

**Engineer<sup>IT</sup>**  
**Control Builder F**

# Engineering Manual

## System Configuration



**ABB**

### **Notice**

Information provided in this manual is subject to change without prior notice and represents no obligation on the part of ABB Automation Products.

The industrial standards and regulations (e.g. DIN, VDE, VDI, etc.) applicable in the Federal Republic of Germany are used. Outside the Federal Republic of Germany, the relevant national specifications, standards and regulations must be observed.

ABB Automation Products reserves all rights, especially those arising out of BGB, UWG, UrhG as well as out of industrial property rights (patents, utility models, trademarks, service trademarks and flavor samples).

The designations used and the products shown/mentioned in this manual have not been specifically marked regarding existing industrial property rights.

No part of this manual may be reproduced without prior written permission from ABB Automation Products.

Should you find any mistakes in this manual, please make a copy of the appropriate page(s) and send it/them to us with your comments. Any suggestions which may help to improve comprehension or clarity will also be gratefully accepted.

Please send your suggestions to:

Product Management Dept., DEAPR/LMS-Hannover, Fax: +49 (0)511 6782 701

# **Engineering Manual System Configuration**



## Engineering Manuals

### System Configuration

- A General Information
- B Installation DigiTool
- C Project Manager
- D Project Tree
- E Hardware Structure
- F Commissioning
- G Documentation
- Glossary
- New Features
- Index

### IEC 61131-3 Programming

- A General Information
- B Variables
- C Tags
- D Function Block Diagram (FBD)
- E Instruction List (IL)
- F Ladder Diagram (LD)
- G Sequential Function Chart (SFC)
- H User Function Blocks
- Index

### Operator Station

- A General Information
- B Messages and Hints
- C Standard Displays
- D Graphic Display
- E Logs
- Index

## Engineering Manuals

### Process Station – Rack based System

- 1 General
- 2 Loading the operating system and EPROM'S
- 3 Configuration of resource D-PS and D-PS/RED in project tree
- 4 Processing and failure action
- 5 Configuration of rack-based process station in the hardware structure
- 6 Commissioning the process station
- 7 Redundancy

### Process Station – FieldController

- 1 General
- 2 Loading the operating system and EPROM'S
- 3 Configuration of resource D-PS in project tree
- 4 Processing and failure action
- 5 Configuration of process station FC in the hardware structure
- 6 Commissioning of FC

### Process Station – ABB FieldController 800

- 1 General
- 2 Loading the operating system and EPROM'S
- 3 Configuration of resource D-PS and D-PS/RED in project tree
- 4 Processing and failure action
- 5 Configuration of process station AC 800F in the hardware structure
- 6 Commissioning of AC 800F
- 7 Redundancy

## Engineering Reference Manuals

### Functions and Function Blocks

A	Getting Started
B	General Description and Overview
C	Analog Function Blocks
D	Binary Function Blocks
E	Controller Function Blocks
F	Acquisition Blocks
G	Monitoring Function Blocks
H	Open-loop Control Function Blocks
J	Standard Function Blocks
K	Arithmetic Blocks
L	Converter Blocks
M	Constants
N	System Functions
V	Abbreviations
W	Glossary
X	Index

### Communications and Fieldbuses

A	Getting Started
B	General Description and Overview
C	Profibus
E	Modbus Master Function Blocks
F	Modbus Slave Function Blocks
G	Send / Receive Blocks
H	Rack Modules
V	Abbreviations
W	Glossary
X	Index

## Operators Manual

### Operator Station

A	Getting started: DigiVis
B	Installation DigiVis
C	Operating Philosophy
D	Messages and Hints
E	Overview Display
F	Group Display
G	Graphic Display
H	Trend Display
I	Faceplates
J	SFC Display
K	Time Scheduler Display
L	Logs
M	System Display
V	System Messages
W	Glossary
X	Index

## A General Information

A 1	Visual Orientation Hints .....	A-5
A 2	General Hints for Operating DigiTool .....	A-6
A 3	New Features, Version V6.1 .....	A-9

## B Installation DigiTool

B 1	General Notes .....	B-5
B 2	First installation of the DigiTool software .....	B-7
B 3	Upgrade / Update of an existing DigiTool Version to V6.....	B-18
B 4	Correcting the Installation.....	B-20
B 5	Installing the DigiTool Help System .....	B-22
B 6	Checking the Installation with DigiCheck.....	B-24

## C Project Manager

C 1	General Description - Project Manager .....	C-5
C 2	Starting the Project.....	C-6
C 3	Editing Projects .....	C-9
C 4	Project Management Options.....	C-21
C 5	Appendix - Toolbar-Buttons.....	C-26

## D Project Tree

D 1	General Description - Project Tree.....	D-5
D 2	Structure of the Project Tree .....	D-8
D 3	Configuring the Project Tree .....	D-16
D 4	Configuring the Project Objects.....	D-25
D 5	General Functions of the Project Tree .....	D-52
D 6	Project Tree Options .....	D-56

## E Hardware Structure

E 1	General Description - Hardware Structure .....	E-5
E 2	Hardware Structure User Interface.....	E-8
E 2	General Functions of the Hardware Structure.....	E-20
E 3	Editing the Hardware Structure .....	E-25
E 4	I/O Editor .....	E-36
E 5	Network Configuration.....	E-44
E 6	General Parameters of the Hardware Structure Objects .....	E-46

