Pollingzeit: 0 ms
- Nummer: 33037
- Array Index: 392
- Zugriff:
 - Read
 - Write
 - Read/Write

Buttons: OK, Abbrechen, Hilfe

 The markers used for the rho3 are no longer used for the rho4 with PCL.

PHG2000 system variables

Signals

Meaning		Basic type	Index	Code	R/W
Signals				33024	
PLC inputs	bit	Bit	0 to 2047	33025	R
PLC outputs (RCI)	bit	Bit	0 to 2047	33026	R
PLC inputs	byte	Byte	0 to 255	33028	R
PLC outputs (RCI)	byte	Byte	0 to 255	33029	R
PLC inputs	word	Word	0 to 254	33031	R
PLC outputs (RCI)	word	Word	0 to 254	33032	R
PLC inputs	double	Double	0 to 252	33034	R
PLC outputs (RCI)	double	Double	0 to 252	33035	R
RC outputs	bit	Bit	128 to 1407	33037	R/W
RC outputs	byte	Byte	16 to 175	33038	R/W
RC outputs	word	Word	16 to 174	33039	R/W
RC outputs	double	Double	16 to 172	33040	R/W
PHG2000 outputs	bit	Bit	64 to 127	33041	R/W
PHG2000 outputs	byte	Byte	8 to 15	33042	R/W
PHG2000 outputs	word	Word	8 to 14	33043	R/W
PHG2000 outputs	double	Double	8 to 12	33044	R/W
Number of analog inputs		Byte	–	33057	R
Number of analog outputs		Byte	–	33058	R
Analog inputs		Real	1 to 32	33059	R
Analog outputs		Real	1 to 32	33060	R

PHG2000 system variables

Positions

Meaning	Basic type	Index	Code	R/W
Positions			33280	
JC position set	Real	1 to 24	33281	R
JC position actual	Real	1 to 24	33282	R
WC position set	Real	1 to 24	33283	R
Reserve			33284	
OC position	Real	1 to 24	33285	R
BC position	Real	1 to 24	33286	R
SW limit switch, positive WC	Real	1 to 24	33287	R
SW limit switch, negative WC	Real	1 to 24	33288	R
SW limit switch, positive JC	Real	1 to 24	33289	R
SW limit switch, negative JC	Real	1 to 24	33290	R
Axis names	String	1 to 24	33291	R
Coordinate names	String	1 to 24	33292	R
Lag	Real	1 to 24	33293	R
Inpos ('*' or ' ')	String	1 to 24	33294	R
End point	Real	1 to 24	33296	R
Number of axes	Byte	–	33297	R
Number of belts	Byte	–	33298	R
Belt	Real	1 to 16	33299	R
Belt name	String	1 to 16	33300	R

All position variables can be accessed via

- simple index: serial axis number (entire system)
- coded index: kinematic number/axis number of the kinematic

For the coded index, the value to be entered is composed of the identification of the coded index in the uppermost bit (15), the kinematic number and the number of the component (axis).

1	Kinematic 1 to 127	Component 1 to 255
---	--------------------	--------------------

Calculation: 32768 + kinematic number * 256 + component number

PHG2000 system variables

Further kinematic info

Meaning	Basic type	Index	Code	R/W
Further kinematic info			33536	
Number of axes dependent on kinematic	Byte	1 to 16	33537	R
Number of belts dependent on kinematic	Byte	1 to 16	33538	R
Tool name	String	1 to 16	33539	R
Tool	Real	coded	33540	R
WC system	Real	coded	33541	R
Kinematic name	String	1 to 16	33542	R

A, V, D

Meaning	Basic type	Index	Code	R/W
A, V, D			33792	
VFACTOR global	Real	–	33793	R/W
AFACTOR global	Real	–	33794	R/W
DFACTOR global	Real	–	33795	R/W
VFACTOR global limit min	Real	–	33796	R
VFACTOR global limit max	Real	–	33797	R
AFACTOR global limit min	Real	–	33798	R
AFACTOR global limit max	Real	–	33799	R
DFACTOR global limit min	Real	–	33800	R
DFACTOR global limit max	Real	–	33801	R
VFACTOR kin	Real	1 to 16	33802	R/W
AFACTOR kin	Real	1 to 16	33803	R/W
DFACTOR kin	Real	1 to 16	33804	R/W
VFACTOR kin limit min	Real	1 to 16	33805	R
VFACTOR kin limit max	Real	1 to 16	33806	R
AFACTOR kin limit min	Real	1 to 16	33807	R
AFACTOR kin limit max	Real	1 to 16	33808	R
DFACTOR kin limit min	Real	1 to 16	33809	R
DFACTOR kin limit max	Real	1 to 16	33810	R

PHG2000 system variables

System info

Meaning	Basic type	Index	Code	R/W
System info			34048	
Clock time [ms]	Word	–	34049	R
Time	String	–	34050	R
Date	String	–	34051	R
Date/Time	String	–	34052	R
Version P1	String	–	34053	R
Version P2 (Reserve)	String	–	34054	R
Version SK1 (Reserve)	String	–	34055	R
Version SK2 (Reserve)	String	–	34056	R
Version SK3 (Reserve)	String	–	34057	R
Version 8085	String	–	34058	R
PLC program name (Reserve)	String	–	34059	R
Number of warnings	Word	–	34060	R
Number of errors	Word	–	34061	R
Warning ASCII axis + code	String	1 to 32	34062	R
Warning ASCII + axes	String	1 to 32	34063	R
Warning ASCII	String	1 to 32	34064	R
Warning axis cod.	Word	1 to 32	34065	R
Warning code	Word	1 to 32	34066	R
Error ASCII axis + code	String	1 to 32	34067	R
Error ASCII + axes	String	1 to 32	34068	R
Error ASCII	String	1 to 32	34069	R
Error axis cod.	Word	1 to 32	34070	R
Error code	Word	1 to 32	34071	R

PHG2000 system variables

Process info

Meaning	Basic type	Index	Code	R/W
Process info			34304	
Number of normal processes	Word	–	34305	R
Number of permanent processes	Word	–	34306	R
Number of sub-processes	Word	–	34307	R
Number of processes with errors	Word	–	34308	R
Process list (list box)	String	1 to 100	34309	R
Selected process	Word	–	34310	R/W
Process type	String	–	34311	R
Process priority	Word	–	34312	R
Process status	String	–	34313	R
External program	String	–	34314	R
Program level	Byte	–	34315	R
Process name	String	–	34316	R
qll-line	Word	–	34317	R
qll-INCLUDE-line	Word	–	34318	R
Kinematic	Word	–	34319	R
Process error code	Word	–	34320	R
Process error text	String	–	34321	R
Stop process	Byte	–	34322	W


File info

Meaning	Basic type	Index	Code	R/W
File info			34560	
File lists – name/length/date	String	1 to 1000	34561	R
File lists – name/length	String	1 to 1000	34562	R
File lists – name/date	String	1 to 1000	34563	R
File lists – name	String	1 to 1000	34564	R
Cursor index	Double	–	34565	R/W
Control variable	Double	–	34566	R/W
Storage utilization	Double	–	34567	R
Available storage	Double	–	34568	R
Occupied storage	Double	–	34569	R
Free storage	Double	–	34570	R

PHG2000 system variables

Variable text lists

Meaning	Basic type	Index	Code	R/W
Variable text lists			34816	
VTL000	String	1 to 999	34816	R
VTL001	String	1 to 999	34817	R
VTL002	String	1 to 999	34818	R
. 256 dat-files	one line of the file	index corresp. to line-number		
VTL253	String	1 to 999	35069	R
VTL254	String	1 to 999	35070	R
VTL255	String	1 to 999	35071	R

 **If projects that are already available for the rho3 are to be further used with the PHG2000 on the rho4, you are able to adapt the changed indices with the BDT editor.**

Example

The signal PLC output (RCI) Emergency Stop, not has got the index 16 in the rho3. In the rho4 the signal has got the index 128.

PHG2000 system variables

Notes:

Selection of PHG functions

4 Selection of PHG functions

4.1 Direct selection

The menu tree, see appendix 3, contains all functions which can be selected with the PHG2000. The relationship between the individual menu points is shown in a structured way. So the selection of the individual menu points is facilitated essentially.

The numbers in the boxes are the mode numbers. With the help of these mode numbers, the desired function can be selected directly on the PHG2000.

Instead of scrolling with the cursor keys through the PHG display until e.g. the menu point 'Set machine parameters' appears, it is possible to select the menu point directly by entering 'Mode 7.8.2'.



Please note that the individual digits are separated by a point.

By acknowledging with



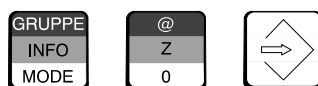
the jump to 7.8.2 is activated.

In order to get e.g. from mode 7.8 'mach. parameter' into the sub-mode 7.8.6.4 'Write machine par.' of the same hierarchical branch, you can proceed in an analog way. In this case, you have to enter 'Mode 6.4'.



4.1.1 Direct return

With the input of 'Mode 0'



it is possible to return from any sub-mode into the basic level. From there, it is possible to go again into each sub-menu branch.

Selection of PHG functions

4.1.2 Direct change of operating levels

The change of operating levels, e.g. from 'Diagnosis' to 'Help functions' is possible by entering a leading '0' before the desired function.

Example:

By input of 'Mode 0.11.4' in any other level, the sub-mode 'VFACTOR' is selected via the basic level.

A change across the hierarchical branches of the menu tree can, in general, only be carried out via the basic level.

The check for an admissible sub-mode is only carried out when all levels placed before have been passed through. An exception is the check of admissible ASCII characters ('0' to '9' and '.'). In case of error, the direct selection stops in the level in which an invalid sub-mode has been selected, i.e. the direct selection is interrupted in mode 9.4, when 'Mode 9.4.10' is entered, as mode 9.4.10 does not exist. An error message is not given out in this case. The user can then select one of the valid sub-modes, here 1, 2 or 3.

If, on the way to a sub-mode, the selection of a file name is required, the direct selection is running including this file selection.

The selection of individual functions of the menu tree can also be carried out by entering the corresponding mode. Returning from one level to the level before is also possible. To do so, press



Selection of PHG functions

4.2 Selection via interface

The rho4 offers the possibility to select PHG functions via interface signals. So the selection can be carried out from a user process. The user has furthermore a tool for creating an operating surface of his own.

4.2.1 Functional description

A coded selection of the PHG functions via interface is optional. It can only be used if the corresponding option bit (Option 41) has been set in machine parameter P28.

Selection

Due to a preset interface structure (RC inputs/outputs), only a coded selection is possible. The coding is carried out hex with 8 bits. So it is possible to select 256 different functions.

The activation of the coded PHG functions is started with the positive edge of the RC input 240, rho4 interface addr. A30.0, 'Strobe, coded PHG function selection'.

Code=0 with trailing edge of the strobe signal serves the selection of the basic level.

A coded selection can only be carried out if the basic level of the PHG operating surface is active. Via the RC output 152, rho4 interface addr. E19.0, 'Basic PHG level is active', this status is indicated with '1'.

4.2.2 Allocation file

The allocation of the code number to the desired PHG function is done in a file.

The file name for the German version is 'PHGKODE.dat' and for the English version 'PHGCODE.dat' correspondingly.

Layout of the allocation file

In the allocation file, a function of the PHG menu tree is assigned to each code number. The notation of the PHG function corresponds to the direct selection of PHG functions.

Example:

Selection of PHG functions

Layout of the file PHGCODE.DAT


Code No.	=	Allocated PHG-mode	Comment
00	=	0	;Code=0 with negative edge effects ;return to basic level
1B	=	3.1.4	;Compile
28	=	2	;Manual
3F	=	4.2	;Teach
42	=	7.1	;Axis positions
A6	=	7.3.2	;Process statuses
FF	=	11.1	;Basic position at the PHG

With code=1B the PHG function 'compile BAPS program' is to be activated. With the traditional input mode 3, mode 1 and mode 4 would have to be entered, with direct selection correspondingly mode 3.1.4.

So the allocation file '1B=3.1.4' is to be entered.

Spaces, null lines and comments are admitted in the allocation file. Comments are introduced by a semicolon ';' and are valid for the rest of the corresponding line.

The code No. is to be written as a two-digit hex-number (00 to FF).

 **Not all codes must be occupied. It is also permitted to allocate one PHG function to several different codes.**

4.2.3 Abort, return to basic level

The abort of a PHG function resp. the return to the basic level can be carried out with Code=00 and negative edge of the strobe signal.

This return can be carried out at any time, i.e. also with active dynamic displays (e.g. lag) or during an input (e.g. VFACTOR).

For the abort, it is not important which PHG function is allocated to the code '00' in PHGCODE.dat. So the code '00' can also be used for the selection. It has to be taken into account, however, that the return to the basic level is carried out immediately when taking away the selection signal (with Code=00).

Selection of PHG functions

4.2.4 Interface assignment

For the RC inputs, the following signals are available for the coded selection of PHG functions per interface.

 **The selection via interface is not possible when the basis level of the PHG operator desktop is not active.**


RC inputs

Ser. No.	Symbol name PLC	rho4 inter- face address	Signal description
240	COD_PHG_RCI	O30.0	Strobe, coded PHG function selection
248	PHGFCT_1_RCI	O31.0	Coded PHG function selection, valence 1
249	PHGFcT_2_RCI	O31.1	Coded PHG function selection, valence 2
250	PHGFcT_4_RCI	O31.2	Coded PHG function selection, valence 4
251	PHGFcT_8_RCI	O31.3	Coded PHG function selection, valence 8
252	PHGFcT16_RCI	O31.4	Coded PHG function selection, valence 16
253	PHGFCT32_RCI	O31.5	Coded PHG function selection, valence 32
254	PHGFCT64_RCI	O31.6	Coded PHG function selection, valence 64
255	PHGFCT128RCI	O31.7	Coded PHG function selection, valence 128

RC outputs

Ser. No.	Symbol name PLC	rho4 inter- face address	Signal description
152	BASE_AC_RCO	I19.0	Basic-PHG level is active

- Within the PHG function designation in short notation, no spaces are admitted (e.g. 7. 3 .2 is not permitted).
- The character directly behind the last mode No. of a line can only be a space ' ', a semicolon ';' or <CR><LF>. As introduction for a comment a semicolon ';' must be used. The rest of the line is then overread.

 **If there is no assignment for a code or if the allocation file contains a mistake, the message 'error PHGCODE.dat' is issued. Errors in the allocation file are only detected during the running time, i.e. when selecting a PHG function. Additionally, a corresponding error code is issued at the interface.**

Selection of PHG functions

Notes:

Info function of the PHG

5 Info function of the PHG

The info function is designed for supplying operator instructions for the operation of the rho4 with the PHG if, in a menu branch, the selected function is not carried out.

Let us suppose that the user wants to approach reference points. He has selected mode 1, reference points, and tries to move the axes. The control does not react because e.g. the controller release is missing or the referencing via machine parameter P 402 is switched off.

When selecting the info function, the operator is informed by display why the execution of the selected function is not carried out. If no error exists, the display 'No Info available' appears.

If the info function is left again, the operation can be continued at the change over position.


5.1 Availability of the info function

The info function can be called in each PHG menu branch.

For the following PHG modes, information is displayed at the moment

- Mode 1 Reference points
- Mode 2 Manual
- Mode 3 Program BAPS
- Mode 3.1 Programming BAPS
- Mode 4 Define/Teach
- Mode 4.1 Define
- Mode 4.2 Teach
- Mode 7 Diagnosis
- Mode 7.1 Axis displays
- Mode 7.8 Mach. parameters
- Mode 9 Device file I/O
- Mode 9.1 Default
- Mode 10 Select progr.
- Mode 10.1 Select processes
- Mode 10.2 Stop processes
- Mode 11 Help functions

The remaining modes indicate 'No info texts' when the info function is called.

 **Compiler errors, e.g. which are already signalled by the compiler, cannot be called via the info function. There is also no info function in case of current system messages.**

Info function of the PHG

5.2 Operation

The selection of the info function is carried out by actuating



Leaving the info function or display of the status message in case of a current error with



Scrolling back is carried out with



Scrolling forward with



Leaving the info function with



Info function of the PHG

5.3 Possible info texts

MANUAL and AUTOMATIC

PHG text	Comment
Manual disabled	Manually disabled
Dis.M4.1,9.1,11.4..6	Disable mode: 4.1, 9.1, 11.4, 11.5, 11.6
Progr. BAPS disabled	Progr. BAPS is disabled
Device/file I/O disab	Device file I/O is disabled
Progr.start disabled	Program start is disabled
Restart of RC disabl	RC re-start via PHG is disabled
PHG keys disabled	Disable PHG keys for RC
Del/mod files disabl	Delete and modify program is disabled
Def/teach in disable	Define/Teach is disabled
Reference points disab	Referencing is disabled
Disable mess.ref.pt.	Disable message of referencing
Disable online funct	Disable Online functions
Disable coupling PG	Disable coupling to programming unit
Disable printer	Disable printer output
READ/WR: SER1 disab	Disable read/write ser. 1
READ/WR: SER2 disab	Disable read/write ser. 2
READ/WR: SER3 disab	Disable read/write ser. 3
READ/WR: SER4 disab	Disable read/write ser. 4

Mode 1 Reference points

PHG text	Comment
no reference points!	Reference points have not yet been approached
No referenc. (P402=0)	Referencing inactive (MP 402 = 0)
emerg.op. without RC	Emergency operation Emergency Stop (via error display)
not INPOS of all axes	INPOS is missing
FEED ALLOW not avail.	Advance enable is missing
CONTRL.ALLOW a. axes	Controller enable is missing
DRIVE ON not avail.	Drive-on is missing
TRAVEL ALLOW n.avail	Move enable is missing
Kinematics:1;2;3;...	Kinematics: 1; 2; 3;...

Info function of the PHG

Mode 2 Manual

PHG text	Comment
Kinem.: ROBI_1	Selected kinematic: ROBI_1 ☞ The selected kinematic is displayed here, because it is not always clear from the axis display which kinematic is currently selected.
no reference points!	Reference points have not been approached yet
Coord. not selected	Coordinate system is not yet selected
emerg.op. without RC	Emergency operation Emergency Stop (via error display)
not INPOS of all axes	Axes are not in position
FEED ALLOW not avail.	Advance enable is missing
CONTRL. ALLOW not avail.	Controller enable is missing
DRIVE ON not avail.	Drive on is missing
TRAVEL ALLOW n.avail.	Move enable is missing
Kinematics:1;2;3;...	Kinematics: 1; 2; 3;...

Mode 3 Program BAPS

PHG text	Comment
Progr. BAPS disabled	Progr. BAPS disabled

Mode 3.1 Programming BAPS

PHG text	Comment
Progr. BAPS disabled	Progr. BAPS disabled

Mode 4 Define/Teach

PHG text	Comment
Def/Teach in disable	Define/teach disabled
	(no PKT file available -> is already displayed)

Info function of the PHG

Mode 4.1 Define

Mode 4.2 Teach

PHG text	Comment
no referenc. (P402=0)	Reference points have not yet been approached
Coord. not selected	Coordinate system has not yet been selected
emerg.op. without RC	Emergency operation Emergency Stop (via error display)
not INPOS of all axes	INPOS is missing
FEED ALLOW not avail.	Advance enable is missing
CONTRL. ALLOW a. axes	Controller enable is missing
DRIVE ON not avail.	Drive-on is missing
TRAVEL ALLOW n.avail	Move enable is missing
Kinematics:1;2;3;...	Kinematic: 1; 2; 3;...

Mode 7 Diagnosis

Mode 7.1 Axis displays

PHG text	Comment
not INPOS of all axes	INPOS is missing Moving range limit is reached (via error display)

Mode 7.3 System statuses

Mode 7.3.2 Process statuses

PHG text	Comment
error in USER-TASK	Process error
error(s) or warning(s)	Status messages/warnings (analog Mode 7.2)

Mode 7.8 Mach. parameters

Mode 7.8.2 Set mach. parameters

PHG text	Comment
Del/mod files disabl	Change of machine parameters disabled

Info function of the PHG

Mode 9 Device file I/O

Mode 9.1 Default

PHG text	Comment
Interface: n.avail.	Interface _ physically not available
Interface: u.double	Double assignment of the interface e.g.: 1

Mode 10 Select progr.

Mode 10.1 Select processes

PHG text	Comment
Progr.start disabled	Program start is disabled

Mode 10.2 Stop processes

Mode 11 Help functions


PHG text	Comment
Restart of RC disabl	RC re-start via PHG is disabled
V/A/Dfactor disabled	VFACTOR/AFACTOR/DFACTOR is disabled

5.4 Output at PHG

There is a priority of the info text display on the PHG compared to other texts (they overwrite other displays).

The priority of the outputs is as follows

Priority	
1	Messages of the test system (only with active test system)
2	WRITE PHG of BAPS
3	Coded text output
4	Info texts
5	Standard operation at the PHG

 **As the info function has a lower priority than the BAPS-READ-Operation, the BAPS-READ-operation must at first be terminated before the info function can be activated.**

Controlling the PHG2000 output

6 Controlling the PHG2000 output

6.1 General remarks

- The subroutine calls listed in section 6.3 Subroutine call in the main program part, do not represent a complete or logical program, but only explain the function of transfer parameters and calls of subroutines.
- The interface, e.g. X35, by which the PHG2000 is operated, must e.g. be adjusted with MODE 9.1.1.x, to interface 2 V24_2, protocol 2 data with echo resp. protocol 5 data without echo. The plug remains on X35 (rho4.1).
- The file phgausg.exe in the directory rho of the mailbox phone 06062/7217 can be used as an example program.
- Positionings are normally indicated in pixels. The size of the display is 256 x 128 pixels. The origin of the coordinates ($x = 0$, $y = 0$) is at the left on the top.
- The selection of the character attributes and the character set must only be carried out once, if one can be sure that these are not changed by other processes or system outputs in the meantime.

6.1.1 Cursor positioning for different character sets

Example 1

Character set 6*8, 6 pixels wide and 8 high

For the 7th character of the 3rd line, the pixel position is $x = 36$, $y = 16$

The following formula has to be applied

$$\begin{aligned} \text{x-pixel position} &= (\text{column} - 1) && * \text{horizontal character size} \\ 36 &= (7 - 1) && * 6 \end{aligned}$$

$$\begin{aligned} \text{y-pixel position} &= (\text{line} - 1) && * \text{vertical character size} \\ 16 &= (3 - 1) && * 8 \end{aligned}$$

Controlling the PHG2000 output

Example 2

Character set 9*11, 9 pixels wide and 11 high

For the 5th character in the 7th line, the pixel position is $x = 36$, $y = 66$

The following formula has to be applied

$$\begin{aligned} \text{x-pixel position} &= (\text{column} - 1) && * \text{horizontal character size} \\ 36 &= (4 - 1) && * 9 \end{aligned}$$

$$\begin{aligned} \text{y-pixel position} &= (\text{line} - 1) && * \text{vertical character size} \\ 66 &= (7 - 1) && * 11 \end{aligned}$$

Controlling the PHG2000 output

6.2 BAPS3 declaration part

PROGRAM TRANSPAR

; Begin of declaration part

; Commands of the transparent mode

; Low byte = Command, high byte = Length of command

```
CONST:                                ;low + high

    CD_CLS          = 1 + (0 * 256),
    CD_GOTOXY       = 2 + (4 * 256),
    CD_ATTRIB       = 3 + (1 * 256),
    CD_TEXT         = 5,                ; Length is set to 80 characters
    CD_REFRESH      = 7 + (4 * 256),
    CD_END          = 8 + (0 * 256),
    CD_LINE         = 12 + (8 * 256),
    CD_N_MASK       = 15 + (2 * 256),
    CD_CHARSET      = 18 + (1 * 256),
    CD_ZEI_REL      = 22 + (2 * 256)
```

; Display attributes

```
CONST:

    NORMAL          = 0,                ; Characters are displayed normal
    BLINK           = 1,                ; Characters are blinking
    INVERS          = 2,                ; Characters are displayed inverted
    DOPPEL_BREIT    = 4,                ; Characters have the double width
    DOPPEL_HOCH     = 8,                ; Characters have the double height
```

; Character sets

```
CONST:

    CS_6_TIMES_8    = 0,                ; max. 42x16 characters can be displayed
    CS_9_TIMES_11   = 1,                ; max. 28x11 characters can be displayed
```

Controlling the PHG2000 output

```
; Mask selection
```

```
Constant:
```

```
    BEL_MASK      = 65535,           ; 65535 = always display PHG outputs  
    START_MASK    = 1,              ; Start mask  
    MK_IST_POS     = 2               ; Mask for actual position display of axes  
                                     ; in JC
```

```
; System variables
```

```
CONST:
```

```
    VTL000_NR     = 34816           ; Basic number of the VTL files
```

```
; -----
```

```
FILE: PHG2000
```

```
; End of declaration part
```

Controlling the PHG2000 output

6.3 Subroutine call in main program part

; Begin of main program part

BEGIN

```

ASSIGN PHG2000, 'V24_4.      '      ;The interface 'V24_4' is allocated
                                     ;to the file variable 'PHG200'.
                                     ;3 blanks stand for the file extension

JUMP example                        ;The following subroutine calls do not
                                     ;represent a complete or reasonable program
                                     ;but explain only the function of operation
                                     ;parameter and subroutines calls
                                     ;Therefore "JUMP example"

SC_FOR_MASK(BEL_MASK)              ;Each PHG2000 output must start
                                     ;with the mask number for which
                                     ;this output is determined. If another
                                     ;mask number is currently displayed, the
                                     ;output is rejected. If 65535 is transferred
                                     ;as value, the output is always carried out
                                     ;independent from the currently active mask
                                     ;The output must be terminated by
                                     ;SC_END

SC_NEW_MASK(START_MASK)            ;Select start mask

SC_NEW_MASK(MK_IST_POS)            ;Select mask of actual position display
                                     ;of the axes in JC

SC_CLS                             ;The PHG display is completely deleted

SC_LINE(0,0,255,127)               ;A line is drawn from the left top
                                     ;(x=0, y=0) to the right bottom
                                     ;(x=255, y=127)

SC_CHARSET(CS_9_TIMES_11)          ;Selection of the character size. Character
                                     ;matrix is here 9 x 11 pixels.

SC_ATTRIB(NORMAL)                  ;Set attribute for normal display

SC_ATTRIB(BLINK+INVERS)            ;Set attribute for blinking and inverted
                                     ;display. Constants can be added.

SC_GOTOXY(30,40)                   ;Position cursor to x=30, y=40 pixels

SC_TEXT('HALLO WORLD')             ;Character sequence 'HALLO WORLD' on PHG
                                     ;display

SC_GOTOXY(30,60)                   ;Position cursor to x=30, y=60 pixels

SC_DEC(-1234.45)                   ;Display the decimal number -1234.45
                                     ;on PHG

SC_GOTOXY(30,80)                   ;Position cursor to x=30, y=80 pixels

```

Controlling the PHG2000 output

```

SC_INTEGER(1234,8)           ;Display integer 1234. The second transfer
                             ;parameter (8) indicates the maximum
                             ;length. The output is right-adjusted.
                             ;Empty positions are filled with blanks.

SC_REFRESH(VTL000_NR+5,3)   ;Refresh of VTL005.dat with line cursor on
                             ;line 3

SC_REFRESH(34068,1)        ;Refresh of error display with line cursor
                             ;on line 1

SC_ZEI_REL(-1,0)           ;Moves the writing position by one
                             ;character width to the left

SC_CHARACTER(':',')        ;Display character ':' on PHG

SC_END                       ;This command starts the output to the
                             ;PHG. All data between
                             ;SC_FOR_MASK and SC_END
                             ;are sent to the PHG

;-----
;Begin example
;-----

Example:

SC_FOR_MASK (Bel_Maske)     ;whatever which mask is active at the moment

SC_NEW_MASK (Mask2)        ;selects mask 2. Deletes the display and
                             ;displays the mask

SC_END                       ;starts output to the PHG

SC_FOR_MASK (Maske2)       ;execute only, if mask 2 is active

SC_ATTRIB (NORMAL)         ;reset attribute

SC_CHARSET (CS_9_TIMES_11) ;character set 9*11

SC_GOTOXY (10,20)          ;cursor position x=10, y=20

SC_TEXT ('Meas. Value')    ;text output

SC_GOTOXY (10,35)          ;cursor position x=10, Y=35

SC_INTEGER (MW, 8)         ;variable output in integer format

SC_END                       ;starts output to the PHG

;-----
;End example
;-----

PROGRAM_END                 ;End of main program

; End of main program

```

Controlling the PHG2000 output

6.4 Outputs to the PHG2000 in subroutine technique

Decoding and output of the high and low control bytes

The transfer parameter param contains the value of the command which is to be sent to the PHG2000. The character sequence ".." effects that the output is stored intermediately and only sent to the PHG2000 by the command SC_END.

 **This subroutine is also required by other SUB's (subroutines) for the transfer of the command to the PHG2000 !**

```
SUB SCHR_GNZ_WRD (VALUE INTEGER:PARAM)
BEGIN
    PARAM = PARAM MOD 65536
    WRITE PHG2000, CHR(PARAM MOD 256), CHR(PARAM/256), ..
SUB_END
```

Indication for which mask the output is determined

The variable 'mask No' contains the mask number for which the output is determined.

```
SUB SC_FOR_MASK (VALUE INTEGER:MASK NO)
BEGIN
    SCHR_GNZ_WRD (MASK NO)
SUB_END
```

Select mask for the display on the PHG

The variable 'mask No' contains the mask number which is to be displayed on the PHG2000.

```
SUB SC_NEW_MASK (VALUE INTEGER:MASK NO)
BEGIN
    SCHR_GNZ_WRD (CD_N_MASK)
    SCHR_GNZ_WRD (MASK NO)
SUB_END
```

Controlling the PHG2000 output

Output of the commands to the PHG2000

The WRITE command without ".." starts the transfer to the PHG2000.

```
SUB SC_END
BEGIN
    WRITE PHG2000, CHR(CD_END), CHR(0)
SUB_END
```

Select display type

The transfer parameter 'attribute' contains the value for the display attribute. The available attributes and their values are described in the constant declaration. If several attributes are to be set simultaneously, the constants can be added, see example in main program.

```
SUB SC_ATTRIB (VALUE INTEGER:ATTRIBUTE)
BEGIN
    SCHR_GNZ_WRD (CD_ATTRIB)
    WRITE PHG2000, CHR (ATTRIBUTE), ..
SUB_END
```

Select point matrix

The variable 'character set' contains the value for the point matrix of the characters to be displayed. The available character sets and their values are described in the constant declaration.

```
SUB SC_CHARSET (VALUE INTEGER:CHARACTER SET)
BEGIN
    SCHR_GNZ_WRD (CD_CHARSET)
    WRITE PHG2000, CHR (CHARACTER SET), ..
SUB_END
```

Determine writing positions on the display

The transfer parameter 'x_pixel' contains the value for the horizontal pixel position onto which the cursor is to be positioned in the display. In 'y_pixel' the value for the vertical position is displayed.

```
SUB SC_GOTOXY (VALUE INTEGER:X_PIXEL, Y_PIXEL)
```


Controlling the PHG2000 output

```
BEGIN
    SCHR_GNZ_WRD (CD_GOTOXY)
    SCHR_GNZ_WRD (X_PIXEL)
    SCHR_GNZ_WRD (Y_PIXEL)
SUB_END
```

Draw line

The transfer parameters 'from_x' and 'from_y' contain the values in pixels for the starting position, the parameters 'to_x' and 'to_y' contain the values in pixels for the end position of the line.

```
SUB SC_LINE (VALUE INTEGER:FROM_X, FROM_Y, TO_X, TO_Y)

BEGIN
    SCHR_GNZ_WRD (CD_LINE)
    SCHR_GNZ_WRD (FROM_X)
    SCHR_GNZ_WRD (FROM_Y)
    SCHR_GNZ_WRD (TO_X)
    SCHR_GNZ_WRD (TO_Y)
SUB_END
```

Output of an individual character

The transfer parameter 'character_value' contains the character to be displayed. The length is 2 (CHR (2)), as the final character (CHR (0)) must also be counted.

```
SUB SC_CHARACTER (VALUE CHARACTER: CHARACTER_VALUE)

BEGIN
    WRITE PHG2000, CHR (CD_TEXT), CHR (2), CHARACTER_VALUE, CHR (0), ..
SUB_END
```

Output of text string

The variable 'text' contains the character sequence to be displayed. As length for the character sequence, the length of the line can be indicated with 80 + 1 (+1 for CHR(0)). CHR (0) is the end identification of the output.

Controlling the PHG2000 output

```
SUB SC_TEXT (VALUE TEXT:DISPLAY)

BEGIN

WRITE PHG2000, CHR(CD_TEXT), CHR(81), DISPLAY, CHR(0), ..

SUB_END
```

Update display

The transfer parameter 'line_No' contains the line to be actualized, var_no the number of the system variable.

```
SUB SC_REFRESH (VALUE INTEGER:VAR_NR, ZEILEN_NR)

BEGIN

    SCHR_GNZ_WRD (CD_REFRESH)

    SCHR_GNZ_WRD (VAR_NO)

    SCHR_GNZ_WRD (LINE_NO)

SUB_END
```

Shift writing position by n characters

The transfer parameters 'delta_column' and 'delta_line' contain the new column resp. the new line number relative to the preceding positions.

```
SUB SC_ZEI_REL (VALUE INTEGER:DELTA_COLUMN, DELTA_LINE)

BEGIN

    SCHR_GNZ_WRD (CD_ZEI_REL)

    WRITE PHG2000, CHR(DELTA_COLUMN), CHR(DELTA_LINE), ..

SUB_END
```

Clear PHG display

The complete PHG display is cleared.

```
SUB SC_CLS

BEGIN

    SCHR_GNZ_WRD (CD_CLS)

SUB_END
```

Controlling the PHG2000 output

Display decimal number

The transfer parameter 'dec_value' contains the decimal number to be displayed. The length of the number, in this case CHR(9), must also be transferred to the PHG. CHR(0) is the end identification of the output.

```
SUB SC_DEC (VALUE DEC:DEC_VALUE)
BEGIN
    WRITE PHG2000, CHR(CD_TEXT), CHR(9), DEC_VALUE, CHR(0), ..
SUB_END
```

Display integer

The transfer parameter integer_value contains the integer to be displayed. Before the display of the value is carried out, it is checked whether an error (W_ERR <> 0) has occurred during the conversion INTEGER → ASCII. In this case, only lines are displayed instead of the value. CHR (0) is the end identification of the output.

```
SUB SC_INTEGER (VALUE INTEGER:INTEGER_VALUE, LENGTH)
    TEXT:DISPLAY
    INTEGER:W_ERR, I
BEGIN
    INTEGER_ASC(INTEGER_VALUE, DISPLAY, 1, LENGTH, W_ERR)
    DISPLAY[LENGTH+1] = CHR(0)
    IF W_ERR<>0                ; <>0= error of conversion
                                ; has occurred
        THEN BEGIN
            I=1
            REPEAT LENGTH TIMES
                DISPLAY[I]='-' ; display of lines only in case of error
                I = I + 1
            REPEAT_END
        END
    SC_TEXT(DISPLAY)
SUB_END
```

Controlling the PHG2000 output

Example

```
SC_NEW_MASK(Mask2) ; Select mask
```

```
SC_END
```

```
SC_FOR_MASK(Mask2)
```

```
SC_CLS ; Clear screen
```

```
SC_ATTRIB(NORMAL) ; Reset attributes
```

```
SC_CHRSET(CS_9_TIMES_11) ; Character set 9*11
```

```
SC_END
```

```
SC_FOR_MASK(Mask2)
```

```
SC_GOTOXY(10,20) ; Cursor position x = 10, y = 20
```

```
SC_TEXT('Messwert=') ; Text output
```

```
SC_GOTOXY(25,20) ; Cursor position x = 25, y = 20
```

```
SC_INTEGER(MW) ; Variable output in integer format
```

```
SC_END
```

If the BAPS3 subroutines are not available, the PHG2000 output can also be accessed directly with the commands used in the subroutines.

Define/Teach

7 Define/Teach

The operating mode Define and Teach offers the possibility to represent several points of a point file simultaneously on the display of the PHG2000.

7.1 Display layout

7.1.1 Display structure of Define

Define Prog. name (7) Robot 1 (1)	X	Y	Z (8)
Starting pos	100.000	100.000	50.000
Picking pos	-1234.56	-1234.56	-1234.56
Discharge pos	100.000	20.000	55.000
Placing pos	1000.05	0.000	66.000
Pallet pos	300.000	200.000	70.000
Intermed. pos	11.000	30.005	4.000
Distributor pos	400.000	300.000	0.000
Destin. pos	50.000	40.000	0.001
Test pos (2)	--	--	--
POS (3)	-1111.11	330.000	0.000
Input: # (4) (5)			(6)

Remarks

- 1 Display of the current kinematic, here robot 1.
- 2 The point test pos is not yet defined. Therefore, the point components X,Y,Z are displayed by -- . With a point array, the field index is displayed below the point name in the next line.
- 3 The current position is displayed according to the selected point (@point/point).
- 4 Value input for the selected point.
Additional functions
 - <Mode> + <Shift S>: search desired point.
 - <Mode> + <Shift E>: terminate operating mode Define.
 Alternatively, the operating mode can be left with <Shift Del>.
- 5 Status messages
- 6 Because of the separating lines in the display, max. 15 lines can only be displayed.
- 7 Name of the pkt-file.

Define/Teach

- 8 For 10 to 12 axes 4 head lines, for 13 to 15 axes 5 head lines and for 16 to 18 axes 6 head lines are necessary for the representation of the axes resp. coordinate names. For JC and WC points, the display of the axis resp. coordinate names is carried out according to the selected point.

For machines with five axes it follows this picture

Define Prog. name Robot 1	X A	Y B	Z
Starting pos	100.000 90.000	400.000 30.000	50.000
Picking pos	-1234.56 555.000	-1234.56 0.000	-1234.56
Discharge pos	100.000 76.005	20.000 54.008	55.000
Placing pos	1000.05 20.000	0.000 30.000	66.000
POS	-1111.11 90.000	330.000 40.000	0.000
Input: #			

7.1.2 Display structure of Teach

The operating mode Teach requires the assignment of the axis keys in addition to the display of the points. Therefore, the following display appears for the operating mode Teach for a system with three axes.

Define Prog. name Robot 1	X	Y	Z
Starting pos	100,000	100.000	50.000
Picking pos	-1234.56	-1234.56	-1234.56
Discharge pos	100.000	20.000	55.000
Placing pos	1000.05	0.000	66.000
Pallet pos	300.000	200.000	70.000
Intermed. pos	11.000	30.005	4.000
Distributor pos Y plus	400.000	300.000	0.000
Destin. pos	50.000	40.000	0.001
Test pos	--	--	--
POS (WC) (3)	-1111.11	330.000	0.000
X plus	Y plus	Z plus	(1)
X minus	Y minus	Z minus	(2)

Define/Teach

- 1/2 These lines must be created by the BDT editor. To do so, a mask must be generated which contains exactly these two lines. The positions are entered by the rho4 into the free arrays of this mask.
- 3 The actual position is displayed according to the selected coordinate system. In the example, world coordinates (WC) are selected.

 **The remarks for Define are also valid for Teach with the exception of remarks 3, 4 and 5.**

For systems with five axes, the following screen is displayed

Define Prog. name Robot 1	X A	Y B	Z
Starting pos	100.000 90.000	400.000 30.000	50.000
Picking pos	-1234.56 555.000	-1234.56 0.000	-1234.56
Discharge pos	100.000 76.005	20.000 54.008	55.000
Placing pos	1000.05 20.000	0.000 30.000	66.000
POS (WC)	-1111.11 90.000	330.000 40.000	0.000
X plus	Y plus	Z plus	
X minus	Y minus	Z minus	

 **For systems with four and six axes, the display corresponds to that of a format with five axes.**

Define/Teach

7.2 Operation

7.2.1 Operation of Define

Defining the point values in the operating mode Define is carried out by numerical input.

The positioning onto the desired point component is carried out with the cursor keys. The selected component is displayed inverted on the PHG.