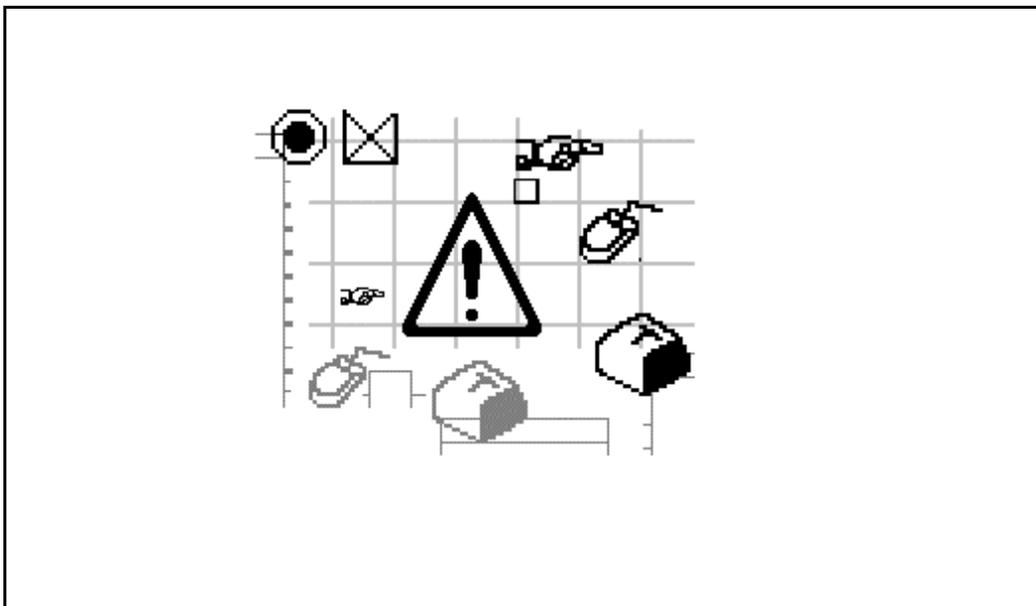
**F Commissioning**

F 1 General Description - Commissioning ..... F-5  
F 2 Starting Commissioning ..... F-7  
F 3 Commissioning User Interface ..... F-12  
F 4 Time Settings ..... F-22  
F 5 Loading the Project ..... F-25  
F 6 Changes with Side Effects on Other Project Objects ..... F-33  
F 7 Displaying and Writing Values in a Window ..... F-35

**G Documentation**

G 1 General Description - Documentation ..... G-5  
G 2 Compilation of the Drawing Header / Footer ..... G-7  
G 3 Define the Documentation Job ..... G-14  
G 4 Documentation Scope ..... G-35  
G 5 Outputting Documentation ..... G-40  
G 6 Documentation Manager ..... G-43  
G 7 Comment field editing ..... G-48  
G 8 Tables ..... G-50

## A General Information





## Contents

<b>A 1</b>	<b>Visual Orientation Hints</b> .....	<b>A-5</b>
<b>A 2</b>	<b>General Hints for Operating DigiTool</b> .....	<b>A-6</b>
A 2.1	Operation with mouse or keyboard.....	A-6
A 2.2	Recurring keys.....	A-7
<b>A 3</b>	<b>New Features, Version V6.1</b> .....	<b>A-9</b>
A 3.1	FieldController Redundancy .....	A-9
A 3.2	Templates Module .....	A-9
A 3.3	Hart Support .....	A-9
A 3.4	FDT / DTM-Support (Field Device Tool / Device Type Manager) .....	A-9
A 3.5	Maestro NT Coupling.....	A-10
A 3.6	Flexible System Limits.....	A-10
A 3.7	Batch.....	A-11
A 3.8	Trend Server.....	A-11
A 3.9	True Color Screen Resolution .....	A-12
A 3.10	DigiTool.....	A-13
A 3.10.1	Scalable Window Size in DigiTool .....	A-13
A 3.10.2	Moveable Toolbars .....	A-13
A 3.10.3	Loading of Selected Objects.....	A-13
A 3.10.4	Display of the Redundancy Memory.....	A-13
A 3.10.5	Greater Number of Logs .....	A-14
A 3.10.6	Variables and Tag Lists .....	A-15
A 3.10.7	Gateway Write Permission .....	A-16
A 3.10.8	Project tree .....	A-17
A 3.10.9	Boot Parameters.....	A-18
A 3.10.10	Graphics .....	A-18
A 3.11	DigiVis.....	A-19
A 3.11.1	Acknowledgement .....	A-19
A 3.11.2	Selection in the Graphic Display.....	A-19
A 3.12	DigiBrowse.....	A-19
A 3.12.1	Print .....	A-19
A 3.12.2	Command Line for Batch Files .....	A-19



## A 1 Visual Orientation Hints

To grant direct access to information, we have used different types of scripts and symbols.

### Script                      Meaning

*Italics*                      Representation for (selectable) menu items or parameters.

SMALL CAPITALS              Inputs to be made via the keyboard, also via virtual keys.

**boldface**                      Highlights important information, also as an orientation hint.

### Symbols                      Meaning



Selection with mouse.

The various instruction steps are separated by arrows.

Example: → *Edit* → *Dimensions* →... .

In this example, the menu item *Edit* is to be selected followed by the menu item *Dimensions*.



Operating alternative with the mouse



Information on operation with the keyboard, inasmuch as it differs from the Windows Standard.

Example:     Select module → ALT → E → D →... .

Having selected the module via the ARROW KEYS, the keys ALT, E and D must be pressed successively.

If two keys are to be pressed simultaneously:

... → SHIFT + INSERT →... .



Alternative keyboard operation



Hints



Special hints, must be observed!

Preconditions                      Preconditions which must be fulfilled to implement commands or for satisfactory results.

## A 2 General Hints for Operating DigiTool

Operation of DigiTool is based on the Windows Standard.

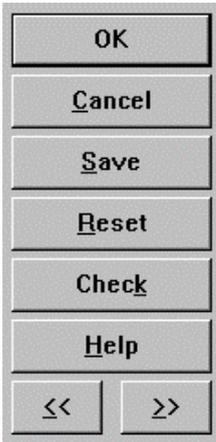
Therefore knowledge of general operation under Windows is strongly recommended, see Windows Manual.

The "typical Windows operation" will therefore not be dealt with in detail when describing the various editors.

### A 2.1 Operation with mouse or keyboard

	Mouse 	Keyboard 
Select menu items	Cursor on menu item + left mouse button.	ALT + underlined letters
Select within pull-down menu	Cursor on menu item + left mouse button.	Enter only underline letters
Select individual elements	Cursor on program element + left mouse button	Cursor on program element + SPACE
Select multiple elements	Cursor on start position <ul style="list-style-type: none"> <li>• press left mouse button and keep pressed down</li> <li>• move to desired position and release mouse button</li> </ul>	Cursor on start position <ul style="list-style-type: none"> <li>• Press SPACE and hold</li> <li>• move to desired position and release SPACE</li> </ul>

## A 2.2 Recurring keys



di0212uk.bmp

OK	The active parameter window is quit and the parameter status <b>saved</b> .
CANCEL	The active parameter window is quit <b>without saving</b> the parameter status. A warning appears if parameter definition data are lost.
SAVE	The current parameter status is <b>saved</b> and the window remains active.
RESET	The parameters of the active parameter window are reset completely to the previously <b>set values</b> . Any parameters previously saved and differing from the default settings can be fetched again by canceling and recalling the parameter window.
CHECK	The function block is checked for plausibility with the current parameters, even if they have not been saved.
HELP	Call up the Windows On-line help (with F1 also). To get information about functions on monitor without using the documentation. Thematically structured information is displayed in the help-window.

- <<                    Change to the previous  
>>                    or next parameter mask.  
This is displayed with shading if no further parameter window exist.
- Check boxes  
A setting or parameter is turned on or off.
- Option fields  
Option fields are presented when **one** of a group of mutually  
parameters are to be chosen.
- F5                    The function key F5 calls the list of cross references for the selected vari-  
able or tag. This function is also available in parameter mask fields with  
referenced variables or tags.
- F6 bzw. SHIFT+F6                    These functions are available after the list of all existing cross references  
was called (key F5). F6 calls the program which contains the next occur-  
rence in the cross reference list, SHIFT+F6 calls the program of the previ-  
ous list entry.

## A 3 New Features, Version V6.1

### A 3.1 FieldController Redundancy

The ABB FieldController 800 can now be operated in redundancy.

It is possible to set up the following redundancies:

- Master redundancy
- Profibus Line redundancy (a new hardware module, the **Redundancy Link Module RLM 01** is available for setting up this redundancy)
- Component redundancy

See the **Engineering Manual, Process Station, ABB FieldController 800, Redundancy**

### A 3.2 Templates Module

For modular slaves (Remote I/O), templates are available at the **module level**.

Templates can now also be compiled by the re-using of template parts, with Export/Import.

See the **Engineering Reference Manual, Communications and Fieldbusses, Profibus**

### A 3.3 Hart Support

HART protocol-capable devices can now be linked to the FieldController through Profibus.

The HART parameters are made available through the DPV1 parameters.

See the **Engineering Reference Manual, Communications and Fieldbusses, HART Protocol** in the.

### A 3.4 FDT / DTM-Support (Field Device Tool / Device Type Manager)

Devices which provide the FDT Interface can be linked to the FieldController. The requirements and configuration are described in the **Engineering Reference Manual, Communications and Fieldbusses, FDT** and in the **Engineering Manual, Process Station, FieldController**.

### A 3.5 Maestro NT Coupling

There is now a freely configurable interface available under Windows, as an operation and observation interface. The coupling of the Process Stations and the FieldController (real-time process values) is done through an OPC Gateway.

The operation and observation is furthermore possible with DigiVis.

Also see the **Manual Integration with Maestro NT**.

### A 3.6 Flexible System Limits

The number of the stations in a system is no longer restricted to the previous limits.

The only restriction is that up to 10 communications, that is 10 **active stations** can be linked to **one** process station.

The configuration of the connections is done in the CONF nodes of the project tree.

	GW01	GW02	LS01	LS02	LS03
FC01 (3/10)	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
PSRE (2/10)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
PS01 (4/10)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
PS02 (3/10)	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

ta004us.bmp

By combining local operator stations (operator stations only for selected process stations) and central operator stations, a many-sided operation and observation concept is possible. See the **Engineering Manual, System configuration**.

### A 3.7 Batch

In addition to the previous batch package “DigiBatch” a new batch package, can also be installed. This is fully integrated into Maestro NT and offers additional functions such as

- Batch recording
- A sequence scheduler for the individual compiling of operations
- Materials management
- Batch server redundancy
- Trends and history acquisition
- Interface to Enterprise Systems (EPR)

Coupling to Maestro NT is done through a coupling module in **PhaseX** (similar to the PLI module of DigiBatch).

See the **Manual for the PhaseX Module** and the manuals for the **Operation and Configuration of the Batch Package**.

### A 3.8 Trend Server

Previously, the number of Trends per operator station was restricted to 42.

These limits are no longer valid. As many variables as desired can be acquired, all from the Variables List, and also variables from external sources. The trend acquisition module is necessary for these options. The data transfer is done through a Trend Gateway. This is configured in DigiTool as a node and also integrated in the hardware structure as a gateway.

The determination as to whether this should be transferred by trend acquisition module or gateway is done in the parameter mask of the trend display. The variables can also be given a time stamp.

Parameters: Trend Display TR\_D-05

General data

Name: TR2      Short text: Pressure

Long text: AF800 Variable-Acquisition

Acquisition | Display | Area options | Archive | File transfer

Use acquisition function block     Use variable acquisition

Tag name:

Sample time: T#10s

	Time stamp	Value
Variable 1:	PSRE.DateTime	AF800_Pressure
Variable 2:	<input type="text"/>	<input type="text"/>
Variable 3:	<input type="text"/>	<input type="text"/>
Variable 4:	<input type="text"/>	<input type="text"/>
Variable 5:	<input type="text"/>	<input type="text"/>
Variable 6:	<input type="text"/>	<input type="text"/>

OK    Cancel    Save    Reset    Check    Help

ta008us.bmp

See the **Engineering Manual, Operator Station, Trend display**.

### A 3.9 True Color Screen Resolution

Work can now be done with the True Color setting.  
 The screen resolution can be set to 1600 x 1200 pixels.  
 These settings are done in the system control.  
*System Control* → *Display* → *Settings*

The screen resolution is no longer asked for when setting up DigiTool and DigiVis.

## A 3.10 DigiTool

### A 3.10.1 Scalable Window Size in DigiTool

Previously, it was only possible to operate DigiTool in a full window. Now the size of the window can be adjusted and therefore viewed in parallel with other applications. The setting is done with the Windows minimize button and then re-sizing or positioning.

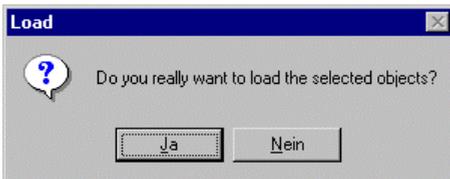
### A 3.10.2 Moveable Toolbars

In DigiTool, the toolbars can be moved from the top group-wise and can be arranged on the screen as desired.

The toolbars for selecting *configuration* and *start-up* are now by default in the first position in the toolbar strip. As a result a quick changeover is possible.

### A 3.10.3 Loading of Selected Objects

When loading selected objects, an additional box appears with a safety prompt.



ta001us.bmp

This additional prompt gives you the opportunity to reconsider the procedure and/or to undo an incorrect selection.

See the **Engineering manual, System Configuration, Start-up, Side Effects.**

### A 3.10.4 Display of the Redundancy Memory

By selecting objects in the project tree, it is possible to see the amount of the redundancy memory that is in use.

*System* → *Display all objects* or •

*System* → *Display selected objects*

### A 3.10.5 Greater Number of Logs

The records can now be archived in 400 files rather than the 100 previously. The memory required should be noted.

**Parameters: Signal sequence log SSLN**

General data

Name: SIG2      Short text: signal log

Long text: operator station 1

General | Log files | Format | File transfer

Start/stop

Automatic      Tag name of event function block:

Event controlled

Manual

Filing

In 400 files named LOGS1

With max. runtime of T#2h per file but not more than 1000 events.

Disk space requirements for this log: 454326 KByte

OK    Cancel    Save    Reset    Check    Help

ta002us.bmp

### A 3.10.6 Variables and Tag Lists

The configuration within the lists is now also possible with a context menu.

A quick selection of a Variable or Tag list can be done with *Search* → *Type ahead*, then entering the first letters of the name.