Point components which have not yet been defined are represented with "--".

The new point value is stored by pressing <Enter>.

Operation of Define with three axes

Define	X	Y	Z
Prog. name Robot 1			
Starting pos	100,000	100.000	50.000
Picking pos	-1234.56	-1234.56	-1234.56
Discharge pos	100.000	20.000	55.000
Placing pos	1000.05	0.000	66.000
Pallet pos	300.000	200.000	70.000
Intemed. pos	11.000	30.005	4.000
Distribut. pos Y plus	400.000 50.000	300.000 40.000	0.000 0.001
Destination pos	--	--	--
Test pos			
POS	-1111.11	330.000	0.000
Input: #			

In the example, the Y-component of the point Discharge pos is to be changed.

Operating steps

- 1 Position the cursor keys onto the desired component Discharge pos.Y.
- 2 Numeric input of the point value in the display with input of : #.
- 3 Storing of the new point value with the Enter key.
- 4 Further point values can be changed by the corresponding operation.

<Mode> + <Shift> <S>: search further point

<Mode> + <Shift> <E>: terminate operating mode Define

 **Point values which have not yet been defined, as e.g. in the point Test pos, are defined by the same procedure.**

Define/Teach

7.2.2 Operation of Teach

Operation of Teach with three axes

The following example shows a system with three axes. By scrolling with the cursor keys the point for the teach operation can be selected.

Define	X	Y	Z
Prog. name Robot 1			
Starting pos	100.000	100.000	50.000
Picking pos	-1234.56	-1234.56	-1234.56
Discharge pos	100.000	20.000	55.000
Placing pos	1000.05	0.000	66.000
Pallet pos	300.000	200.000	70.000
Intemed. pos	11.000	30.005	4.000
Distribut. pos Y plus	400.000 50.000	300.000 40.000	0.000 0.001
Destination pos	--	--	--
Test pos			
POS (WC)	-1111.11	330.000	0.000
X plus	Y plus	Z plus	
X minus	Y minus	Z minus	

Operating steps

- 1 Selection of the desired point Picking pos with the cursor keys.
- 2 Teaching of the new position is carried out in jog mode with the axis keys indicated in the display. The actual position is always displayed dynamically in the lower part of the display.
- 3 Store of the new point value with the Enter key.
- 4 Further point values can be changed by the corresponding operation.
- 5 With <Move> resp. <Move> + <Linear>, the direct approach of points already taught can be selected. The movement is carried out by actuating the permission and the Enter key.
- 6 With <Shift> + <Arrow right>, it is possible to search a desired point.
- 7 The operating mode is left with <Shift> + <Arrow left>.

 **Points that have not yet been taught, as e.g. Test pos, are taught as described in steps 1 to 4.**

Define/Teach

Operation of Teach with five axes


The following example shows a system with five axes.

The operation is at first carried out as for a system with three axes.

Define Prog. name Robot 1	X A	Y B	Z
Starting pos	100.000 90.000	400.000 30.000	50.000
Picking pos	-1234.56 555.000	-1234.56 0.000	-1234.56
Discharge pos	100.000 76.005	20.000 54.008	55.000
Placing pos	1000.05 20.000	0.000 30.000	66.000
POS (WC)	-1111.11 90.000	330.000 40.000	0.000
X plus	Y plus	Z plus	
X minus	Y minus	Z minus	

Operation steps

- 1 Select the desired point Picking pos with the cursor keys.
- 2 Approach new position in manual mode via the axis keys X,Y,Z, etc. indicated in the display. Dynamic display of the actual position in the lower part of the display in array POS.
- 3 If the axes A and B are to be moved it is necessary to switch over the axis key assignment with <Goup> or another key which must be defined.
- 4 Store the new point value with <Enter>.
- 5 The input of further points is carried out correspondingly
 <Shift> + <Arrow right>: search desired point.
 <Shift> + <Arrow left>: leave operating mode.

 **For systems with several kinematics, the switch-over is carried out as before with <KIN>. There is no display of the actual position for systems with 13 axes and more.**

SRCAN functions

8 SRCAN functions

The Bosch swivel arm robots turboscara SR6 und turboscara SR8 are equipped with a SRCAN module on which the serial number of the mechanics and a machine-specific parameter set are stored.

 **The SRCAN functions described in the following are only valid for the Bosch swivel arm robots SR6 and SR8!**

8.1 Electric type plate

Mode 7.8.6 Elec. type plate

In this mode functions for reading and writing the SRCAN module are available for the user.

 **In this mode inputs are only possible with a set supervisor password, see section 8.2, otherwise these functions are disabled!**

The functions are kinematic-dependent. The current kinematic is displayed on the PHG for all modes in the first line. Under mode 7.8.6.5 the desired kinematic can be selected with the KIN key.

8.1.1 Read serial number

Mode 7.8.6.1 Read serial number

Reads the serial number from the SRCAN module and compares it with the serial number from machine parameter 314.

8.1.2 Write serial number

Mode 7.8.6.2 Write serial number

Writes the serial number of the selected kinematic stored in machine parameter 314 into the SRCAN module.

 **With all SRCAN-write-operations, the ready-contact is opened!**

SRCAN functions

8.1.3 Read machine parameter set

Mode 7.8.6.3 Read machine para.

Reads the machine parameter set filed in the SRCAN module and filtes it with the file SRCAN.bnr. It is determined in this file which parameters are to be taken over.

8.1.4 Write machine parameter set

Mode 7.8.6.4 Write machine para.

Writes the complete parameter set of the rho4 into the SRCAN module.

 **With all SRCAN-write-operations, the ready-contact is opened!**

8.1.5 Selection of the kinematic

Mode 7.8.6.5 Kinematic selection

Under this mode the desired kinematic for the SRCAN functions available under Mode 7.8.6.1 to 7.8.6.4 is selected.

8.2 Supervisor password

Mode 7.8.4 Set Supervisor

Under this mode the supervisor password is set. The password to be entered is similar to the MPP password.

This password must be entered before selecting a SRCAN function and before changing the file attribute 'Hidden'.

Mode 7.8.4 Reset Supervisor

Under this mode an active supervisor mode is deactivated. As input, the MPP password is expected.

After a successful selection, the message 'supervisor disabled' is displayed on the PHG.

File and User Memory Functions

9 File and User Memory Functions

9.1 Changing the attributes of the files

Mode 9.9 File attributes

Under this mode the file attributes of the files in the user memory can be changed.

 **The attribute 'Hidden' can only be set if the supervisor password (Mode 7.8.4) has been set.**

Permitted attributes are:

R (Read)	File can only be opened for reading
W (Write)	File can only be opened for writing
D (Delete)	File can only be deleted
H (Hidden)	File is not displayed in 'RC-list' of control and in Online

The attributes displayed inverted are R, W, D.

 **All attributes are only active with file operations in the control!**

9.2 User memory functions

Mode 9.6 Store user memory

The complete user memory of the rho4 is stored on the hard disk (dump). The files cannot be read under Windows as the complete user memory is stored as binary file.

Mode 9.7 Restore user memory

The copy (dump) of the rho4-user-memory available on the hard disk is stored back into the control. Files which possibly exist in the rho4 are overwritten or deleted during this process.

File and User Memory Functions

Notes:

File list

10.1 Creation of the objects in the BDT editor

In order to create the mask shown in the preceding illustration, all necessary objects must at first be created. For that, the required texts are at first edited.

In the second step, the variables marked by place holders are declared according to the PHG2000-variable-coding with the code numbers 34561 resp. 34567. The data types entered in the BDT editor must correspond to the basic types of the description of the PHG2000-variable-coding.

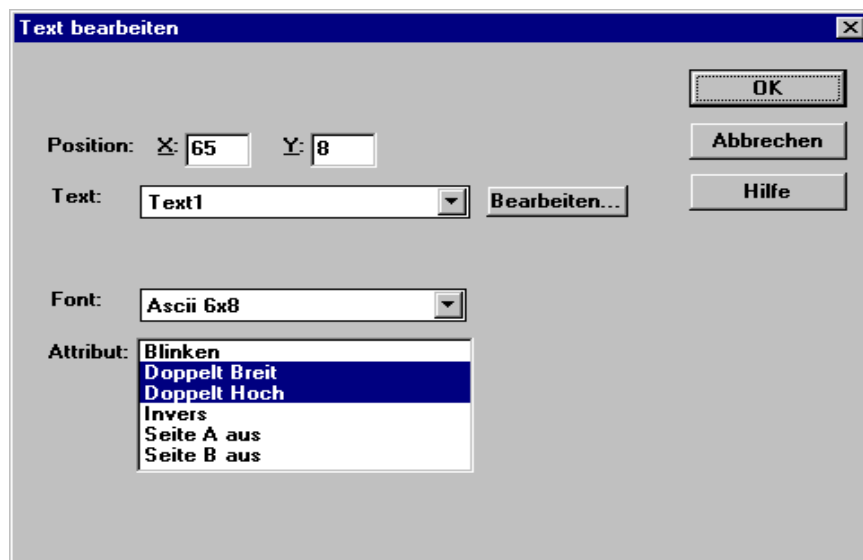
Furthermore, it must be determined in the field Array whether it is an indexed variable or a simple variable and in the area 'access', whether the variable is read or written. (Access: Read/Write).

10.1.1 Headline 'Dateiliste' on the PHG2000

For the headline, a text variable must be declared so that the text Dateiliste is displayed. In the BDT editor the following declaration must be met.

The text variable has the following characteristics

Name:	FText1
Text:	Text1
Attribute:	Double height, double width



The text 'Dateiliste' can directly be entered into the mask.

File list

10.1.2 List box for the file list

The variable for 'Dateiliste', No. 34561, see chapter 3, must be declared in order to display the file names within the list box. When this variable is declared, the variable text list can be created. In the BDT editor the following declarations must be made for that.

The two variables have the following characteristics

File list (No. 34561)

The 'Variable bearbeiten' dialog box is used to configure a variable. It contains the following fields and options:

- Name:** DAT_LISTE
- Datentyp:** String (variable Textliste)
- Pollingzeit:** 0 ms
- Nummer:** 34561
- Array** Index: 1
- Zugriff:**
 - Read
 - Write
 - Read/Write

Buttons: OK, Abbrechen, Hilfe

The text list has then the following characteristics

The 'Textliste bearbeiten' dialog box is used to configure a text list. It contains the following fields and options:

- Rückgabe-Variable:** <keine Variable> (with a 'Bearbeiten' button)
- X:** 6 **Spalten:** 40 **Font:** Ascii 6x8
- Y:** 33 **Zeilen:** 8
- Textlistenart:**
 - fest
 - variabel
- Variable:** DAT_LISTE (with a 'Bearbeiten' button)

Buttons: OK, Abbrechen, Hilfe

File list

10.2 Display of the memory allocation

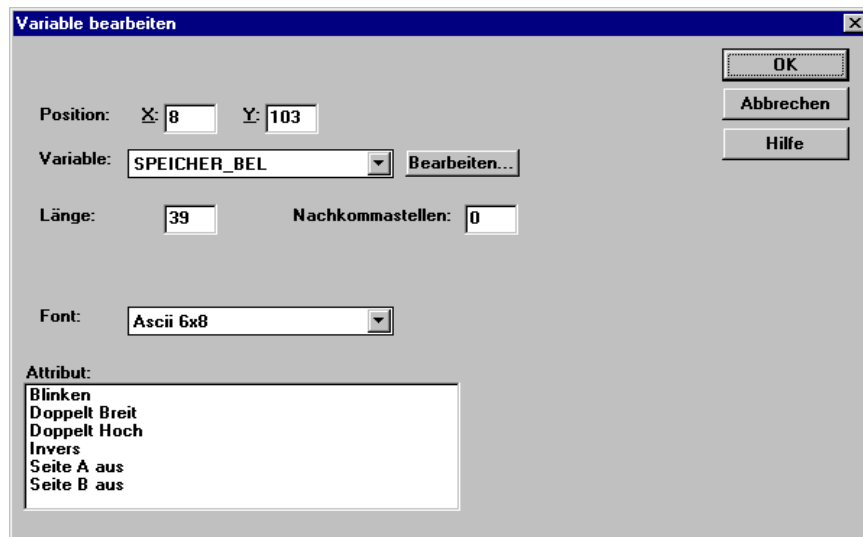
The variable `memory_allocation`, No. 34567, see chapter 3 must be declared to be able to display the information about the memory allocation of the user memory. When this variable is declared, the text variable can be created. In the BDT editor the following declarations must be made for that.

Memory allocation, No. 34567

Name: SPEICHER_BEL
Data type: String
Number: 34567
Array: no
Index: -1
Access: Read

The text variable has then the following characteristics

Name: Fvariable2
Variable: SPEICHER_BEL
Length: 38
Positions after comma: 0



 The names of the variables, text list and text variable can be freely selected by the user and do not influence the function.

File list

10.3 PHG2000 display

The following illustration shows the PHG2000 display with a file list and the data about the memory allocation. The file display contains the name, the size and the creation date. The current position of the cursor is displayed inverted. In this example, the cursor is positioned in the list box of the file PHG_2002.ird. Scrolling within the list box is carried out with <Shift> + <Up arrow> resp. <Shift> + <Down arrow>. So it is also possible to list the files which cannot be displayed in the list box at the moment.

Below the list box different pieces of information about the user memory is displayed.

These are:

- Total storage size
- Occupied storage
- Free storage

Dateiliste				
PHG_2001.PKT	848	Byte	24.09.95	09:40
PHG_2002.ERR	1527	Byte	25.08.95	14:10
PHG_2002.IRD	1052	Byte	25.08.95	14:11
PHG_2002.PKT	1060	Byte	25.08.95	14:10
PHG_2002.QLL	747	Byte	13.09.95	13:35
PHG_2002.SYM	2160	Byte	25.08.95	14:10
PHG_2003.ERR	1527	Byte	29.08.95	12:12
PHG_2003.IRD	1072	Byte	29.08.95	12:12
131072 Bytes		29841 bel.	90880 frei	

File list

Notes:

Process info

11.2 Creation of the objects in the BDT editor

For set-up of the mask in the preceding illustration, it is at first necessary to create all required objects.

For that purpose, the 16 required texts are at first edited

		Process Info	
Number Processes	: 1		
normal	: 2		7
permanent	: 3		
sub	: 4		
faulty	: 5	Priority	: 8
		State	: 9
Process list	: 6	Process type	: 10
		Kinematic	: 11
		QLL-line	: 12
		QLL-insert line	: 13
		Ext. programm	: 14
		level	: 15
		Process error	: 16

In the second step, the variables marked by place holders, are declared, see chapter 6, according to the PHG2000-variable-coding with the code numbers 34304 to 34322. The data types entered in the BDT editor must be conform with the basic types of the description of the PHG2000-variable-coding. It must further be determined whether it is an indexed or a simple variable and whether the variable is read or written (access: Read/Write). For the complete process display 17 variables are to be declared. With the variable 34322, a selected process can be stopped.

		Process Info	
Number processes			
normal	: 01 1	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1	
permanent	: 01 2		5
sub	: 01 3		
faulty	: 01 4	Priority	: 0001 6
		State	: 000001 7
Prozess list		Process type	: 000001 8
		Kinematic	: 0001 9
		QLL-line	: 0001 10
		QLL-insert line	: 0001 11
		Ext. programm	: 00000001 12
		level	: 0001 13
		Process error	: 0001 14
			00000000000000000001 15

Process info

No.	Name	Data type	Polling	Code No.	Array	Index	Access
1	ANZ_NORM_PROZ	Word	0	34305	No	-1	R
2	ANZ_PERM_PROZ	Word	0	34306	No	-1	R
3	ANZ_SUB_PROZ	Word	0	34307	No	-1	R
4	ANZ_FEHL_PROZ	Word	0	34308	No	-1	R
5	PROZ_NAME	String	0	34316	No	-1	R
6	PROZ_PRIO	Word	0	34312	No	-1	R
7	PROZ_ZUSTAND	String	0	34313	No	-1	R
8	PROZ_ART	String	0	34311	No	-1	R
9	KINEMATIK	Word	0	34319	No	-1	R
10	QLL_ZEILE	Word	0	34317	No	-1	R
11	QLL_EINF_ZEILE	Word	0	34318	No	-1	R
12	EXT_PROG	String	0	34314	No	-1	R
13	PROG_EBENE	Byte	0	34315	No	-1	R
14	PROZ_FEHLER_CODE	Word	0	34320	No	-1	R
15	PROZ_FEHLER_text	String	0	34321	No	-1	R

11.2.1 List box for the process list

The most important variable is the process list No. 34309. It must be declared in any case to be able to display process names within the list box. The selected process is returned to the control via the return variable 'selected process No. 34310. When these two variables are declared, the variable text list can be created with their help.

In the BDT editor, the following declarations are to be met for these objects

<pre> Number processes normal : 01 permanent : 01 sub : 01 faulty : 01 Process list 16, 17 XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX </pre>	<h2 style="margin: 0;">Process Info</h2> <p style="font-size: 2em; margin: 0;">00000000000001</p> <pre> Priority : 0001 State : 000001 Process type : 000001 Kinematic : 0001 QLL line : 0001 QLL insert line : 0001 Ext. programm : 00000001 level : 0001 Process error : 0001 00000000000000000001 </pre>
---	---

Process info

The two variables have the following characteristics

No.	Name	Data type	Polling	Code No.	Array	Index	Access
16	PROZ_LISTE	String (variable text list)	0	34309	Yes	1	R
17	PROG_ANGEW	Word	0	34310	No	-1	R/W

The text list has then the following characteristics:

Name:	P_LISTE
Return variable:	PROZ_ANGEW
Columns:	15
Lines:	7
Type of text list:	variable
Variable:	PROZ_LISTE

The names of the variables and the text list can be freely selected by the user and do not influence the function.

11.2.2 Global process information

The variables with the numbers 34305 to 34308 are determined for the display of global process information as e.g. the number of normal and permanent processes. In the BDT editor, the declarations are to be made as in the two following examples

Name	Data type	Polling	Code No.	Array	Index	Access
ANZ_NORM_PROZ	Word	0	34305	No	-1	R
ANZ_PERM_PROZ	Word	0	34306	No	-1	R

11.2.3 Information on selected process

All further variables refer to the selected process, as e.g. process priority or qll-line. Not all variables must be used. If a user e.g. does not use external programs or include instructions, he can omit the texts Ext. program, level and qll-include-line and the corresponding variables. The BDT variable declarations are explained by the two examples below process priority and qll-line.

Name	Data type	Polling	Code No.	Array	Index	Access
PROZ_PRIO	Word	0	34312	No	-1	R
QLL_ZEILE	Word	0	34317	No	-1	R

Process info

11.3 PHG2000 display

The following illustration shows the PHG2000 display with six active user processes. The process ROBOT_A is selected. The pieces of information in the right half of the display refers to this process. The scrolling within the process list box mentioned at the beginning is carried out with the PHG keys <Shift> and <Up arrow> resp. <Shift> and <Down arrow>. In this way, the preceding process PAR_EA S03 or the following process ROBOT_B can for example be selected.

The variable process name (No. 34316) is used in the actual example to display the name of the selected process in the right display field below the headline Prozess-Info. If no process is active this field and the list box are empty.

Process errors are displayed with error code and with an error text with 20 characters. In the example for the process display, see following illustration, no process error has occurred. As error code 0 and as error text 20 spaces are displayed. The position at which the error text for the placeholder 00000000000000000001 is displayed if an error has occurred, is shown in the illustration under point 11.2.1x.

Number Processes		Process Info
normal	: 2	ROBOT_A
permanent	: 1	Priority : 100
sub	: 3	State : RUNNING
faulty	: 0	Process type : norm.
Process list		Kinematic : 1
PAR_EA		QLL line : 39
PAR_EA S01		QLL insert line : 0
PAR_EA S02		
PAR_EA S03		
ROBOT_A		Ext. programm :
ROBOT_B		level : 1
		Process error : 0

Process info

11.4 Stopping processes

The system variable with the code number 34322 can be used to realize a simple stop of processes via the process display. If any PHG key, e.g. <Delete>, is linked with this variable, it is possible to stop the just selected process in the process display with this key. The effect is similar to the operation of the PHG3 mode 10.2 stop processes. Different to this function, no safety request is carried out here. Additionally, individual sub-processes can be stopped, this is not possible in the PHG mode 10.2.