The screenshot shows the 'Variable list' window in DigiTool. The window title is 'Freelance 2000 DigiTool'. The menu bar includes 'Variable list', 'Search', 'Edit', 'System', 'Cross references!', 'Options', 'Back!', and 'Help'. The toolbar contains various icons for file operations and editing. The main area displays a table of variables with columns: Name, Comment, Type, Res., X, Object, Location, P, and Initial value. A 'Type ahead' dialog box is open over the table, with the text 'Variable name: T' and a 'Close' button. The table lists variables such as PSRE\_RAM\_Size, PSRE\_RadioCkAv, PSRE\_RedBufFlow, PSRE\_RedCPUFlow, PSRE\_RedCPUSlot, PSRE\_RedLinkLoad, PSRE\_RedState, PSRE\_ResState, PSRE\_SendErr, PSRE\_StationLoad, PSRE\_StationNo, PSRE\_StationType, PSRE\_UserStopped, Status\_I2F1, TF211\_TEMP, a01\_out, a1, e2, Flipflop\_q, hrn\_out, input, krit, lia01\_in, lia01\_s11, lia01\_s14, lia\_s11, p01\_out, pt100, and test10\_in.

Name	Comment	Type	Res.	X	Object	Location	P	Initial value
PSRE_RAM_Size		UDINT	PSRE	N			Y	
PSRE_RadioCkAv		BOOL	PSRE	N			Y	
PSRE_RedBufFlow		UDINT	PSRE	N			Y	
PSRE_RedCPUFlow		UDINT	PSRE	N			Y	
PSRE_RedCPUSlot		UDINT	PSRE	N			Y	
PSRE_RedLinkLoad		UDINT	PSRE	N			Y	
PSRE_RedState		UDINT	PSRE	N			Y	
PSRE_ResState		UDINT	PSRE	N			Y	
PSRE_SendErr		BOOL	PSRE	N			Y	
PSRE_StationLoad		UDINT	PSRE	N			Y	
PSRE_StationNo							Y	
PSRE_StationType							Y	
PSRE_UserStopped							Y	
Status_I2F1						C3_F3_2_Stationstatus	N	
TF211_TEMP						C3_F3_4_Value	Y	
a01_out		BOOL	PSRE	N			Y	
a1		REAL	PSRE	N			N	
e2		REAL	PSRE	N			N	
Flipflop_q		BOOL	PSRE	N			N	
hrn_out		BOOL	PSRE	N			N	
input		UDINT	FC01	N			Y	
krit		BOOL	PSRE	N			N	
lia01_in		REAL	PSRE	N			Y	
lia01_s11		BOOL	PSRE	N			N	
lia01_s14		BOOL	PSRE	N			N	
lia_s11		BOOL	PSRE	N			N	
p01_out		BOOL	PSRE	N			N	
pt100		INT	FC01	N	PROFI_S_DEU	FC3_F3__1_M7_IN_INT0	N	
test10_in		REAL	PSRE	N			N	

ta006us.bmp

Up to 10 pre-defined search criteria can be stored and be later recalled, e.g. with the Toolbars 1 to 10. It is also possible to select multiple search criteria (AND operations).

The screenshot shows the 'Freelance 2000 DigiTool' application window. The menu bar includes 'Tag list', 'Search', 'Edit', 'System', 'Cross references!', 'Options', 'Back!', and 'Help'. Below the menu is a toolbar with various icons. The main window displays a table with the following columns: Name, Type name, L, and P. The table contains two rows of data:

Name	Type name	L	P
TF211	H_ANA	S	@
TF211_FULL	PROFI_S_DEU	S	@

A search criteria dialog box is overlaid on the table, showing the following fields:

- Name: T\*
- T: \*
- Area: \*
- R: \*
- Type name: \*
- L: \*
- P: \*
- Access by gateway station: \*
- Show unused tags: N

At the bottom right of the window, it says '2 of 188 entries | NOLOCK'.

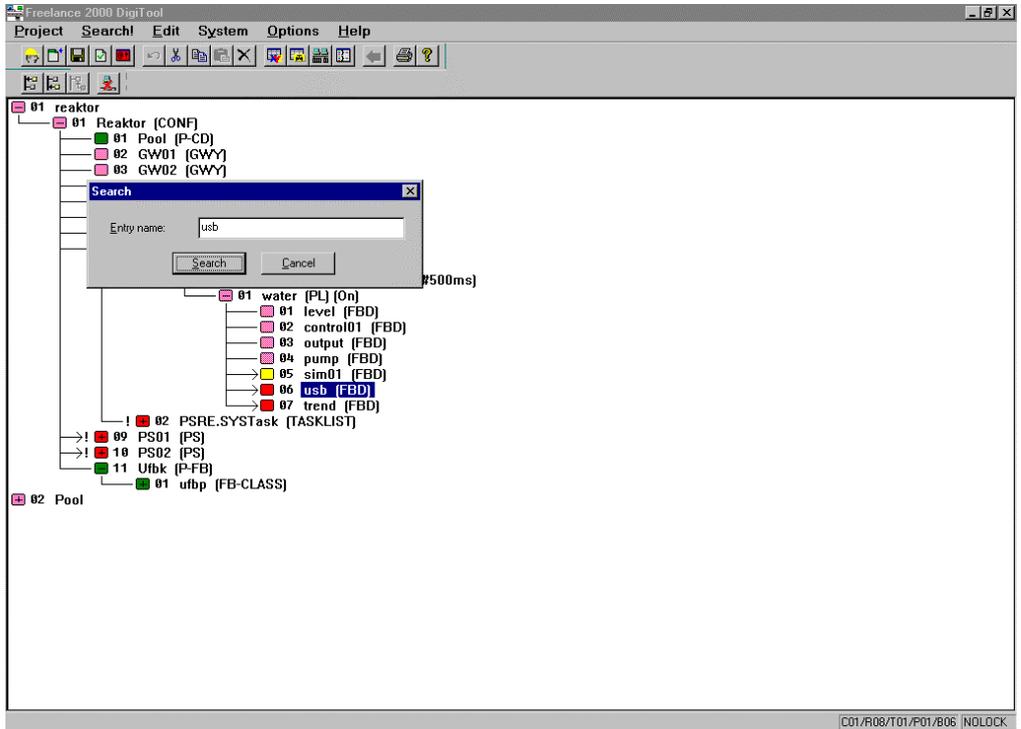
ta005us.bmp

### A 3.10.7 Gateway Write Permission

The write permission for a configured gateway is entered by default.

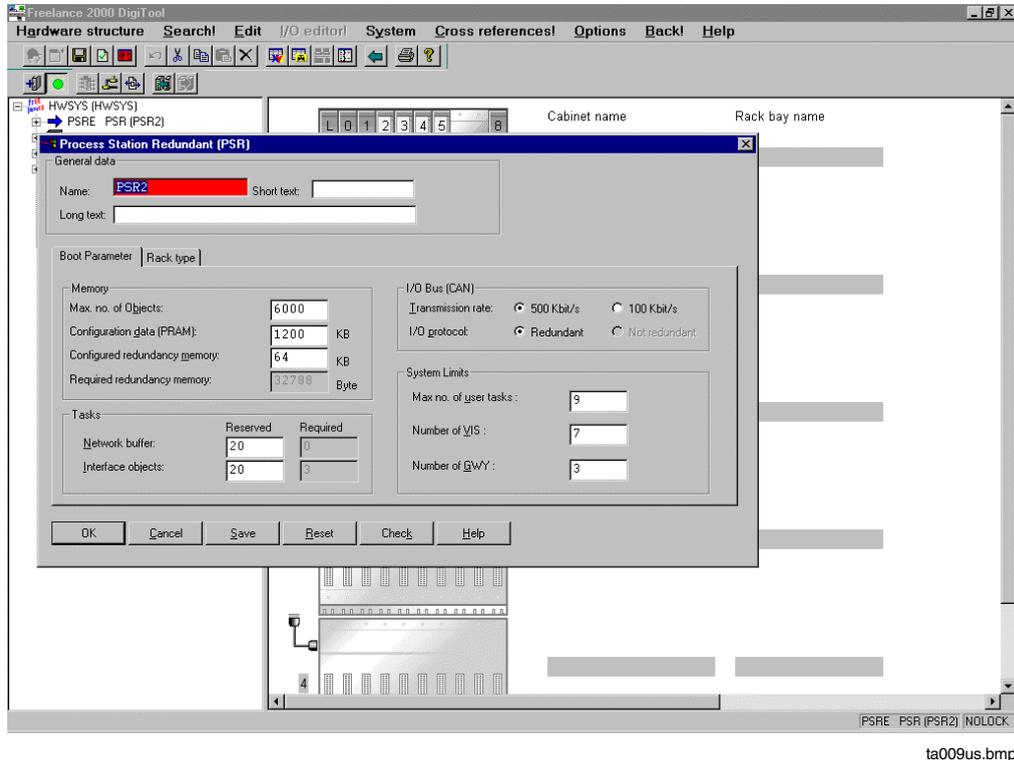
### A 3.10.8 Project tree

In the Project tree, the desired program can be quickly selected with *Search* and entry of one or more characters. Side effects caused by re-configuration are also visible when the project tree is not expanded, since they are passed upwards.



### A 3.10.9 Boot Parameters

The configuration of the boot parameters for the resources was taken over into the hardware structure. In order to do this, the resource node is selected.



### A 3.10.10 Graphics

In the graphics editor, it is possible to automatically adjust the size in different drawing areas.

For complete display of a graphical image, it is possible to call up an overview image. In this, elements which lie outside of the editing area can be seen again.

If you are working with lower resolution, a ruler shows the area visible later.

See the **Engineering Manual Operator Station, Graphics Editor**

## A 3.11 DigiVis

### A 3.11.1 Acknowledgement

The acknowledgement of sounds can be done **for all operator stations** together in the horn module.

### A 3.11.2 Selection in the Graphic Display

If an operator display of a measuring point (e.g. overlay image) is activated, and then a graphic display in which this is configured is called up, the activated measuring point is shown as selected in the display.

## A 3.12 DigiBrowse

### A 3.12.1 Print

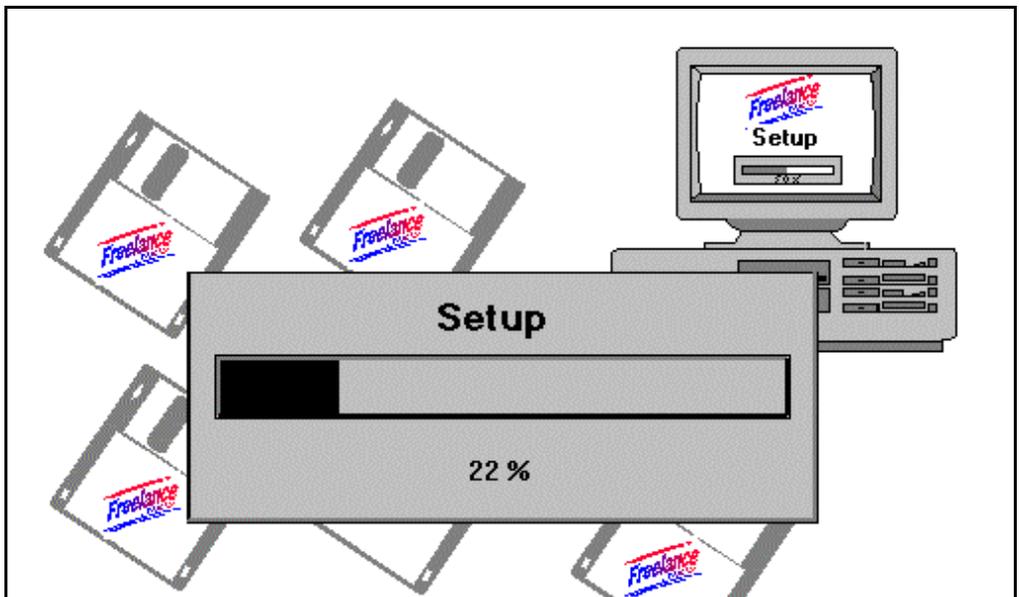
Trends and logs can now be printed from within DigiBrowse.

### A 3.12.2 Command Line for Batch Files

A command line interface is available for the creation of batch files. It is therefore possible to have incoming trend or log files automatically converted in the background. The configuration of the interface is done through the Windows input prompt. See the **DigiBrowse Manual**.