Because of the higher risk of stopping processes unintendedly, this function is not activated in the standard process info. But it can be realized by the user in the BDT editor as follows

- 1 Declare system variable No. 34322 (stop process)

Name	Data type	Polling	Code No.	Array	Index	Access
STP_ANGEW_PROZESS	Byte	0	34322	No	-1	W

- 2 Allocate the desired key in the BDT editor, menu Mask/Key. (In the example the <Delete> K35 is used as usual)

Name	Key	Pictogram	Action	Value (Press)	Value (Release)
STP_ANGEW_PROZESS	K35 norm.	No picture	Variable	1	0

This declaration means that when pressing <Delete> the currently selected process is stopped.

A higher safety in avoiding unintended process stops can be obtained by selecting a complicated key combination, e.g. <Shift> + <Delete> instead of <Delete>.

Restoring the PHG display

12 Restoring the PHG display

This expansion of the operating system better supports a change between operating system displays on the PHG and displays of BAPS processes.

Store the PHG display

With the ascending edge of the signal DIS_PHGM_RCI, the current PHG display is stored.

With the trailing edge of this signal, a display is initiated on the PHG by the previously stored display. If a dynamic display is active e.g. axis display, the stored display is then immediately updated again.

If, however, a PHG-interface-selection is carried out while DIS_PHGM_RCI is active, no display is initiated for the trailing edge of this display.

Application example 1

A message is to be displayed by the BAPS process on the PHG and is to be acknowledged by the user by the actuation of any key. Afterwards, the operator should be able to continue with his current operation.

Program extract:

```

Dis_Phgm_RCI = 1                ; Output which acts onto the RCI-signal
                                ; No. 227 (O28.3) in the PLC program

WAIT 0.1                        ; Signal runtime RC -> SPS -> RC

WRITE PHG, message, any_key

Cond_Phgm = CONDITION(PHG)      ; Waits until output

WAIT 2.0                        ; Minimum time for the display, to avoid
                                ; unintended acknowledging

READ_BEGIN PHG                  ; Deletes characters possibly already entered

READ PHG, key                   ; Waits until key is actuated

Dis_Phgm_RCI = 0                ; Displays the old display again

```

Restoring the PHG display

Application example 1

Operating system functions (e.g. MSD display, Process display, Define) are to be called by the BAPS-dialog-process.

It is determined by the file PHGCODE.DAT which functions can be called, see point 4.2.2.

Example of a PHGCODE.DAT

```
00 = 0           ;Return to basic level
01 = 4.2         ;Teach
02 = 7.2         ;Status messages
03 = 7.12.2     ;automatic MSD-display
```

Program extract:

```
Dis_Phgm_RCI = 1           ; Output which acts on the RCI-signal
                           ; No. 227 (O28.3) in the PLC-program
                           ; Dialog on PHG for function selection
                           ; Writing of the variable CODE, e.g. CODE = 3

IF NOT Base_Ac_RCO        ; Input which corresponds to the RCO-signal
                           ; No. 152 (I19.0) in the PLC-program

THEN BEGIN

    PHGCODE=0             ; INTEGER-output, connected with the signals
                           ; For the PHG functions in the PLC-program
                           ; Trailing edge of PHGCODE=0 effects
                           ; Return to basic level

    WAITE UNTIL Base_Ac_RCO=1 MAX_TIME=3.0
                           ; Spec. treatment if nec. e.g. because of EDITOR

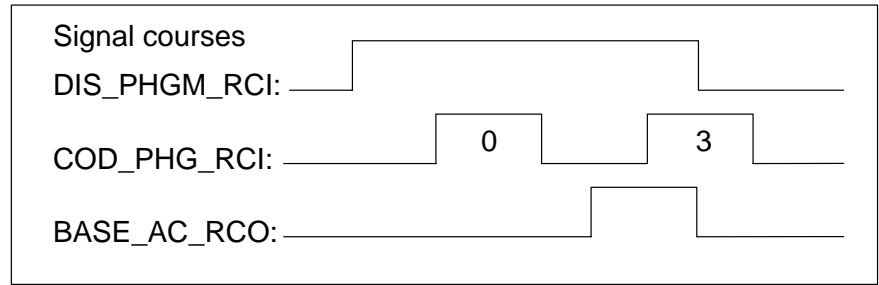
END

PHGCODE=CODE              ; Selection of desired function via
                           ; INTEGER-output connected with the signals for
                           ; The PHG-functions in the PLC-program

WAIT UNTIL Base_Ac_RCO=0 MAX_TIME=3.0

Dis_Phgm_RCI = 0         ; Displays directly the new PHG function and
                           ; Permits again operating system displays
```

Restoring the PHG display



Extension of the command READ PHG

If the signal DIS_PHGM_RCI is set, PHG keys are no longer immediately interpreted by the system, but are collected in a buffer. From this buffer, BAPS processes with the command READ PHG or the operating system are reading after having cancelled the display lock. So it is possible to enter characters at the PHG with DIS_PHGM_RCI = 1 already before, also if a BAPS process is not yet positioned on the command READ PHG. The PHG-key-signals are, however, still made available at the internal interface of the PLC.

Restoring the PHG display

Notes:

Variable assignment of PHG keys

13 Variable assignment of PHG keys

The PHG2000 has 36 keys each of which has a standard assignment in the four levels:

- Basic level
- Shift level (additional actuation of <Shift>)
- Alt level (additional actuation of <Alt>)
- Level for the PCL (Mode 3.2.1 and Mode 6.1, additional actuation of <Shift> + <Alt>)


Via the file Key.bnr; the standard assignment can be changed.



Attention

The operator of the system is responsible for unintended reactions due to a wrong file Key.bnr! Changes concerning the standard key assignment should only be carried out in accordance with Bosch.

The file name is fix. The file length is exactly 256 bytes, from character number 0 to character number 255. The following tables include the standard assignment (key code) in the second column. The indices of the individual keys referring to the tables, can be taken from the picture under PHG-key-indices. The key indices are always valid for all four levels. They indicate the sequence of the entries in the file Key.bnr in the respective level.

 **If during runup of the rho4 a file with the name Key.bnr is available in the user memory, the key assignment is carried out with the content of this file.**

If this file is defective or not available, the standard assignment applies. For a wrong Key.bnr file, no error message is generated by the rho4.

The file Key.bnr is created with a commercial hex editor. The basis is the standard assignment which is available in the file Sta_tast.bnr.

Variable assignment of PHG keys

The following character numbers have a fix assignment and must not be changed


Character number	Content	Key code
	Hex	ASCII
0	5A	'Z'
1 to 7	20	' '
44 to 63	20	' '
64	5A	'Z'
65 to 71	20	' '
108 to 127	20	' '
128	5A	'Z'
129 to 135	20	' '
172 to 191	20	' '
192	5A	'Z'
193 to 199	20	' '
236 to 255	20	' '

If a key is not intended to have any reaction, the code 00 H (Hex) is to be entered for the corresponding key. If the levels Alt and Shift are used, the keys Alt and Shift must be positioned at the same place as in standard assignment.

If the keys Alt and/or Shift are assigned with a code different to 00 H, the corresponding reaction is released by these keys and the Alt- and/or Shift level do not exist any longer.

The effect on the individual key codes is apparent from the standard key assignment and the control reaction.

If the control can no longer be operated from the PHG due to a wrong Key.bnr file, the Key.bnr file can be deleted via ROPS4. A backup of the operating system also leads back to the standard key assignment and with that to the operatability of the control via the PHG.

 **A backup of the operating system, a backup of the machine parameters etc. is always carried out with the same physical keys as in standard operation, also if a Key.bnr file presets other key codes than the standard assignment codes.**

Variable assignment of PHG keys

Example 1

With the K1-key in the basic level, the same reactions are to be released as with the K36-key in the basic level (<Enter> in the standard assignment).

Procedure

- The key index for <Enter>, 23 results from the illustration under the PHG key indices, page 13–5.
- With the key index 23, column 6, the key code for <Enter> to 0A H results from the table on page 13–6, column 3.
- This key code must be inserted instead of the key code for the K1-key in Key.bnr.
- The K1-key has got the key index 40.
- From the table, column 6, character number 40 is identified with key index 40, index in Key.bnr, column 1.
- In Key.bnr, for character number 40, instead of 74 H, 't' in the standard assignment, the code for <Enter>, 0A H, is entered.

Example 2

With the K10-key in the Alt level, the same reaction is to be effected as with the K29-key in the Shift level (<arrow left> in the standard assignment).

Procedure

- The key index for the K29-key, 9 results from the illustration under PHG-key-indices.
- With the key index 9, column 7, the key code for <arrow left> in the shift level to 18 H results from the table, column 3. This key code must be entered instead of the key code for the K10-key in the Alt level in Key.bnr.
- The K10-key has the key index 37.
- The character number in the Alt level, 165 results with key index 37 results from column 8.
- In Key.bnr, for character number 165, instead of 5D H, ']' in the standard assignment, the code für <arrow left>, 18 H, is entered.

Example 3

With the K28-key in the Alt level the same reaction is to be released as with the K33-key in der basic level (<0> in the standard assignment). With that, all functions of the Alt level are shielded.

Procedure

- In Key.bnr with character number 19, the code for <0>, 30 H, is entered instead of 00 H in the standard assignment.

Variable assignment of PHG keys

Example 4

The keys K1 to K8 should not release any reaction in the basic level.





































Procedure

In Key.bnr, the code 00 H is entered for the following character numbers

- K1-key, character number 40, 00 H instead of 74 H, 't' in the standard assignment
- K2-key, character number 41, 00 H instead of 72 H, 'r' in the standard assignment
- K3-key, character number 42, 00 H instead of 69 H, 'i' in the standard assignment
- K4-key, character number 43, 00 H instead of 6B H, 'k' in the standard assignment
- K5-key, character number 38, 00 H instead of 6D H, 'm' in the standard assignment
- K6-key, character number 39, 00 H instead of 66 H, 'f' in the standard assignment
- K7-key, character number 30, 00 H instead of 73 H, 's' in the standard assignment
- K8-key, character number 31, 00 H instead of 67 H, 'g' in the standard assignment

Variable assignment of PHG keys

PHG key indices

Key number	K1	K2	K3	K4
Key index				
	K5	K6	K7	K8
				
	K9	K10	K11	K12
				
	K13	K14	K15	K16
				
	K17	K18	K19	K20
				
	K21	K22	K23	K24
				
	K25	K26	K27	K28
				
	K29	K30	K31	K32
				
	K33	K34	K35	K36
				

Variable assignment of PHG keys

Index in Key.bnr (character number)		Content of Key.bnr for standard key assignment (key code)		Annotation and meaning of the PHG-keys for standard key assignment	PHG-key index in the basic level	PHG-key index in the shift level	PHG-key index in the alt level	PHG-key index in the level for the PCL
DEC	HEX	HEX	ASCII					
0	00	5A	'Z'	-	1	-	-	-
1	01	20	' '	-	-	-	-	-
2	02	20	' '	-	-	-	-	-
3	03	20	' '	-	-	-	-	-
4	04	20	' '	-	-	-	-	-
5	05	20	' '	-	-	-	-	-
6	06	20	' '	-	-	-	-	-
7	07	20	' '	-	-	-	-	-
8	08	30	'0'	0	8	-	-	-
9	09	31	'1'	1	9	-	-	-
10	0A	32	'2'	2	10	-	-	-
11	0B	33	'3'	3	11	-	-	-
12	0C	34	'4'	4	12	-	-	-
13	0D	35	'5'	5	13	-	-	-
14	0E	36	'6'	6	14	-	-	-
15	0F	37	'7'	7	15	-	-	-
16	10	38	'8'	8	16	-	-	-
17	11	39	'9'	9	17	-	-	-
18	12	2D	'_'	-	18	-	-	-
19	13	00		ALT	19	-	-	-
20	14	00		SHIFT	20	-	-	-
21	15	2E	'.'	.	21	-	-	-
22	16	10		DELETE	22	-	-	-
23	17	0A		ENTER	23	-	-	-
24	18	6F	'o'	V-PTP	24	-	-	-
25	19	3D	'='	=	25	-	-	-
26	1A	65	'e'	REPEAT- END	26	-	-	-
27	1B	68	'h'	HALT	27	-	-	-
28	1C	64	'd'	ELSE	28	-	-	-
29	1D	6A	'j'	JUMP	29	-	-	-
30	1E	73	's'	BEGIN	30	-	-	-

Variable assignment of PHG keys

Index in Key.bnr (character number)		Content of Key.bnr for standard key assignment (key code)		Annotation and meaning of the PHG-keys for standard key assignment	PHG-key index in the basic level	PHG-key index in the shift level	PHG-key index in the alt level	PHG-key index in the level for the PCL
DEC	HEX	HEX	ASCII					
31	1F	67	'g'	END	31	-	-	-
32	20	7B	'{'	MODE	32	-	-	-
33	21	20	' '	WC_NAMES	33	-	-	-
34	22	6C	'l'	REPEAT	34	-	-	-

Variable assignment of PHG keys

Index in Key.bnr (character number)		Content of Key.bnr for standard key assignment (key code)		Annotation and meaning of the PHG-keys for standard key assignment	PHG-key index in the basic level	PHG-key index in the shift level	PHG-key index in the alt level	PHG-key index in the level for the PCL
DEC	HEX	HEX	ASCII					
35	23	6E	'n'	TIMES	35	–	–	–
36	24	71	'q'	IF	36	–	–	–
37	25	70	'p'	THEN	37	–	–	–
38	26	6D	'm'	WAIT	38	–	–	–
39	27	66	'f'	UNTIL	39	–	–	–
40	28	74	't'	MOVE	40	–	–	–
41	29	72	'r'	LINEAR	41	–	–	–
42	2A	69	'i'	VIA	42	–	–	–
43	2B	6B	'k'	TO	43	–	–	–
44	2C	20	''	–	–	–	–	–
45	2D	20	''	–	–	–	–	–
46	2E	20	''	–	–	–	–	–
47	2F	20	''	–	–	–	–	–
48	30	20	''	–	–	–	–	–
49	31	20	''	–	–	–	–	–
50	32	20	''	–	–	–	–	–
51	33	20	''	–	–	–	–	–
52	34	20	''	–	–	–	–	–
53	35	20	''	–	–	–	–	–
54	36	20	''	–	–	–	–	–
55	37	20	''	–	–	–	–	–
56	38	20	''	–	–	–	–	–
57	39	20	''	–	–	–	–	–
58	3A	20	''	–	–	–	–	–
59	3B	20	''	–	–	–	–	–
60	3C	20	''	–	–	–	–	–
61	3D	20	''	–	–	–	–	–
62	3E	20	''	–	–	–	–	–
63	3F	20	''	–	–	–	–	–
64	40	5A	'Z'	–	–	1	–	–
65	41	20	''	–	–	–	–	–
66	42	20	''	–	–	–	–	–

Variable assignment of PHG keys

Index in Key.bnr (character number)		Content of Key.bnr for standard key assignment (key code)		Annotation and meaning of the PHG-keys for standard key assignment	PHG-key index in the basic level	PHG-key index in the shift level	PHG-key index in the alt level	PHG-key index in the level for the PCL
DEC	HEX	HEX	ASCII					
67	43	20	' '	-	-	-	-	-
68	44	20	' '	-	-	-	-	-
69	45	20	' '	-	-	-	-	-
70	46	20	' '	-	-	-	-	-
71	47	20	' '	-	-	-	-	-
72	48	5A	'Z'	Z	-	8	-	-
73	49	18		⚡	-	9	-	-
74	4A	59	'Y'	Y	-	10	-	-
75	4B	0E		⚡	-	11	-	-
76	4C	57	'W'	W	-	12	-	-
77	4D	0F		⬆	-	13	-	-
78	4E	58	'X'	X	-	14	-	-
79	4F	53	'S'	S	-	15	-	-
80	50	54	'T'	T	-	16	-	-
81	51	55	'U'	U	-	17	-	-
82	52	56	'V'	V	-	18	-	-
83	53	00		ALT	-	19	-	-
84	54	00		SHIFT	-	20	-	-
85	55	0C		⬇	-	21	-	-
86	56	0B		DELETE	-	22	-	-
87	57	0A		ENTER	-	23	-	-
88	58	51	'Q'	Q	-	24	-	-
89	59	52	'R'	R	-	25	-	-
90	5A	4F	'O'	O	-	26	-	-
91	5B	50	'P'	P	-	27	-	-
92	5C	4B	'K'	K	-	28	-	-
93	5D	4C	'L'	L	-	29	-	-
94	5E	47	'G'	G	-	30	-	-
95	5F	48	'H'	H	-	31	-	-
96	60	7A	'z'	INFO	-	32	-	-
97	61	07		KIN	-	33	-	-
98	62	4D	'M'	M	-	34	-	-

Variable assignment of PHG keys

Index in Key.bnr (character number)		Content of Key.bnr for standard key assignment (key code)		Annotation and meaning of the PHG-keys for standard key assignment	PHG-key index in the basic level	PHG-key index in the shift level	PHG-key index in the alt level	PHG-key index in the level for the PCL
DEC	HEX	HEX	ASCII					
99	63	4E	'N'	N	–	35	–	–
100	64	49	'I'	I	–	36	–	–
101	65	4A	'J'	J	–	37	–	–
102	66	45	'E'	E	–	38	–	–
103	67	46	'F'	F	–	39	–	–
104	68	41	'A'	A	–	40	–	–
105	69	42	'B'	B	–	41	–	–
106	6A	43	'C'	C	–	42	–	–
107	6B	44	'D'	D	–	43	–	–
108	6C	20	''	–	–	–	–	–
109	6D	20	''	–	–	–	–	–
110	6E	20	''	–	–	–	–	–
111	6F	20	''	–	–	–	–	–
112	70	20	''	–	–	–	–	–
113	71	20	''	–	–	–	–	–
114	72	20	''	–	–	–	–	–
115	73	20	''	–	–	–	–	–
116	74	20	''	–	–	–	–	–
117	75	20	''	–	–	–	–	–
118	76	20	''	–	–	–	–	–
119	77	20	''	–	–	–	–	–
120	78	20	''	–	–	–	–	–
121	79	20	''	–	–	–	–	–
122	7A	20	''	–	–	–	–	–
123	7B	20	''	–	–	–	–	–
124	7C	20	''	–	–	–	–	–
125	7D	20	''	–	–	–	–	–
126	7E	20	''	–	–	–	–	–
127	7F	20	''	–	–	–	–	–
128	80	5A	'Z'	–	–	–	1	–
129	81	20	''	–	–	–	–	–
130	82	20	''	–	–	–	–	–

Variable assignment of PHG keys

Index in Key.bnr (character number)		Content of Key.bnr for standard key assignment (key code)		Annotation and meaning of the PHG-keys for standard key assignment	PHG-key index in the basic level	PHG-key index in the shift level	PHG-key index in the alt level	PHG-key index in the level for the PCL
DEC	HEX	HEX	ASCII					
131	83	20	' '	–	–	–	–	–
132	84	20	' '	–	–	–	–	–
133	85	20	' '	–	–	–	–	–
134	86	20	' '	–	–	–	–	–
135	87	20	' '	–	–	–	–	–
136	88	40	'@'	@	–	–	8	–
137	89	21	'!'	!	–	–	9	–
138	8A	2F	'/'	/	–	–	10	–
139	8B	25	'%'	%	–	–	11	–
140	8C	2B	'+'	+	–	–	12	–
141	8D	2A	'*'	*	–	–	13	–
142	8E	3A	':'	:	–	–	14	–
143	8F	3C	'<'	<	–	–	15	–
144	90	3E	'>'	>	–	–	16	–
145	91	06		6+	–	–	17	–
146	92	20	' '	6–	–	–	18	–
147	93	00		ALT	–	–	19	–
148	94	00		SHIFT	–	–	20	–
149	94	2C	','	,	–	–	21	–
150	96	10		DELETE	–	–	22	–
151	97	0A		ENTER	–	–	23	–
152	98	20	' '	5+	–	–	24	–
153	99	20	' '	5–	–	–	25	–
154	9A	20	' '	4+	–	–	26	–
155	9B	20	' '	4–	–	–	27	–
156	9C	20	' '	3+	–	–	28	–
157	9D	20	' '	3–	–	–	29	–
158	9E	20	' '	2+	–	–	30	–
159	9F	20	' '	2–	–	–	31	–
160	A0	08		GROUP	–	–	32	–
161	A1	20	' '	SPACE	–	–	33	–
162	A2	28	'('	(–	–	34	–