## B Installation DigiTool





## Contents

<b>B 1</b>	<b>General Notes .....</b>	<b>B-5</b>
B 1.1	Freelance 2000 Service Hotline .....	B-5
B 1.2	System requirements .....	B-5
B 1.3	Product options .....	B-5
B 1.4	Hard key .....	B-6
B 1.5	Add-on modules .....	B-6
<b>B 2</b>	<b>First installation of the DigiTool software.....</b>	<b>B-7</b>
B 2.1	Please note the following before installation .....	B-7
B 2.1.1	Windows NT installation completeness.....	B-7
B 2.1.2	Network and IP address settings.....	B-8
B 2.1.3	Virtual memory size.....	B-9
B 2.1.4	Network Configuration check .....	B-9
B 2.2	Starting DigiTool Setup .....	B-10
B 2.3	Installation options.....	B-11
B 2.4	Destination path for installation .....	B-12
B 2.5	Resource ID and IP Address.....	B-13
B 2.6	Time synchronization .....	B-14
B 2.7	Dimension and message text language .....	B-15
B 2.8	Authorization key installation.....	B-15
B 2.9	Boot EPROM download to D-PS, D-FC and DCP gateway station .....	B-16
B 2.10	Installation successfully completed .....	B-17
<b>B 3</b>	<b>Upgrade / Update of an existing DigiTool Version to V6.....</b>	<b>B-18</b>
B 3.1	Please note the following before an upgrade or update.....	B-18
B 3.1.1	Exporting project files from older versions as CSV-files .....	B-18
B 3.1.2	Documenting function block parameters .....	B-19
B 3.2	Uninstall before a new installation.....	B-19
<b>B 4</b>	<b>Correcting the Installation .....</b>	<b>B-20</b>
<b>B 5</b>	<b>Installing the DigiTool Help System .....</b>	<b>B-22</b>
<b>B 6</b>	<b>Checking the Installation with DigiCheck .....</b>	<b>B-24</b>



## B 1 General Notes

### B 1.1 Freelance 2000 Service Hotline

**Hotline** Should you encounter any unexpected problems during installation which are not covered in these instructions, our technical service will be happy to assist you.

Tel.: +49 (0)1805 12 35 80

Please call your local Freelance 2000 provider or your local ABB office.

### B 1.2 System requirements

**Operating system** Version 6.1 of the Freelance 2000 software has been tested and released only under the **Microsoft Windows NT 4.0** operating system with Service pack 5 installed.

### B 1.3 Product options



For Freelance 2000, you need a hard key and an authorization disk. These will enable you to work with the modules of DigiTool and/or DigiVis which you have ordered.

If you order additional licenses you will receive a new authorization code that should be entered as follows: → *Start* → *Programs* → *Freelance 2000 Tools* → *DigiTool Configure* → AUTHORIZATION CODE.

If you start the Freelance 2000 software without the hard key, it will run as a demo version for 100 days.



You will only be put into demo mode if there is no hard key installed on the PC during installation.

If a faulty hard key is detected the software goes into a safety mode (emergency mode) which—like the demo mode—lasts for 100 days. As long as the hard key is replaced within these 100 days the rights conferred by the license will remain in effect.

Similarly, if the hard key is unplugged after startup the software will go into safety mode.

## B 1.4 Hard key

Freelance 2000 products are protected by means of the hard key and associated authorization code, which activate only those products for which official orders have been processed.

There are the following versions of hard key:

- DigiTool  
DigiTool Select
- DigiVis  
DigiVis Select
- Combination (DigiTool and DigiVis)

## B 1.5 Add-on modules

Please order additional add-on modules separately.  
Add-on modules are:

16 character long tag names  
Sequence of Events  
Phase Logic Interface Function Block (for open batch)  
FPX Function Block (PhaseX Control for ABB Batch)  
User-defined FBs (function blocks)  
DigiCSO (Coupling to Symphony Maestro UX workstations)  
Maestro NT Add-On  
Modbus Master FBs  
Modbus Slave FBs  
Telecontrol Library IEC 870-5  
Sartorius Scale Interface  
Interbus FB  
Loop Tuner  
Freelance 100 / Protronic FB  
DigiDDE32  
DigiLock  
DigiBrowse  
OPC Server  
DMS / API  
Trendserver for DigiVis

## B 2 First installation of the DigiTool software

### B 2.1 Please note the following before installation

#### Checklist

- Is the **hard key** attached and screwed down to the LPT port?
- Is the Windows NT installation **complete** and does it include the Service Pack 5? See Section “**Windows NT installation completeness**”, **page B-7**.
- Is the **TCP/IP network protocol** installation complete and have the addresses been assigned? See Section “**Network and IP address settings**”, **page B-8**.
- Are the necessary **network services** installed?  
See Section “**Network and IP address settings**”, **page B-8**.
- Which colour palette is the computer set for?  
(→ Control panel → Display → Settings)?  
You can use Freelance with colour settings from *256 Colors* to *TrueColor*. The *TrueColor* setting is needed if photo-realistic bitmaps are to be displayed in images. Please note that the *TrueColor* setting requires much more memory (a factor of 3, since for each color, 24 rather than 8 bits are required) and CPU speed.  
The *256 Color* setting is recommended if DigiTool is being used on a computer with lower speed, or if the computer speed is required for other applications on the computer.
- Is the **screen saver** deactivated?
- Is the **virtual memory** size set to 100 Mb? See the section “**Virtual memory size**”, **page B-9**.

#### B 2.1.1 Windows NT installation completeness

Check to be sure that Service Pack 5 for Windows NT is installed. The Service Packs are available free of charge from Microsoft Corporation (also on the Internet at <http://www.microsoft.com>).

## B 2.1.2 Network and IP address settings

The network protocol used by Freelance 2000 is TCP/IP. No other protocol is required. Proper operation of Freelance 2000 with other network protocols is not guaranteed.

### IP address settings

For the PCs connected, the use of IP addresses from 172.16.1.20 up to 172.16.1.255 is recommended.



Taskbar → Start → Settings → Control Panel → Network → Protocols  
→ TCP/IP Protocols → Properties

IP address:           172.16.x.x                   exx. 172.16.1.20  
Subnet mask:        255.255.240.0

### Subnet mask for time synchronization

There is a coding switch on the rear panel of the CPU module with which the resource ID of the process station is set. For the FieldController the coding switch is located on the Ethernet module. The resource ID is normally used as part of the IP address.

If the coding switch is not set to zero (i.e. set to a value from 1 to F), then the subnet mask and the IP address are assigned as 255.255.240.0 and 172.16.1.x respectively, (where x = 1,...,15 as determined by the coding switch setting 1,...,F). If the coding switch of a process station or a FieldController is set to 0 (zero), then the IP address and subnet mask can be assigned as required.

This means that in order to permit time synchronization between all stations, the freely configurable stations (PCs or CPU modules with coding switch set to 0) must have their subnet mask set to 255.255.240.0 and their IP address to 172.16.x.y with x=0,...,15 and y=0,...,255.

Stations which are connected over a router do not participate in either normal time synchronization or in lateral communication. The function "Ext. time server" (see chapter **Project Tree**) can be used to implement time synchronization for external process stations and FieldControllers.

### DNS-Server and WINS entries

DNS and WINS entries are not meaningful for isolated systems. "IP Forwarding" should not be turned on under the Routing Entry.

**Installed services**

The following services must be installed under *Control Panel* → *Network* → *Services*:

- Computer Browser
- Workstation
- Server
- NetBios Interface

and, if local FTP access is required, e.g. for automatic export of archives/logs to another drive of the same PC:

- Microsoft Peer Web Server 3.x

**B 2.1.3 Virtual memory size**

A virtual memory size of 100 Mbytes is recommended. The upper and lower limits should be set to the same value.



Taskbar → Start → Settings → Control Panel → System → System Properties  
→ Virtual Memory → Change



The larger the installed physical memory the larger the virtual memory size should be.

**B 2.1.4 Network Configuration check**

The Windows NT and network configuration can be checked by “pinging” a station from the command line (after booting):



Taskbar → Start → Programs → Command Prompt

Enter existing address: ping TCP/IP address (for example ping 172.16.1.20)

Close Windows : “Exit”

## B 2.2 Starting DigiTool Setup

The setup installation program is started in Windows NT from the Explorer. Insert the CD-ROM or **Disk 1** in the drive.

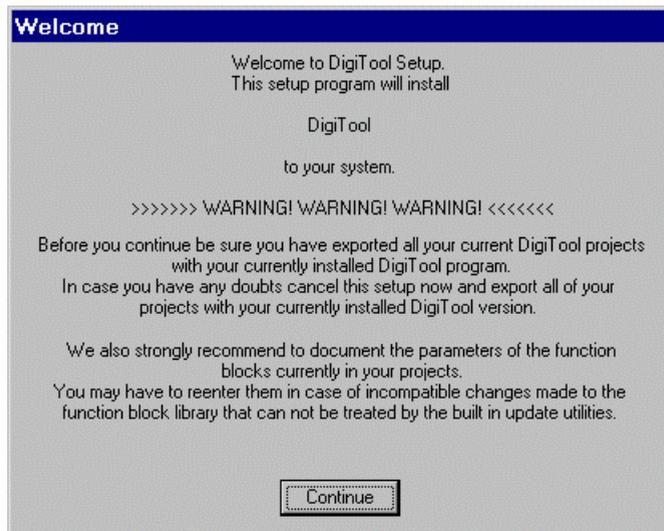


Left mouse button → Explorer → Select drive → Select file  
e.g. **a:\setup**



If you wish to install the program from the hard disk or from a network, make a directory and copy the setup disks supplied into it.  
Start the installation by running the file **SETUP.EXE** in this directory.

The following dialog box will be displayed. The warning message is aimed primarily at users of earlier versions of Freelance 2000.



di1706uk.bmp

CONTINUE opens the **License Agreement** window. To accept, please choose the corresponding button. With EXIT SETUP you will leave Setup.

### Notes on operation during setup

 It is always possible to abort the installation with **EXIT F3** or to call up a help file with **HELP**. Use the two buttons in the lower right corner of the setup display.



### Buttons in the dialog boxes

**CONTINUE** Continue the installation with the next dialog box.  
**BACK** Return to the previous window.

## B 2.3 Installation options



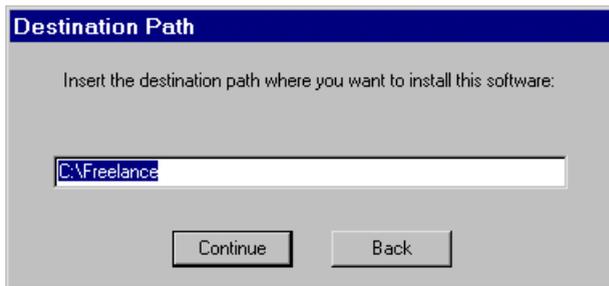
vb007uk.bmp

- |                   |   |
|-------------------|---|
| Standard Setup    | <input checked="" type="radio"/> Installation of DigiTool in the PC   |
| Server Setup      | <input type="radio"/> Installation of DigiTool in the Server for later workstation setup (enter destination path) |
| Workstation Setup | <input type="radio"/> Start setup from Server to the workstation  |

## B 2.4 Destination path for installation

DigiTool, like all other Freelance 2000 software programs, is installed in the directory c:\freelance by default. A number of sub-directories will be created under this directory.

freelance\bitmaps	Bitmap files for graphics editor
freelance\bpr	Operation logs
freelance\curv	Trend archives
freelance\ddesrv	DDE Server
freelance\dmsapi	Applications interface
freelance\doc	Project documentation
freelance\exe	.dll and .exe files for DigiTool
freelance\export	Exported project files or parts of programs
freelance\gsd	Device description files
freelance\macros	Macros
freelance\opcsrv	OPC server
freelance\proj	Project files
freelance\sap	Disturbance course logs
freelance\setctrl	Setup control
freelance\setups	Setup history
freelance\sfp	Signal sequence logs
freelance\wave	Wave files



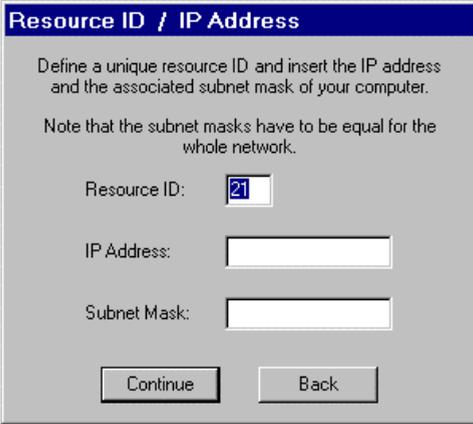
di1710uk.bmp

**Path** Accept the factory default shown or enter the name of another directory to be used for the installation of the Freelance 2000 files.

 If Freelance 2000 programs of the **same version** have already been installed in your system, you will be informed of this during installation. The path cannot be changed. In this case just press the CONTINUE button.

If Freelance 2000 programs of a **different version** have already been installed in your system, you will **not** be informed of this during installation, unless the software is installed in the same directory. Please note the instructions on **page B-19, Uninstall before a new installation.**

## B 2.5 Resource ID and IP Address



**Resource ID / IP Address**

Define a unique resource ID and insert the IP address and the associated subnet mask of your computer.

Note that the subnet masks have to be equal for the whole network.

Resource ID:

IP Address:

Subnet Mask:

tb002us.bmp

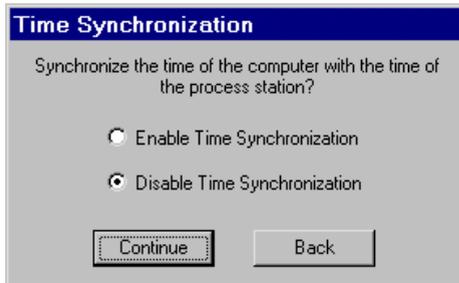
- Resource ID** Your PC requires a clear Resource Identification Number. The Resource ID should be between 20 and 255. Please note the Resource IDs already present in your Freelance 2000 system. The Resource IDs for the process stations are usually allocated as station numbers 1-10 for a Freelance 2000 system. See also the **Mounting and Installation Instructions**.
- IP Address** Enter the IP address  
The IP address uniquely identifies devices in the network.  
Entry: 172.16.1.x  
See also **Network and IP address settings, page B-8**.
- Subnet Mask** Defines whether the device addressed is located in the same network or must be addressed via routers.  
Entry: 255.255.240.0



If you operate your Freelance 2000 system in a network containing other network components, you should consult your network administrator before assigning the IP addresses.