Variable assignment of PHG keys

Index in Key.bnr (character number)		Content of Key.bnr for standard key assignment (key code)		Annotation and meaning of the PHG-keys for standard key assignment	PHG-key index in the basic level	PHG-key index in the shift level	PHG-key index in the alt level	PHG-key index in the level for the PCL
DEC	HEX	HEX	ASCII					
163	A3	29)')	–	–	35	–
164	A4	5B	['	[–	–	36	–
165	A5	5D	']]	–	–	37	–
166	A6	3F	'?'	?	–	–	38	–
167	A7	27	'''	'	–	–	39	–
168	A8	5F	'_'	_	–	–	40	–
169	A9	3B	','	;	–	–	41	–
170	AA	20	''	1+	–	–	42	–
171	AB	20	''	1–	–	–	43	–
172	AC	20	''	–	–	–	–	–
173	AD	20	''	–	–	–	–	–
174	AE	20	''	–	–	–	–	–
175	AF	20	''	–	–	–	–	–
176	B0	20	''	–	–	–	–	–
177	B1	20	''	–	–	–	–	–
178	B2	20	''	–	–	–	–	–
179	B3	20	''	–	–	–	–	–
180	B4	20	''	–	–	–	–	–
181	B5	20	''	–	–	–	–	–
182	B6	20	''	–	–	–	–	–
183	B7	20	''	–	–	–	–	–
184	B8	20	''	–	–	–	–	–
185	B9	20	''	–	–	–	–	–
186	BA	20	''	–	–	–	–	–
187	BB	20	''	–	–	–	–	–
188	BC	20	''	–	–	–	–	–
189	BD	20	''	–	–	–	–	–
190	BE	20	''	–	–	–	–	–
191	BF	20	''	–	–	–	–	–
192	C0	5A	'Z'	–	–	–	–	1
193	C1	20	''	–	–	–	–	–
194	C2	20	''	–	–	–	–	–

Variable assignment of PHG keys

Index in Key.bnr (character number)		Content of Key.bnr for standard key assignment (key code)		Annotation and meaning of the PHG-keys for standard key assignment	PHG-key index in the basic level	PHG-key index in the shift level	PHG-key index in the alt level	PHG-key index in the level for the PCL
DEC	HEX	HEX	ASCII					
195	C3	20	' '	–	–	–	–	–
196	C4	20	' '	–	–	–	–	–
197	C5	20	' '	–	–	–	–	–
198	C6	20	' '	–	–	–	–	–
199	C7	20	' '	–	–	–	–	–
200	C8	30	'0'	0	–	–	–	8
201	C9	31	'1'	1	–	–	–	9
202	CA	32	'2'	2	–	–	–	10
203	CB	33	'3'	3	–	–	–	11
204	CC	34	'4'	4	–	–	–	12
205	CD	35	'5'	5	–	–	–	13
206	CE	36	'6'	6	–	–	–	14
207	CF	37	'7'	7	–	–	–	15
208	D0	38	'8'	8	–	–	–	16
209	D1	39	'9'	9	–	–	–	17
210	D2	2D	'_'	–	–	–	–	18
211	D3	00		ALT	–	–	–	19
212	D4	00		SHIFT	–	–	–	20
213	D5	2E	'.'	.	–	–	–	21
214	D6	10		DELETE	–	–	–	22
215	D7	0A		ENTER	–	–	–	23
216	D8	6F	'o'		–	–	–	24
217	D9	3D	'='		–	–	–	25
218	DA	1F		PE	–	–	–	26
219	DB	1E		NOP1	–	–	–	27
220	DC	4D	'M'	M	–	–	–	28
221	DD	6A	'j'		–	–	–	29
222	DE	41	'A'	A	–	–	–	30
223	DF	67	'g'		–	–	–	31
224	E0	7B	'{'	MODE	–	–	–	32
225	E1	20	' '		–	–	–	33
226	E2	3D	'='	=	–	–	–	34

Variable assignment of PHG keys

Index in Key.bnr (character number)		Content of Key.bnr for standard key assignment (key code)		Annotation and meaning of the PHG-keys for standard key assignment	PHG-key index in the basic level	PHG-key index in the shift level	PHG-key index in the alt level	PHG-key index in the level for the PCL
DEC	HEX	HEX	ASCII					
227	E3	62	'b'	SPB	–	–	–	35
228	E4	73	's'	S	–	–	–	36
229	E5	1D		R	–	–	–	37
230	E6	1C		O	–	–	–	38
231	E7	1A		ON	–	–	–	39
232	E8	1B		U	–	–	–	40
233	E9	75	'u'	AN	–	–	–	41
234	EA	45	'E'	I	–	–	–	42
235	EB	6B	'k'		–	–	–	43
236	EC	20	''	–	–	–	–	–
237	ED	20	''	–	–	–	–	–
238	EE	20	''	–	–	–	–	–
239	EF	20	''	–	–	–	–	–
240	F0	20	''	–	–	–	–	–
241	F1	20	''	–	–	–	–	–
242	F2	20	''	–	–	–	–	–
243	F3	20	''	–	–	–	–	–
244	F4	20	''	–	–	–	–	–
245	F5	20	''	–	–	–	–	–
246	F6	20	''	–	–	–	–	–
247	F7	20	''	–	–	–	–	–
248	F8	20	''	–	–	–	–	–
249	F9	20	''	–	–	–	–	–
250	FA	20	''	–	–	–	–	–
251	FB	20	''	–	–	–	–	–
252	FC	20	''	–	–	–	–	–
253	FD	20	''	–	–	–	–	–
254	FE	20	''	–	–	–	–	–
255	FF	20	''	–	–	–	–	–

Select point file and point name

14 Select point file and point name

With help of special function 16, a special point file can be selected for the operating modes Define Mode 4.1, Teach Mode 4.2 and Print Pkt-file Mode 9.4.2 which is offered when selecting the corresponding operating mode without actuating a special key additionally.

For the operating modes Define/Teach, a point name can be selected additionally within the selected Pkt-file.

14.1 Declaration

Declaration of special function 16 in the BAPS program

```
SPC_FCT:16 = pnt_select(TEXT:pnt_dana
                    TEXT:pnt_puna
                    INTEGER:ret_code)
```

Meaning

pnt_select	Name of special function that can be freely selected
pnt_dana	Name of Pkt-file which is offered for Define/Teach File name must be entered with file extension .pkt
pnt_puna	Name of point which is offered within the file
ret_code	
Return-Code =	
0	Everything OK
1	Error of file name
2	Error of extension (pkt required)
3	File not available
4	File cannot be opened
5	File is empty
11	Point name error
12	Point not found in file
13	Index not found (with Pkt-field)

As point name an 'Empty name' can also be transferred i.e. puna = ' '.
In this case, the first point of the pkt-file is offered for Define/Teach.

For the operating mode 'Print Pkt-file', the point name is not important, always the complete file is printed.

The selected names are reset by selecting the corresponding operating mode.

Select point file and point name

Example

```
PROGRAM SPCF16
```

```
SPC_FCT:16 = pnt_select(TEXT:pnt_dana  
                      TEXT:pnt_puna  
                      INTEGER:ret_code)
```

```
TEXT:pntdana, pntpuna  
INTEGER:ret
```

```
BEGIN
```

```
    pntdana = 'TEST.PKT'  
    pntpuna = 'ANFPOS'  
    ret = 0  
    pnt_select (pntdana, pntpuna, ret)
```

```
PROGRAM_END
```

BDT editor

15 BDT editor

15.1 Principles

The application program BDT editor enables to configurate operator devices with graphical capabilities efficiently and comfortably. The graphic user surface Microsoft® Windows™ is required to be able to represent exactly the operator device display in the program.

The BDT editor is subdivided into three zones:

- menu line for executing the possible commands
- working area for the representation of the mask windows and dialog windows
- status line (can be switched off) for displaying the current information and documenting the progress for rather long internal calculations.


15.1.1 Pre-requisites

Pre-requisites for the operation of the BDT editor or the BDT loader are:

- an IBM or compatible personal computer, on which the operating system MS-DOS V3.3 or an older version, however at least MS-Windows V3.1, is installed
- MS-Windows must run in the extended mode
- a mouse
- a free serial interface for the connection of the operator device

The PC should meet the minimal requirements listed in the table, one should strive for the optimal equipment.

Hardware	Minimal equipment	Optimal equipment
Computer system	386	486/Pentium
Working memory	4 MB	8 to 16 MB
Disk drive	3,5"	3,5"
Hard disk	16 MB	100 MB
Graphic adapter	CGA, EGA, VGA, SVGA	SVGA 1024 * 768
Monitor	appropriate to graphic adapter	Color monitor 17"
Serial interface	COM1 or COM2	free COMx
Operating system	DOS 3.1	MS-DOS 5.0 or 6.2x
Windows	Version 3.1 extended mode	Version 3.1 extended mode

 **During the installation, the used computer configuration is checked for the parameters listed above according to the minimal equipment. If these parameters are not complied with, the installation is interrupted with a corresponding error message.**

BDT editor

15.1.2 Installation

To be able to work with the current version of the configuration software BDT editor and the loading software BDT loader, they have to be first installed. The installation program Setup commonly used under Windows is a help.

- ★ Insert disk into drive A: or B:.
- ★ In program manager, select menu 'Datei' (File), command 'Ausführen' ('Execute').
- ★ In command line, enter A:\Setup or click 'Durchsuchen' (Search) and select 'Setup' on drive A: or B:.

Through the installation of the program, the following program symbols are created in the start menu:



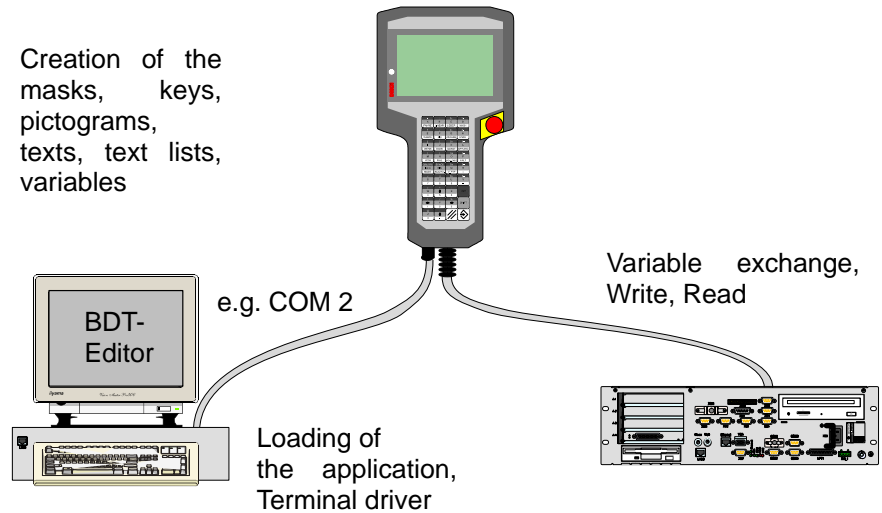
BDT-Editor	Mask editor
BDT-Loader	to download the application
BDT-Help	help file in the direct access
BDT-Readme	Information that is not yet in the manual
BDT-Deinstaller	deletes all files created with the installation

BDT editor

15.1.3 Configuration workstation

For an optimal check of the application in the operator device, the connection to a control unit or a communication network is required, since otherwise the variables are not updated.

A typical workstation for the creation of the operator device application:



15.1.4 Concept

File format of the data

The data of the project is saved in files basing on the xBase file format of dBase. The files are designated in the following as project data base.

To guarantee the consistency of the data in the files and over the files, special check procedures are used. As a result, from the point of view of data consistency, the editing of data base programs supporting xBas formats are not to be recommended. A file edited with such a foreign program is safely recognized by the BDT editor and displayed with a warning message when the project is opened.

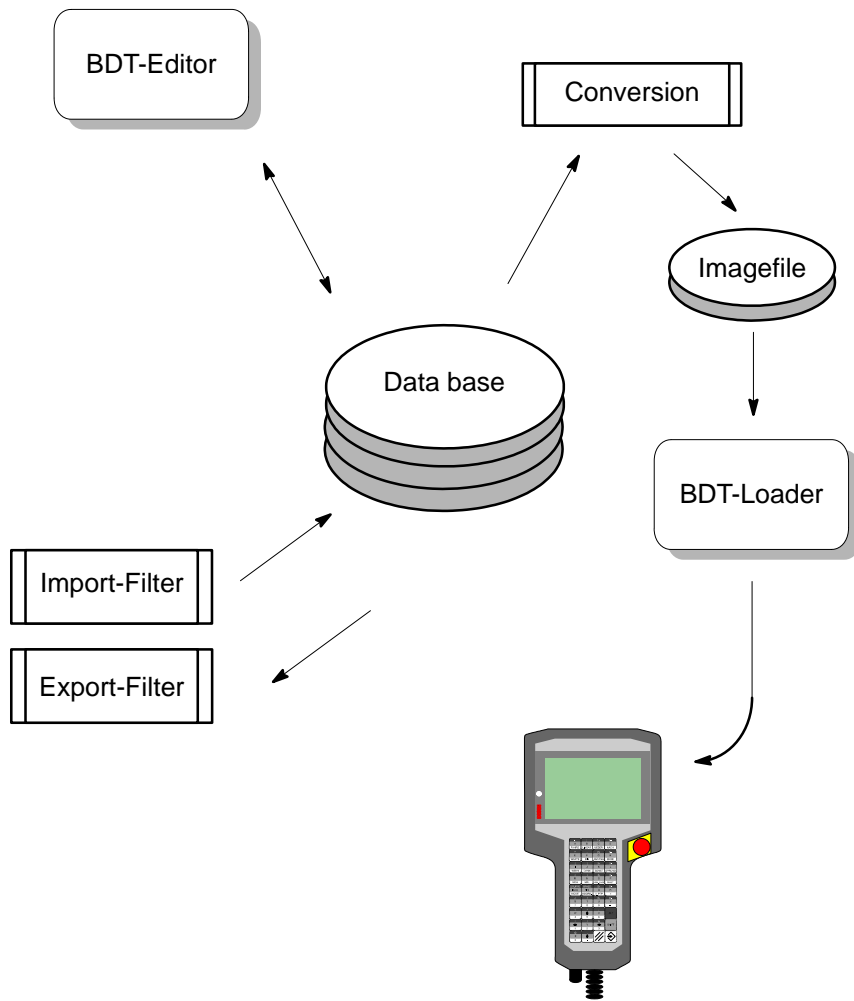
The BDT editor edits exclusively the project data base that is created once for each project. From this project data base, the image file is created through the command 'Generierung' (Generate) with the help of a conversion routine.

The image file contains only the information required absolutely by the operator device for executing the application. This means that e.g. arranged pictograms that are not used by the application are not transferred into the operator device. The image file is transferred with the BDT loader via a serial interface into the operator device.

After a faultless transfer, the operator device can be used. If it has not already been done, it can be connected to the machine control unit.

BDT editor

Structure of the data administration:



BDT editor

Creation of file and directory structure

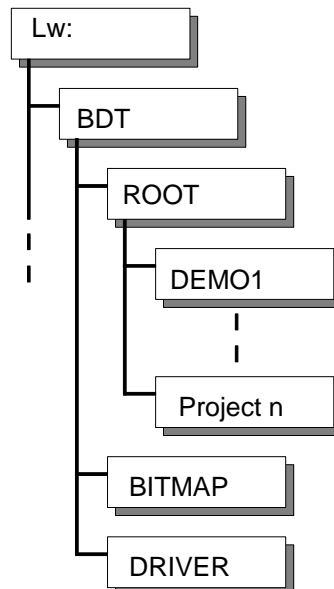
The BDT editor is copied during the standard installation into the directory BDT. Below this directory, a directory ROOT is created. ROOT is the project directory in which the projects are gathered.

The directory with project name is designated as projects. The project specific files, such as e.g. data base files, keyboard layout file etc. are contained in projects.

A project is composed of a directory with the same name and the data base, image and check files contained in it.

In the directory BITMAP, pictograms that are used for all projects (e.g. firm logo, name characters, norm symbols etc.) can be filed.

The directory DRIVER admits special drivers of the operator device. When files of the type <Filename.DRI> are in this directory, they can be transferred into the operator device via the BDT loader.

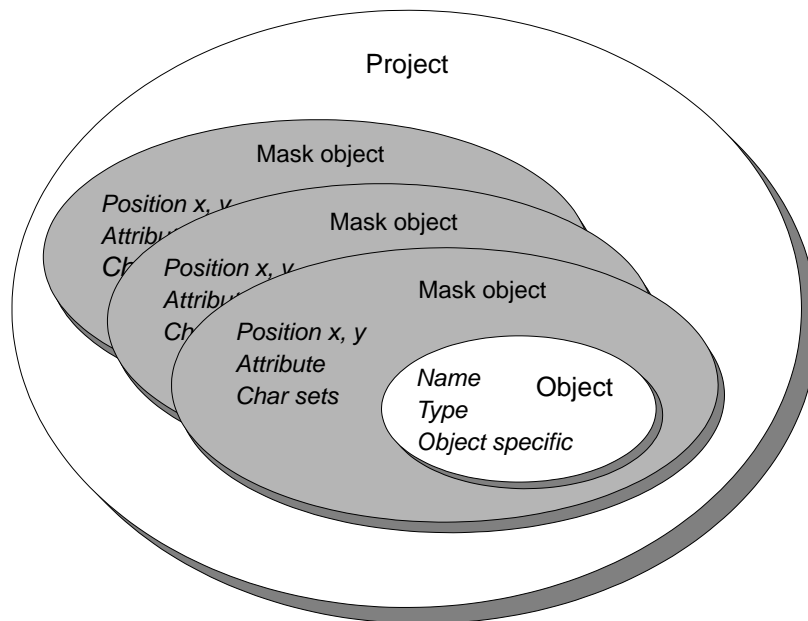


BDT editor

Data structures

An object-oriented carrier is used for the data holding. Objects such as variables, texts or pictograms are used as basis. The elementary information is saved in them (for 'Texts' it is the text itself).

In the next level, these objects are filled with representation information, such as position, attribute, character set etc. and designated as mask objects.



15.1.5 Short overview

There are two possibilities to start the BDT editor:

1. with the left mouse button click twice shortly one after the other on the symbol of the BDT editor. It is located generally in the program group window BDT tools.
2. Set the mark with the mouse or the arrow keys on the program symbol and confirm with the Return key.

The BDT editor always starts in the full screen view. Other applications are hidden, but not ended.

Determine concept

The following manual gives support for the first creation of a operation device application. It shows how a new project (project name: MIXER) is created and extended step by step to a small application.

 **Before the beginning of the work with BDT editor, the concept for the operation should be defined and possibly sketched.**


BDT editor

Create new project


- ★ Create via menu 'Projekt' (Project), Command 'Neu' (New) a new project.
- ★ In the dialog window 'Neues Projekt' (New project) enter the following:
 - Project name (here: MISCHER) according to the DOS conventions for file and directory name
 - select device type
 - further entries are optional

Create objects required for the solution of a problem

- ★ Open via the menu 'Objekt' (Object), Command 'Pictogramm' dialog window in which the pictograms used so far in this project are listed.
- ★ To add further pictograms to this list, click on 'Neu' (New). The existing BMP files are displayed in the dialog window. The reference to the file can be added to the project.

 **The BMP file is not copied into the project data base. The computer only remembers the reference to the BMP file. This has to be taken into account when the project is copied onto another computer, i.e. the BMP files must also be copied.**

- ★ Via the menu 'Objekt' (Object), Command 'Text' open dialog window, in which the texts used so far in this project are listed.
- ★ After clicking on 'Neu' (New) in a further dialog window, enter new texts.
- ★ Via the menu 'Objekt' (Object), Command 'Variable' open dialog window, in which the variables used so far in this project are listed.
- ★ After clicking on 'Neu' (New) create new variables. Depending on the adjusted device type, another dialog window is opened.

 **It is possible to create or define texts and variables in easy text files and adopt them via import into the project. The respective format must be absolutely kept. The easiest way to obtain the format is to export texts or variables from an existing project.**

BDT editor

Open mask

- ★ Via the menu 'Maske' (Mask), Command 'Maske bearbeiten' ('Edit mask'), open dialog window 'Masken' (Masks). A list with the masks existing so far is displayed.
- ★ After clicking on 'Neu' (New) or 'Bearbeiten' (Edit), create new masks or edit selected masks. 'Öffnen' (Open) opens a window, in which the mask content can be edited.

Represent mask objects

- ★ Via the menu 'Masken' (Masks), Command 'Hinzufügen' (Add), dialog window 'Masken' (Masks) represent objects provided with attributes (keys, pictogram, Text, text list, variable) on the display.

The entries depend on the type. A special form of mask objects are the keys. They are used for the creation of menu structures or for directyl setting and resetting variables in the control unit. The keys that can be programmed are available in a list. The keys that are already programmed are marked with a mark. A corresponding pictogram can be assigned and positioned in the display.

Generate project

- ★ Via the menu 'Maske' (Mask), Command 'Generierung' (Generate) create a special application file.

During this generation, a dialog window shows the current state of the generation with a progress bar.

- ★ After a faultless generation of the file, via the menu 'Extras', Command 'Download' load file into the connected operator device.

After the correctly completed loading, the operator device can be used.

BDT editor

15.2 Menus and dialog windows

The BDT editor makes its commands available via the menu line. It is adapted to the corresponding program state. It means that the full function extent is available only when the project is opened.

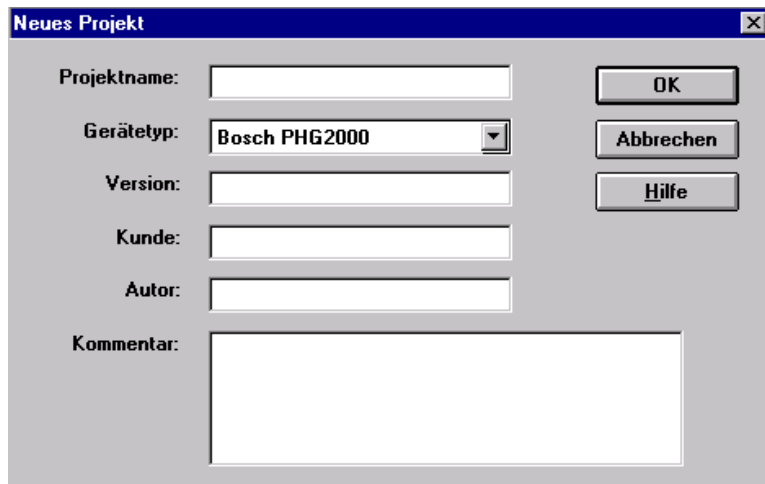
15.2.1 Menu Projekt (Project)



Command Neu (New) (Menu Projekt)

Via 'Neu' (New) a new project is created. The BDT editor interrogates in a dialog window the project description data.

The optional entry of the description data is useful since this information is displayed in the Open dialog and can be very helpful for the exact identification of the project.



BDT editor

The following entries must be made:

- Project name
- Definition of the device type

It is selected either from the given list or adopted by confirming the pre-setting.

Projektname (Project name) for entering the project name

The project name represents the directory which is used on the hard disk to save the data.

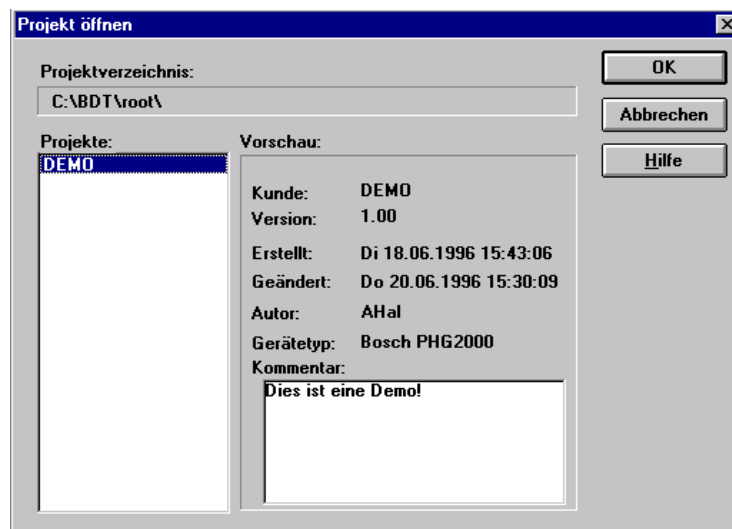
Gerätetyp (Device type) adjusts indirectly the specific physical characteristics of the target group, such as display size, key number, key arrangement or communication module.

Version contains a version number that can be freely defined by the project engineer for the administration and recognition of e.g. the latest project state.

Kunde (Customer),
Autor (Author),
Kommentar (Comment) are intended for any texts that can be used for the documentation of the project and the administration of the projects

Command Öffnen (Open) (Menu Project)

Via 'Öffnen' an existing project is selected to be edited. The BDT editor makes in a preview the project description available as support.

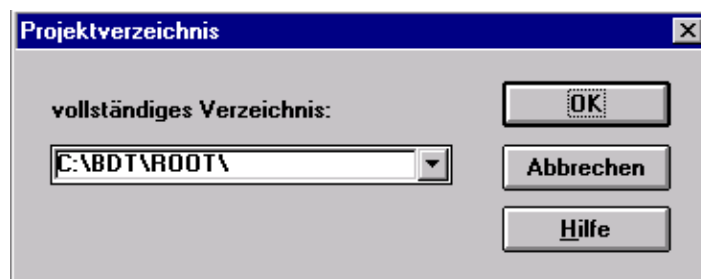


BDT editor

Projektverzeichnis (project directory)	shows the project directory currently set (Standard: working directory + ' \root')
Projekte (Projects)	displays the project names of all projects existing in the current project directory for selection
Vorschau (Preview)	shows the information from the project description of the selected project

Command Projektverzeichnis (project directory) (Menu Project)

With this command, the project directory is adjusted or changed. The project directory is the directory in which all created projects are gathered. As standard, this directory is located on the working directory + ' \root'. If it is changed, the set directory remains valid when the BDT editor is left.



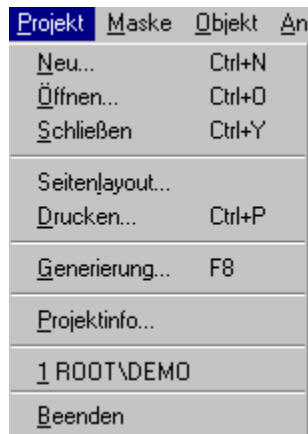
From the existing directories, an appropriate project directory can be selected. Network drives are also allowed. If a new project directory is required, a new directory must be first created.

Command Beenden (Exit) (Menu Project)

The program is left via Beenden (Exit). The BDT editor asks if unsaved modifications are to be saved.

BDT editor

15.2.2 <Projekt: Projektname> (Project: Project name) Menu Project



Command Neu (New) (<Project: Projectname> Menu Project)

Detailed information, see 15–9.

Command Öffnen (Open) (<Project: Projectname> Menu Project)

Detailed information, see 15–10.

Command Schließen (Close) (<Project: Projectname> Menu Project)

Via 'Schließen' (Close), an opened project is closed. The BDT editor asks if unsaved modifications are to be saved.

Command Seitenlayout (Page layout) (<Project: Projectname> Menu Project)

The parameters of a print page are adjusted via page layout, e.g margins, header and footer.

Spacers for special functions in headers and footers are possible:

\l	left flush
\c	centered
\r	right flush
\f	project name
\p	current page number
\t	system time
\d	system date

BDT editor

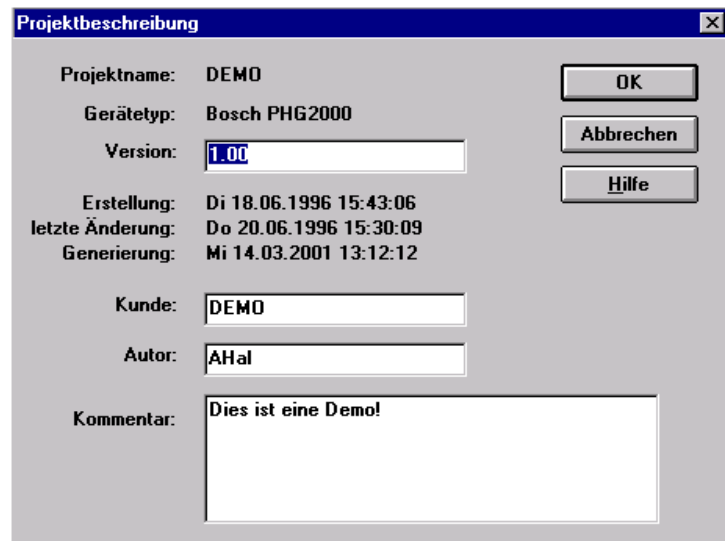
Command 'Generierung' (Generate) (<Project: Projectname> Menu Project)

Via 'Generierung' (Generate), an image file for the permanent memory of the operator device is created. A file 'Projectname.FLA' is created from the project data base. (Overview of formal relations, see figure 'Creation of the data administration', page 15-4).

Errors are displayed in the dialog window above the progress bar. The protocole file 'Projectname.ERR' gives information on location and type of the error. It can be displayed via the command 'Protocole-File' in the menu 'Extras'.

Command Projektinfo (Project info) (<Project: Projectname> Menu Project)

Via Projektinfo, the project description is entered or modified. Optional description data can be entered. Absolutely necessary data has been entered when the data base was created and cannot be changed any longer, see also page 15-10.



The screenshot shows a dialog box titled 'Projektbeschreibung' with a close button (X) in the top right corner. The dialog contains the following fields and buttons:

Projektname:	DEMO	OK
Gerätetyp:	Bosch PHG2000	Abbrechen
Version:	<input type="text" value="1.00"/>	Hilfe
Erstellung:	Di 18.06.1996 15:43:06	
letzte Änderung:	Do 20.06.1996 15:30:09	
Generierung:	Mi 14.03.2001 13:12:12	
Kunde:	<input type="text" value="DEMO"/>	
Autor:	<input type="text" value="AHal"/>	
Kommentar:	<input type="text" value="Dies ist eine Demo!"/>	

BDT editor

15.2.3 Menu Mask



Edit Mask (Menu Mask)

Via 'Edit Mask', a list of all masks is displayed in a dialog window.

A mask can be:

- created
- opened to be edited
- changed in its properties
- removed from the project



Selection list (Selection list) Name shows the masks with name already contained in the project

Öffnen (Button Open) To open a mask

Neu (Button New) To add a new mask

The dialog window 'Maske neu' (Mask new) is opened

Bearbeiten (Button Edit) To edit a mask

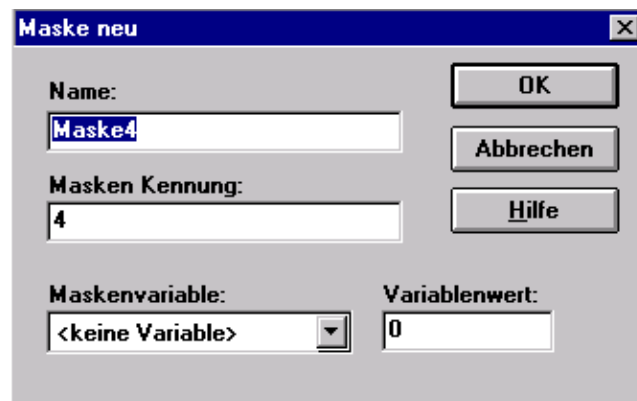
The dialog window 'Maske bearbeiten' (Edit mask) is opened

Löschen (Button Delete) Removes the selected mask from the project (works only when all masks are saved or closed)

BDT editor

Öffnen/Maskenname (Button Open/Mask name)

A selected mask is opened with the button 'Öffnen' (Open), i.e. represented in a separated window.

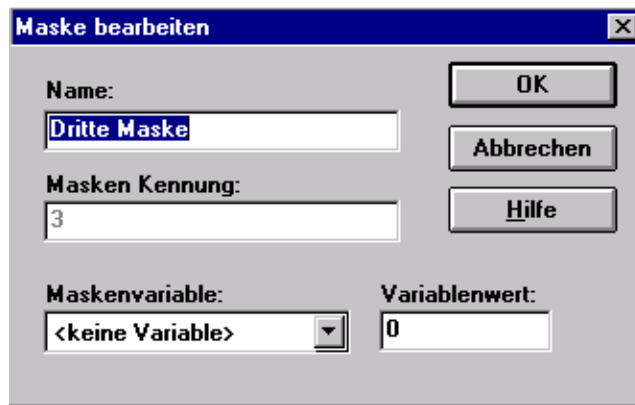
**Button Neu/Maske neu (New/Mask new)**

The dialog fields are preallocated with standard values. The mask name should be replaced through a unequivocal name, the given mask identification kept.

If the protocole allows it, the operator device can send a variable with a constant value when the mask is displayed. If you wish to use the possibility, you can select a variable already existing and a variable value.

BDT editor

Button Neu/Maske bearbeiten (New/Edit mask)



It is no longer possible to change the mask identification. Otherwise the same applies as for 'Button Neu/Maske neu (New/Mask new)', page 15-15.

Command Close mask (Menu Mask)

A mask or the mask window is closed via 'Maske schließen' (Close mask). The BDT editor asks if the unsaved modifications are to be saved.

15.2.4 <Project: Projectname> Menu Maske (Mask)




Command Maske speichern (Save mask) (<Project: Projectname> Menu Mask)

Via 'Maske speichern' (Save mask) the unsaved content of a mask is saved.

BDT editor

Command Alle Maske speichern (Save all masks) (<Project: Projectname> Menu Mask)

Via 'Alle Maske speichern' (Save all masks), the unsaved content of all opened masks is saved. This is required to delete e.g. all objects from the project data base.

 **The deleting of objects from the project data base is only possible when all masks are saved or closed. Only then an unequivocal administration of the objects is possible at this time.**

Command Maske neu zeichnen (Re-draw mask) (<Project: Projectname> Menu Mask)

Via 'Maske neu zeichnen' (Re-draw mask), the mask content can be updated.

This command can be e.g. used when:

- fragments of an object have not been drawn because of faulty functions of the program or Windows
- it is not clear if the mask content is correctly represented.

The mask content is written into the data base as it looks like after executing this command.

Add command (<Project: Projectname> Menu Mask)

Via 'Hinzufügen' (Add), a submenu is opened in which the following commands can be selected.



Taste... (Key)	Overview of a list of all keys that can be programmed in the dialog window 'Keys <Mask name>'
Pictogramm... (Pictogram)	opens the dialog window 'Pictogram' to add a pictogram to the mask
Linie... (Line)	adds a line to the mask
Text...	opens the dialog window 'Text' to add a text to the mask
Textliste... (Text list)	adds a text list to the mask
Variable...	adds a variable to the mask

BDT editor

Taste... (Key)

Via 'Taste' (Key), a list of all keys that can be programmed is displayed in the dialog window keys <Mask name>. The name of the mask in which the key is defined appears in the window title line.



Selection list

belegt Taste

(Selection list

assigned Key)

Overview on the keys that are already allocated

Button Bearbeiten
Edit

opens the dialog window key - <Name> - <Level>

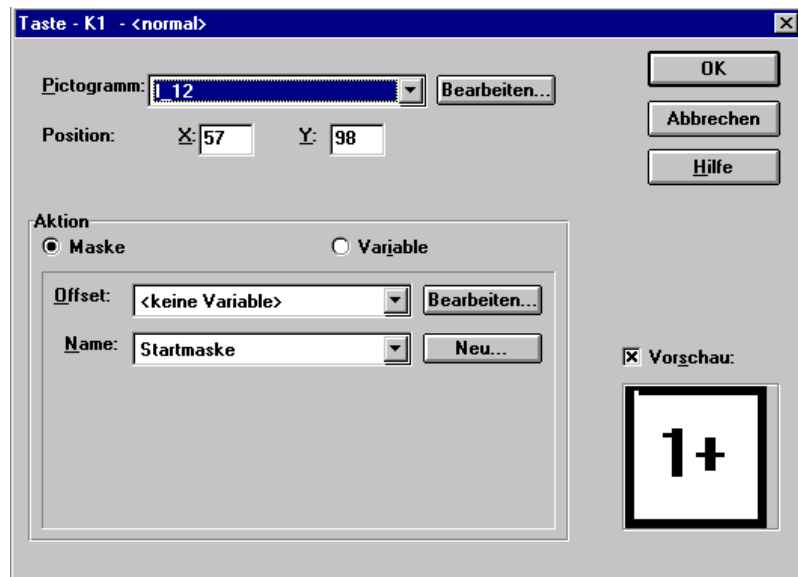
Button

Belegung löschen
Delete assigne-
ment

deletes the assignement

In the window title line, the key name and depending on the adjusted device type, the level (e.g. <Shift>, <Alt>, <Shift+Alt>) are displayed. An action can be allocated to the selected key in this dialog window.

BDT editor

Button Bearbeiten/Taste (Edit/Key) – <Name> – <Ebene> (Level)

Dialog field
Pictogram

A pictogram can be linked with every key that can be programmed, which is shown in the display at the adjustable position

If no pictogram is agreed on or if the correct one is not contained in the selection list, a pictogram can be created and selected by the button 'Bearbeiten' (Edit).

The position of the pictogram in the display can be adjusted or shifted in the mask window with the mouse.

Aktion Maske
(Action Mask)

branched in a mask, the name of which is to be indicated

Through the allocation of an offset variable, up to four masks can be called in dependence on the offset variable.

Aktion Variable
(Action Variable)

allocates to a variable a constant value for pressing and a value for releasing

This value is sent via the defined variable when the operator device is operated correspondingly.

Vorschau
(Preview)
(optional)

shows the mask and helps in this way to search for the correct mask

Buttons Bear-
beiten und Neu
(Edit and New)

gives the possibility to define new variables or masks

BDT editor

Pictogramm...

Via 'Pictogramm' the dialog window Pictogram is opened. It is possible to add a pictogram to the mask.



Dialog field
Name

Selection box for pictograms

For pictograms to appear in this list, they must be made known to the project, which occurs either through the command 'Pictogramm' in the menu 'Objekt' (Object) (see page 15–26) or via 'Neu' (New) in this dialog window.

Button Neu
(New)

see above

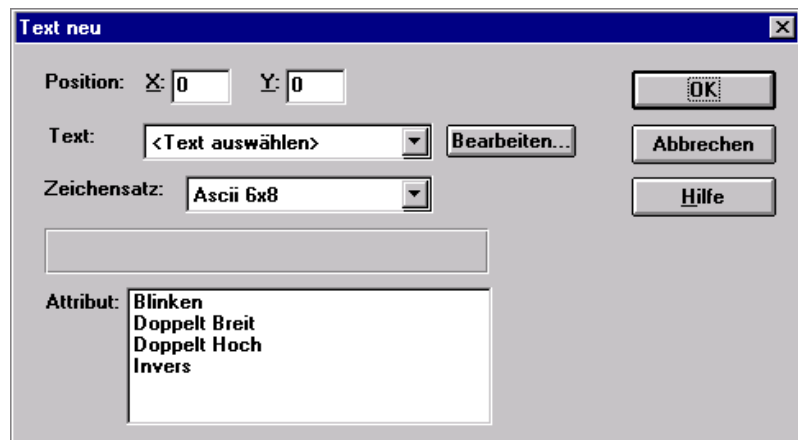
Vorschau
(Preview)
(optional)

shows the pictogram, which in this way helps to search for the right pictogram.

BDT editor

Text...

Via 'Text' the dialog window 'Text neu' (Text New) is opened to add a text to the mask.



Dialog field

Text

to select the text

Button

Bearbeiten
(Edit)opens the dialog window 'Texts' to create a text
(see also page 15–29)

The created or selected text can be provided with attributes such as Blinken, Invers, Doppelt Hoch (blink, inverse, double high) etc. and displayed, the character set can also be adjusted.



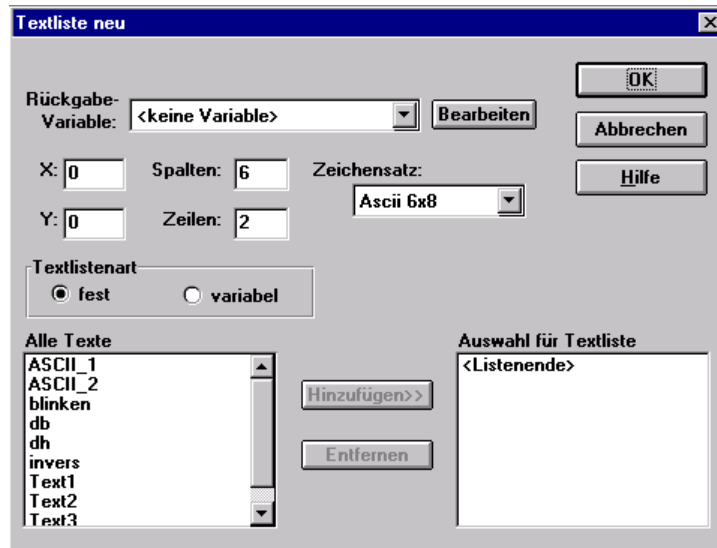
When entering the text, remember that only one-line texts can be represented in the operator device. In the character set Ascii6x8, a maximum of 40 characters suits in a line with the attribute Doppelt Breit only 20 characters.

BDT editor

Textliste... (Text list)

Via 'Textliste' a new text list is added to the mask. A text list is a list of texts that is displayed in a window field on the operator device.

With the help of the arrow keys of the operator device, one can go up and down in the text list and select an entry with the return key (cf. menu under Windows).



Rückgabe-
Variable
(return variable)

contains the number of the text entry of the list by confirming with the return key.

Zeichensatz
(Character
set)

To adjust the character set with which the text list is displayed

Textlistenart fest
(text list type fixed)

Content of the text list is defined from the dialog list 'All texts'

Procedure:

- select desired text
- via 'Hinzufügen' (add), transfer text in dialog list 'Selection for text list'

Wrong entries can be removed with 'Entfernen' (Delete). The order of the texts within the list can be selected freely.

Textlistenart variabel
(text list type variable)

can only be used in relation with variables from data type String (variable Text list)

Through this special variable, the content of the text list is defined via the communication interface by the control unit.

BDT editor

As the result of this dialog window, the text list is in both cases represented in the following form in the BDT editor:

```
#####  
#####  
#####  
#####
```

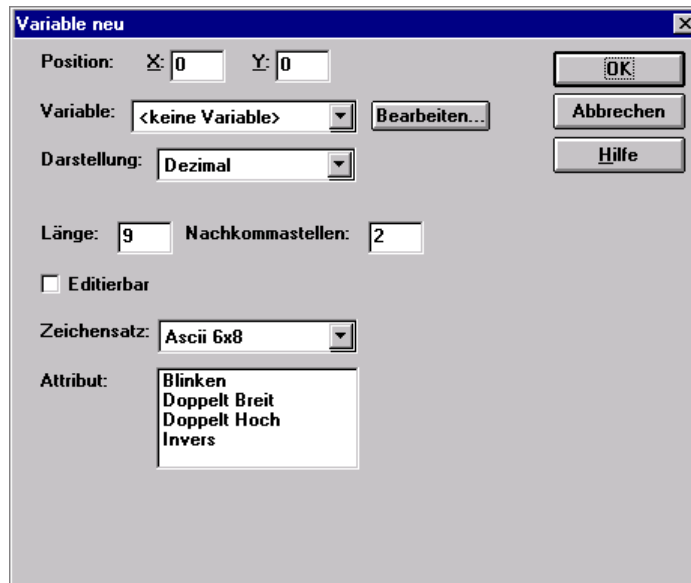
In the operator device, the spacers are filled with valid information.

Variable...

Via 'Variable', a new variable is displayed in the mask.

BDT editor

The type of representation can be selected. The variable properties depend on the selected type of representation.

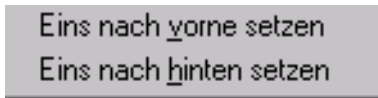


Position	for the exact indication of the position of the variable
Dialog field Variable	for selecting the variable
Button Bearbeiten Edit	opens dialog window 'Variable' to define the required variable.
Dialog field Darstellen (Represent)	to select the type of representation of the numerical value The following settings are possible: <ul style="list-style-type: none"> • Representation as a decimal number <ul style="list-style-type: none"> • Whole length of representation entered in the dialog field 'Länge' • Number of the after-decimal places in the dialog field 'Nachkommastellen' • Select the character set for the display of the variable as number • Select attributes (are combined with each other) • Representation as an hexadecimal number, settings, see representation as a decimal number • Representation as a binary number <ul style="list-style-type: none"> • Binary: values '0' and '1' are displayed • Binary (user-defined): value for 'Low' and 'High' can be defined • Representation as a string to display as string a variable which has been defined as type 'String'

BDT editor

Command Anordnen (Arrange) (<Project: Projectname> Menu Mask)

Via 'Anordnen' (Arrange), a selected mask object is shifted in the display list. In the sub-menu, objects placed one over the other can be set to the front or to the back.



Command Eigenschaften (Properties) (<Project: Projectname> Menu Mask)

Via 'Eigenschaften' (Properties), a selected mask object is processed. The corresponding dialog window is opened depending on the type of the mask object. In this dialog window, the modifications of the properties can be performed, see also section 15.2.5, Menu Object.

Command Ausschneiden (Cut) (<Project: Projectname> Menu Mask)

Via 'Ausschneiden', a selected mask object is copied into the clip board and deleted from the screen.

Command Kopieren (Copy) (<Project: Projectname> Menu Mask)

Via 'Kopieren' a selected mask object is copied into the clip board.

Command Einfügen (Paste) (<Project: Projectname> Menu Mask)

Via 'Einfügen', a mask object is inserted from the clip board into the current mask window.

Command Löschen (Delete) (<Project: Projectname> Menu Mask)

Via 'Löschen', a selected mask object is deleted from the screen.

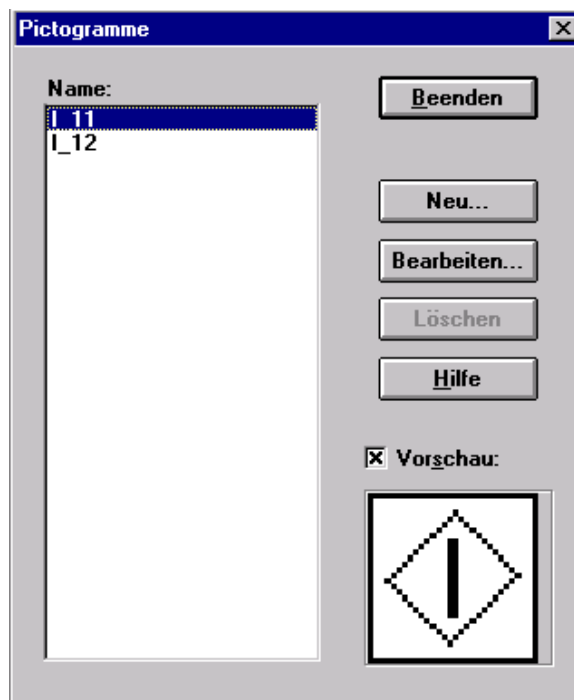
BDT editor

15.2.5 Menu Objekt (Object)



Command Pictogramm (Menu Objekt)

Via 'Pictogramm', a pictogram object is made known to the current project. The dialog window 'Pictograms' is displayed.



Dialog window Name shows a list of all pictograms already known

Button Neu (New) opens the dialog window 'Pictogramm neu'

Button Bearbeiten (Edit) opens the dialog window 'Pictogramm bearbeiten'

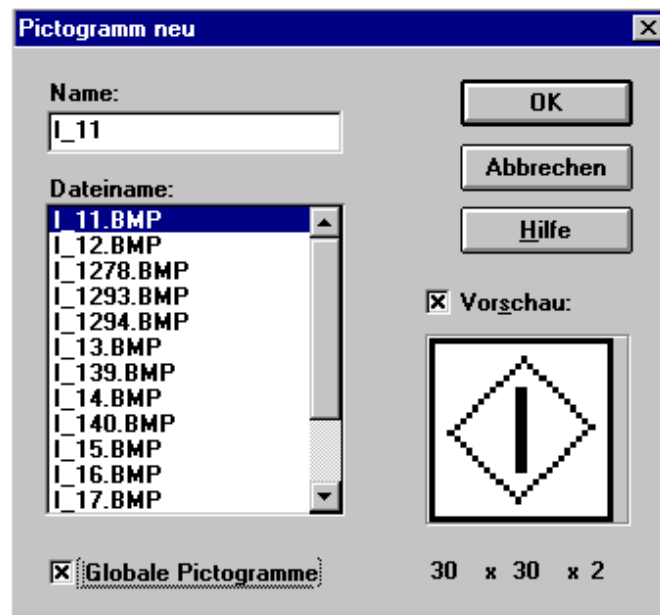
Vorschau (Preview) (optional) shows the pictogram and helps to search for a certain pictogram.

BDT editor

Button New/Dialog window 'Pictogramm New'

A pictogramm must be present as BMP file on the hard disk.

There are two directories in which it can be saved: first the directory 'BIT-MAP' for pictograms relating to several projects, second the project itself. A separation is obtained between the pictograms available for all projects und the pictograms used in the project.




Dialog window Name shows the name of the selected pictogram

Selection list Dateiname (file name) for the selection of a pictogram

Vorschau (Preview) (optional) shows the pictogram and helps to search for a certain pictogram

Pictograms must be present in the form of BMP files in the directory 'BITMAP' or in the project. BMP files that cannot be represented at the operator device from the size or the number of the used colors, are refused by the BDT editor. If the selected BMP file is not valid, this will be shown in the preview, no matter if the preview is switched on or off.

 **The information of the pictograms remain in the BMP files, they are not saved in the project data base. It is important that the BMP files should also be copied when a project is copied.**

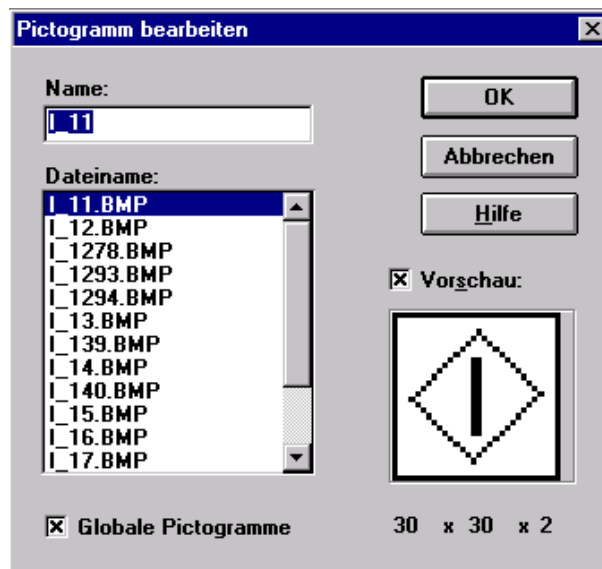
BDT editor

The format of BMP files is not interpreted in the same way by all programs. When the BMP format is a problem, the files must be loaded and saved in Paintbrush.

To delete a pictogram, all masks must be saved and closed.

The BDT editor checks before deleting if the pictogram is used in a project. Since a pictogram used cannot be deleted, the BDT editor issues an error message showing how often the pictogram is used in the project.

Button Bearbeiten (Edit) /Dialog window 'Pictogramm bearbeiten' (Edit pictogram)



Description, see button Neu (New)/Dialog window 'Pictogramm Neu', page 15–27.

Command Text (Menu Objekt)

Via 'Text', any texts can be entered.



BDT editor

Dialog window Texte (Texts)	shows selection list with texts already entered
Button Neu (New)	opens dialog window 'Text eingeben' (Enter texts)
Button Bearbeiten (Edit)	opens dialog window 'Text bearbeiten' (Edit texts)

Button Neu/Text eingeben (New/Enter text)

Dialog field Name	shows selection list with texts already entered
Dialog window Text	opens dialog window 'Text eingeben' (Enter text)
Text ID	Consecutive number of the text
Typ	Select text type, cannot be used at the moment

To delete a text object, all masks must be saved and closed.

The BDT editor checks before deleting if the text object is used in a project. Since a text object used cannot be deleted, the BDT editor issues an error message showing how often the text object is used in the project.

Pecularity

The text entry gives the possibility to insert variables by using format strings (e.g. %v...%). These format strings can be inserted at any place in the text.

- Variables can be inserted as numbers, e.g. voltage: 10.2 V

BDT editor

Syntax	Description
'voltage: %v4.1,Variable1% V'	The text entered appears in the display in the form 'Voltage: 99.9 V', where the number '9' is a spacer for the numerical value from the control unit that can be changed dynamically

The line has the following meaning:

%v	Introduction of the formatting
4.1	Variable length (Whole length with comma and decimal places)
Variable1	the variable corresponding to the text (On the operator device, the spacer is updated through the values supplied by the control unit for the variable)
%	Format end

- Variables can be inserted as a text from a list of texts depending on the variable value, e.g 'Motor: AN'

Syntax	Description
'Motor: %t3,Text1,2,Variable1%'	The text entered appears in the display in the form 'Motor: AAA'

The line has the following meaning:

%t	Introduction of the formatting
3	Length of this insert text
Text1	1st text that is represented (depending on variable1)
2	Number of the text (max. value of the variables: variable1)
Variable1	the variable representing the text
%	Format end

On the operator device, the spacer is updated through the values supplied by the control unit for the variable.

BDT editor

Button Bearbeiten/Text eingeben (Edit/Enter text)

The screenshot shows a dialog box titled "Text bearbeiten" with a close button (X) in the top right corner. The dialog contains the following elements:

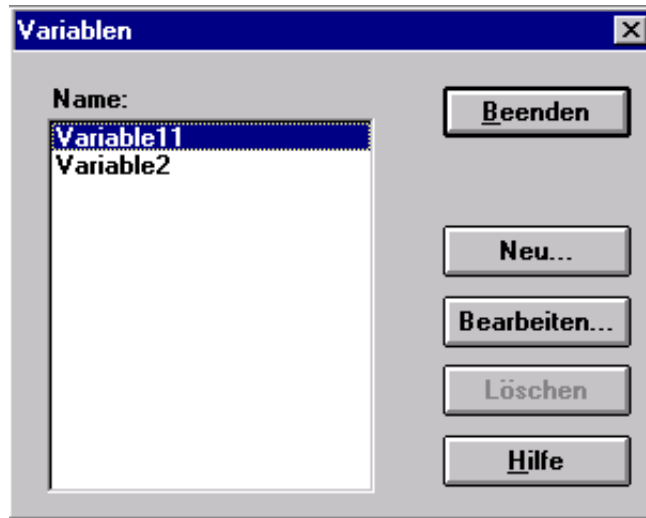
- Name:** A dropdown menu with "Text1" selected.
- Text:** A text area containing the text "Startmaske".
- Buttons:** "OK", "Abbrechen", and "Hilfe" are located on the right side of the dialog.
- Typ:** A group box containing three radio buttons: "Meldung" (selected), "Alarm", and "Text".
- Text ID:** A text input field containing the number "1".

It is no longer possible to change the text ID. Otherwise, the same applies as to Button Neu/Text neu', (new/text new), page 15–29.

BDT editor

Command Variable (Menu Object)

Via 'Variable', variables are defined which create the connection to the connected control unit.



Dialog window Name displays selection list with variables already defined

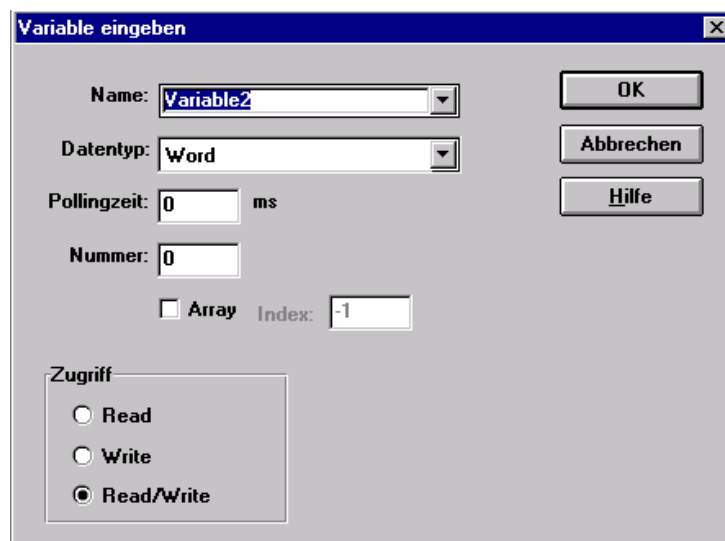
Button Neu (New) To add new variables

The dialog window 'Variable eingeben' (Enter variables) is opened.

Button Bearbeiten (Edit) To edit selected variables

The dialog window 'Variable eingeben' (Enter variables) is opened.

Dialog window Variable eingeben



BDT editor

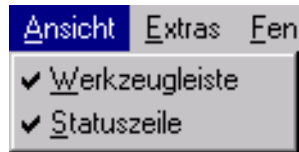
Dialog field Name	defines the name of the variable via which it is addressed in the project
Dialog field Datentyp (data type)	defines the value range of the variable that must match the definition of the control unit
Dialog field Pollingzeit (polling time)	shows the time with which the variable can be sent or requested. This function is not supported at this time by the operator device BIOS
Dialog field Nummer (number)	contains the number of the variable as it is designated by the control unit
Check box Array	must be activated when the array is of data type
Index	gives the position of the variable in the array
Dialog field Zugriff (access)	is required by the operator device so that it know how to access to the variable

Possible data types:

Bit	1	0 or 1
Byte	1	-127 .. +128
Byte (unsigned)	1	0 .. 255
Word	2	-32767 .. 32768
Word (unsigned)	2	0 .. 65535
Double	4	?
Double (unsigned)	4	?
Real	4	IEEE-Format
String	?	individual, as byte
String (variable Text list)	?	individual, as byte

BDT editor

15.2.6 Menu Ansicht



Command Werkzeuggeste (Tool bar) (Menu Ansicht (View))

Via 'Werkzeuggeste', a tool bar is shown or hidden. The tick before the command shows that the tool bar is shown.

Command Statuszeile (Status line) (Menu Ansicht (View))

Via 'Status line', a status line is shown or hidden. The tick before the command shows that the tool bar is displayed.

15.2.7 Menu Extras




Command Download (Menu Extras)

Via 'Download', the program BDT loader for programming or configuring the connected operator device is started. Information on the program BT loader, see page 15–39 ff.

Command Command Zeichenprogramm (Sign program) (Menu Extras)

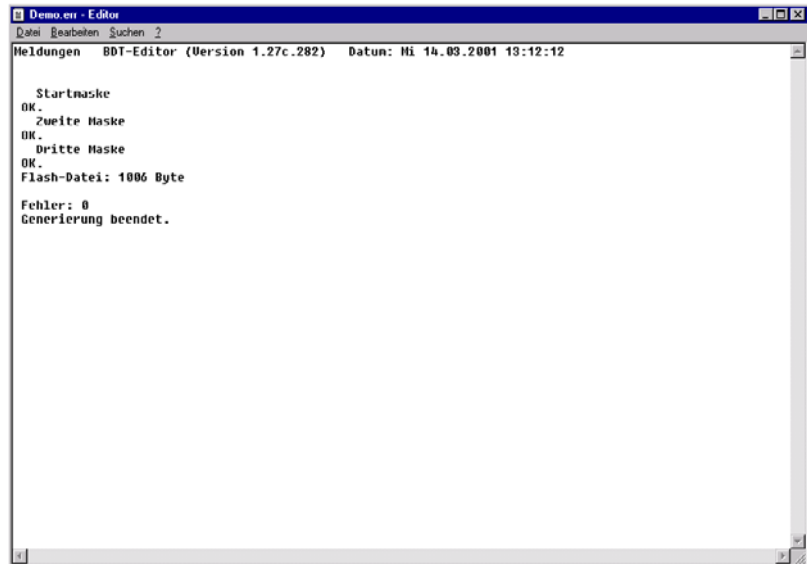
Via 'Sign program', a sign program is called for creating the pictograms. The pre-setting uses the program Paintbrush (part of the scope of the Windows delivery), see page 15–42.

 **We use the Paintbrush BMP format. If difficulties should occur when using other programs, the file must be read and saved in Paintbrush.**

BDT editor

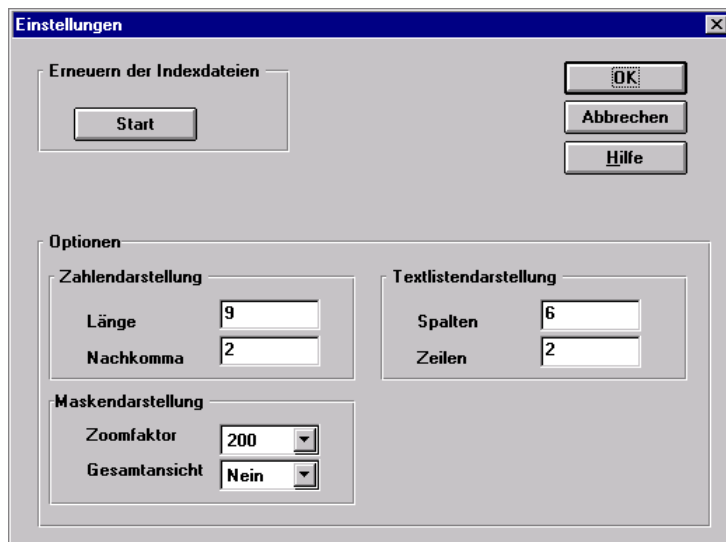
Command Command Protokolldatei (Protocole file) (Menu Extras)

Via 'Protokolldatei' (Protocole file), a file <Projectname>.ERR is displayed in the program NotePad (part of the Windows scope). <Projectname>.ERR contains error descriptions that have occurred during the generation (e.g. missing pictogram files *.BMP).




Command Einstellungen (Settings) (Menu Extras)

Via 'Einstellungen', some standard settings and indications must be set. The dialog window area 'Optionen' (Options) is used for this. These settings are administrated in the file 'bdtedit.ini'.

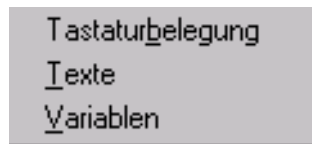


BDT editor

Dialog window area Optionen (Options)	Adjustment of standard settings and indications
Button Start	Renewal of all index files
	Internal connections are checked and corrected if necessary.
	 The renewal of the index files takes much time.

Command Import (Menu Extras)

Via 'Import', a submenu is opened. Different information of the object data base can be added.



'Tastaturbelegung' (keyboard assignment) updates the keyboard layout used in the project. The standard file selection window is called in which a text file '<Textdatei>.LAY' can be selected. This text file located in the work directory contains the description of the keyboard.

The following text file exists as standard:

PHG2000.LAY

This layout is fundamentally adopted as standard – depending on the device type setting.

After a change in the file '<Textdatei>.LAY' the project must be updated via 'Tastaturbelegung' (keyboard assignment). During the import, a plausibility check must be performed. The text file '<Textdatei>.LAY' can be edited with any text editor (e.g. NotePad). For detailed information on the design of the layout file, see comment line field at the beginning of the file.

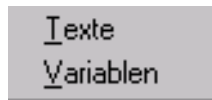
When importing texts or variables, the standard file selection window to select a text file (e.g. exptxt.txt or expvar.txt) is first displayed. After the selection of the file, the import begins.

 **The format of the text files is defined. The easiest way to obtain it is to export an existing project.**

BDT editor

Command Export (Menu Extras)

Via 'Export', different information can be obtained from the object data base.

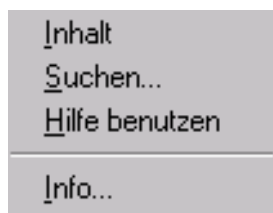


After actuating 'T_exte' or 'V_ariablen', the standard file selection window appears to select a target file.

After actuating 'OK', the operation starts and one gets the text file with the corresponding information. If the selected file already exists, the BDT editor asks if the file is to be overwritten.

At the beginning of the text file, comment lines are inserted from which the syntax rules follow.

15.2.8 Menu Hilfe (Help)



Command Inhalt (Content) (Menu Hilfe (Help))

Via 'Inhalt' (Contents), the contents of the context help file is displayed. The search terms and key words lead to the different topics.

Command Suchen (Search) (Menu Hilfe (Help))

Via 'Suchen' (Search), a dialog window is displayed in which the term to be searched can be entered.

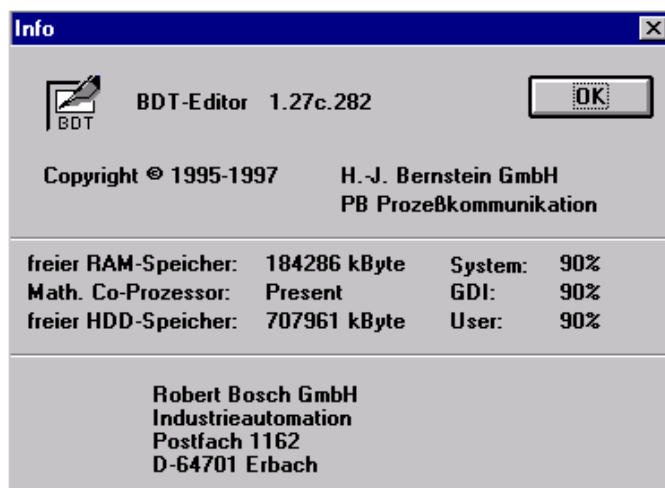
Command Hilfe benutzen (Use help) (Menu Hilfe (Help))

The help can be directly started via 'Use help'.

Command Info (Menu Hilfe)

Information on the product is displayed via 'Info'.

BDT editor



15.2.9 KontextMenu

The most important commands can be also obtained via context menus. Context menus are displayed when you click on the right mouse button. Depending on which screen position the mouse cursor is, another context menu appears.

BDT editor

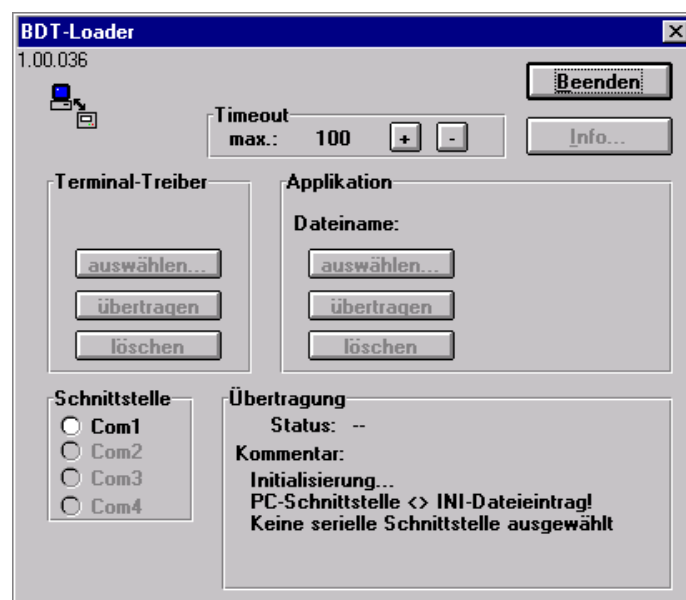
15.3 BDT loader

The BDT loader is an application based on dialog which cannot be reduced to the symbol. After starting the program, a serial interface is first assigned and configured. The settings are adapted to the operator device and cannot be changed by the user. When the interface is active, the BDT loader tries to recognize a connected operator device. If he does not recognize any, the program is ended.

There are two possibilities to start the BDT loader:

- with a double click on the program symbol in the menu Start.
- from the BDT editor via the menu 'Extras', Command 'Download'

15.3.1 Dialog window




A dialog window is displayed with which a selected image file *.FLA can be sent to the operator device.

The <Projectname>.FLA file is located in the project and has the same name as the project (project directory). The old content of the operator device is deleted before the loading of the new content.

Terminal driver

The first area of the BDT loader is the area 'Terminal-Treiber' (Terminal driver). This area is only activated when the directory 'DRIVER' exists, which allows access and when the directory is filled with files.

BDT editor

 **The drivers are for the working of the operator device as important as the application. The modifications – deleting and transferring of the drivers – should only be carried out with great care and by experienced users.**

If the access is allowed, operator device drivers can be loaded:

- ★ via button 'auswählen', select required operator device drivers.

A special dialog window is opened in which possible drivers are listed.

- ★ Adopt selection with 'OK' or reject with 'Abbrechen' (Cancel).
- ★ To delete the drivers present in the operator device, click on button 'löschen' (Delete).
- ★ Click on button 'Übertragen' (Transfer) to send the drivers selected to the operator device.

 **If the area is still engaged through old drivers, the transfer is interrupted. It is always necessary to first delete the drivers in the operator device. Simultaneous deleting and loading of the drivers, such as during the application transmission, does not work.**

Application

The second area of the BDT loader is overwritten with 'Application'. At this place, the created application is transferred to the operator device. In contrast to the area 'Terminal driver', the application in the operator device is deleted with 'Übertragen' before the new one is transmitted. The button 'löschen' (Delete) enables to delete the operator device content.

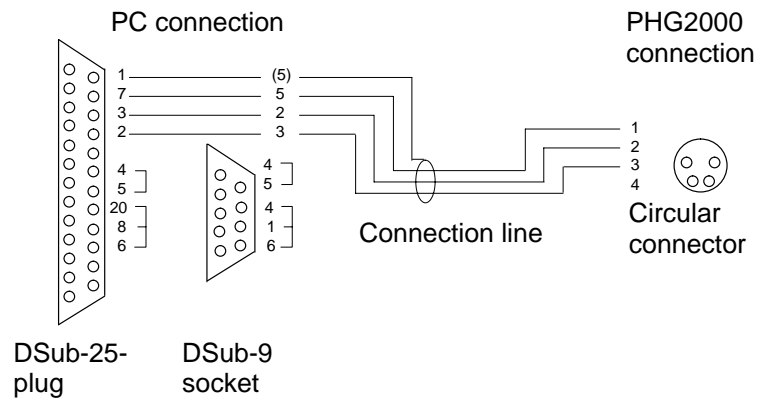
Transmission

The third area 'Übertragung' (transmission) is exclusively an output area. It gives information on the current activities. The status shows the current timeout counter. As a comment, additional messages on the process of the transmission are entered. Each line contains a message, the last line is the most current message of the operator device.

BDT editor

15.4 Technical data

In the PHG2000, importance is attached to a plug connection of type IP-65 at the operator device.



The technical data of the operator devices are to be found in the corresponding manuals.

15.5 Software, operator devices, accessories

Designation	Order No.
BDT-Editor with loader PHG2000, Projecting software	1070 918 447
PHG2000-Project loader	1070 918 675
Projecting cable for PHG2000	1070 918 674
PHG2000 operator device and accessories	1070 918 830
PHG-2000	1070 918 443
Carrying handle	1070 918 444
Holding device	1070 918 445
Folie english	1070 918 446
Manual for software (this description)	1070 072 421

BDT editor


15.6 Create pictograms with Paintbrush

The program Paintbrush [PBRUSH.EXE] is part of the Windows™ scope of delivery and is a drawing program for the creation and edition of pixel graphics.

The following information is thought as an introduction for the untrained user. For more detailed information, see Windows™ manual.

15.6.1 Screen size and attributes

- ★ Via the menu 'Options', Command 'Bild attributes' define the size of the working sheet and the number of colors.
- ★ Open new working sheet to adopt the screen attributes.


 **The limits of the operator device must be taken into account. Pictograms that are larger than the display section are not accepted by the BDT editor. The same applies to the adjusted colors. The information on the size of the display zone depends on the device type and can be found in the corresponding operator device manuals.**

15.6.2 Paintbrush as Pixel editor

- ★ To create exact pictograms with Paintbrush, via the menu 'View', Command 'Magnify' have display represented magnified.

It is then possible to edit each individual pixel.

- ★ Move the displayed frame with the mouse to the place of the screen that is to be magnified.
- ★ Edit represented screen section pixel by pixel by considering that the left mouse button creates a pixel in the foreground color and the right mouse button a pixel in the background color.
- ★ Leave magnification mode via the menu 'Ansicht', Command 'Reduce'.
- ★ Save pictogram as BMP file in global Bitmap directory or current project. Advantage of the global Bitmap directory: all projects can access to this pictogram.

 **The number of the colors must match the number of the colors of the operator device. The dialog window 'Save file as' gives the possibility in the dialog field 'File format' to adjust the number of the colors. One gets to the dialog window via the menu 'File', command 'Save as'.**

Appendix

A Appendix

A.1 Abbreviations

Abbreviation	Meaning
BAPS3	Programming language; Bewegungs- und Ablaufprogrammiersprache, Version 3; programming language
C:	Hard disk drive
CAN	Controler Area Network
DAC	Digital-analog converter
EEPROM	Electronically erasable programmable read-only memory
EGB	Elektrostatic sensitive components
ESD	Electrostatic discharge
LF	Line feed
MPP	Machine parameter program
MSD	Machine state display
PCL	Memory-programmable control
PE	Protective earth
PHG	Hand-held programming unit
POS	Actual position
PTP	Point to point
RC	Robot control
ROD	Incremental encoder
RPM	Rounds per minute
ROPS4	Robot programming system for rho4
TCP	Tool center point
WC	World coordinates

Appendix

A.2 Index**A**

A, V, D, 3–4
actual position, 7–3

B

basic level, 4–1
BDT editor, 2–1, 2–6, 3–1
buffer, 12–3

C

character numbers, 13–2
contrast setting, 2–7

D

Dateverwaltung, 15–4
direct selection, 4–3
display attributes, 6–3
display lock, 12–3
Documentation, 1–7
dump, 9–1

E

EMC Directive, 1–1
EMERGENCY–STOP devices, 1–5
error, 5–2
ESD
 Electrostatic discharge, 1–6
 grounding, 1–6
 workplace, 1–6
ESD–sensitive components, 1–6

F

file info, 3–6
Floppy disk drive, 1–7

G

Grounding bracelet, 1–6

H

Hard disk drive, 1–7

I

information, 5–1
interface, 2–4, 6–1
interface assignment, 4–5
 PHG functions via interface, 4–5

K

keys, 13–1
 indices, 13–5
 standard assignment, 13–1
kinematic info, 3–4

L

Low–Voltage Directive, 1–1

M

machine parameter, 2–4
mailbox, 6–1
masks, 2–6
menu tree, 4–1
Mode, 2–2
mode
 enlarged, 2–2
 PHG3 compatible, 2–2, 2–3
 transparent, 2–4
Modules sensitive to electrostatic discharge. See
 ESD–sensitive components

O

object, 10–2
operating system displays, 12–1
option bit, 4–3

P

parameter set, 8–1
permission key, 2–2
PHGKODE.DAT, 4–3, 12–2
positions, 3–3
priority, 5–6
process info, 3–6
process priority, 11–4

Q

Qualified personnel, 1–2

R

READ PHG, 12–3
Release, 1–8

S

Safety instructions, 1–4
Safety markings, 1–3
serial number, 8–1
signals, 3–2
Spare parts, 1–6
special function 16, 14–1
Standard operation, 1–1
standard signal assignment, 3–1
status message, 5–2
storage utilization, 10–5
sub–mode, 4–1
supervisor, 8–1
swivel arm robot, 8–1
system info, 3–5

Appendix

T

Test activities, 1–5

test mode, 2–7

Trademarks, 1–8

transparent mode, 2–4

- with a transfer rate of 19200 baud, 2–4

- with a transfer rate of 9600 baud, 2–4

- with existing menu structure, 2–4

- with user-oriented menu structure, 2–4

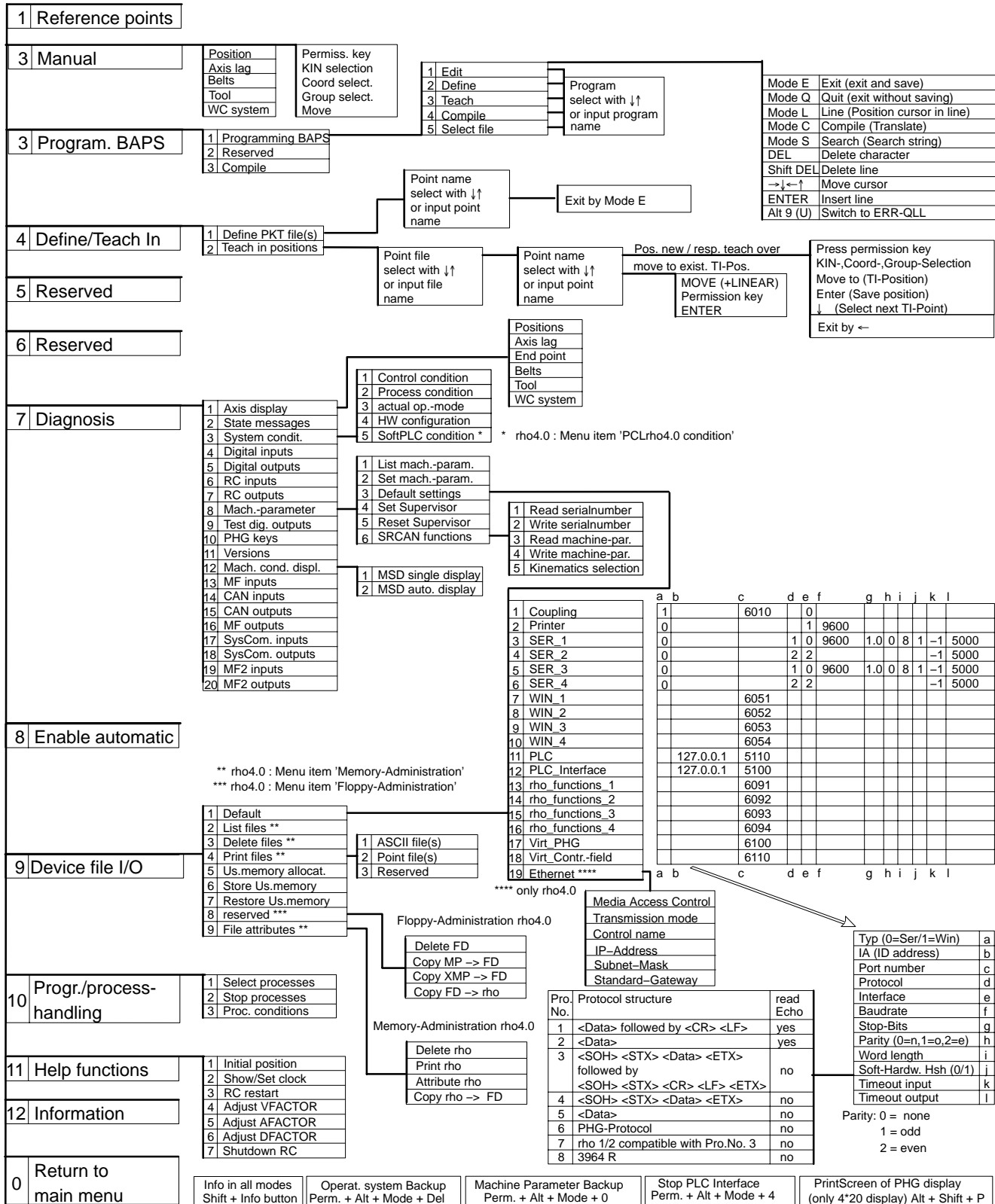
type plate, 8–1

V

variable text lists, 3–7

Appendix

A.3 PHG menu tree



Bosch Rexroth AG
Electric Drives and Controls
P.O. Box 13 57
97803 Lohr, Germany
Bgm.-Dr.-Nebel-Str. 2
97816 Lohr, Germany
Phone +49 (0)93 52-40-50 60
Fax +49 (0)93 52-40-49 41
service.svc@boschrexroth.de
www.boschrexroth.com