## B 2.6 Time synchronization

It is possible to synchronize the time of the computer with the time of the process stations.



di1714uk.bmp



Time synchronization is not active until the time in the process station has been set with *Options* → *Set system time* in Commissioning.

 Time synchronization of external devices /stations /systems see also chapter **Project tree, Ext. time server**

## B 2.7 Dimension and message text language

The selection list contains the language abbreviations for the dimension and message text files already installed in your PC. The text in these files is used in the creation of the user program and is transmitted when the operator station is loaded. Please select the language which you wish to use in the operator station thereafter.

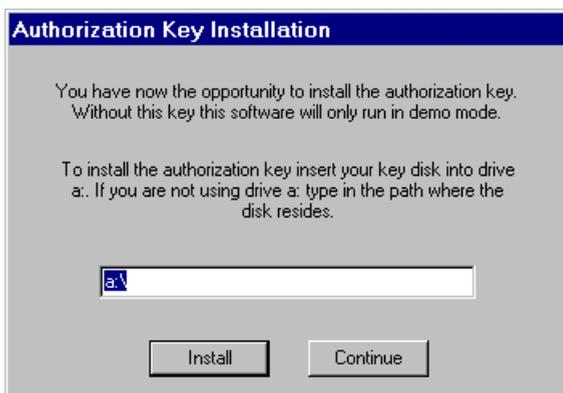


di1735uk.bmp

The files are copied.

## B 2.8 Authorization key installation

Insert the authorization disk enclosed in the installation pack into the disk drive.



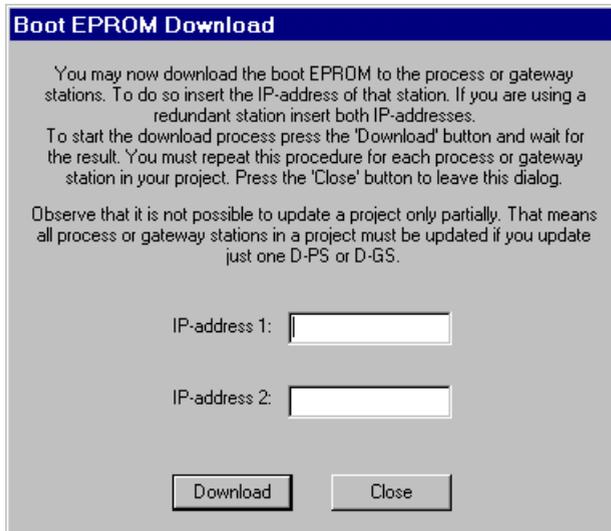
di1715uk.bmp

A message to the effect that installation was successful is displayed.

## B 2.9 Boot EPROM download to D-PS, D-FC and DCP gateway station

The EPROMs in the CPU modules of the process stations, FieldControllers and DCP gateway stations are updated.

The TCP/IP protocol must be loaded and there must be a connection to the process station, FieldController or gateway station via the DigiNet S system bus.



tb003us.bmp

**DOWNLOAD** The EPROM of the CPU module with the given IP address is updated to the current version.

**CLOSE** Ends the EPROM update and completes installation of DigiTool version.

If you have more than one station in your project carry out this procedure for each station.

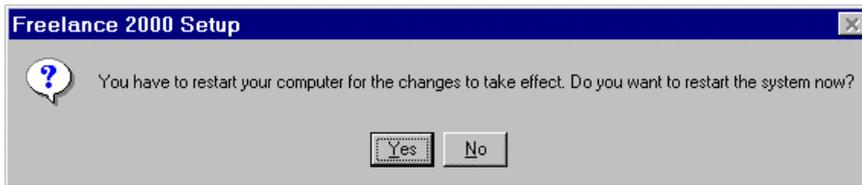
If you use a redundant station, enter both addresses (IP address 1 and IP address 2).

## B 2.10 Installation successfully completed

If all installation operations were successful, the following window will be displayed.



tb004us.bmp



tb012us.bmp

### Your computer must be restarted.

If you wish to install further Freelance 2000 products on this computer, you need restart the computer only once, at the end of the installation procedure.

We wish you every success using Freelance 2000 to perform your automation tasks. DigiTool will greatly reduce the effort required. By adhering to standards such as Windows and IEC 61131-3, DigiTool is an innovative product which leads the way in its field with concepts such as visualization and a common data bank for application program configuration.

Should you have unexpected problems during installation, please contact your local Freelance 2000 provider or your local ABB office.

A **service hotline** is available to you free of charge during the warranty period.

## B 3 Upgrade / Update of an existing DigiTool Version to V6

Definition: **Upgrade** refers to moving from one major version number to the next one.  
For example V5 to V6, (but not V3 to V6).

**Update** refers to a move within a major version number, from V 6.1 to V6.2.

### B 3.1 Please note the following before an upgrade or update

#### Checklist

- Recheck to make sure that you have exported all projects.  
See Section “**Exporting project files from older versions**”, page **B-18**
- Recheck to make sure that the function block parameters in your projects are documented.  
See Section “**Windows NT installation completeness**”, page **B-7**.

#### B 3.1.1 Exporting project files from older versions as CSV-files

If you wish to continue to process your existing projects from former versions, you must **export** them using your old DigiTool version (create .CSV-file) before you execute setup.

After successful installation of the new version, the required projects can only be **reimported** as CSV files

See also, **chapter Project Manager**.



**Caution:** In projects created with version 3.3 and earlier the same name could be used for function blocks (tags) and variables. As of version 4.1 conflicting names between tag names and variable names are not permitted anymore. Instead they will be neutralized!

With import of a project all conflicting names are removed automatically by completing the current variable names with `..._var`. In the variable list changed variable names can easily be selected with the search function. If one further uses the function cross references, the places of use can be discovered easily and be changed if necessary.

### B 3.1.2 Documenting function block parameters

It is recommended that you document the parameters of the function blocks in the projects. In the event of incompatible changes to the function block library which cannot be handled by the update routine, it may be necessary to re-enter parameters.

### B 3.2 Uninstall before a new installation

For an **update** within a version we recommend **uninstalling** the previous version.



Taskbar → START → SETTINGS → CONTROL PANEL → Add/Remove programs  
Select program group, e.g. Freelance 2000 → Add/Remove → OK

Thereafter the software must be installed as described in the chapter “**First installation of the DigiTool software**”, page B-7.



Do not forget to reinstall the authorization code from the authorization key disk to C:\Freelance\EXE!

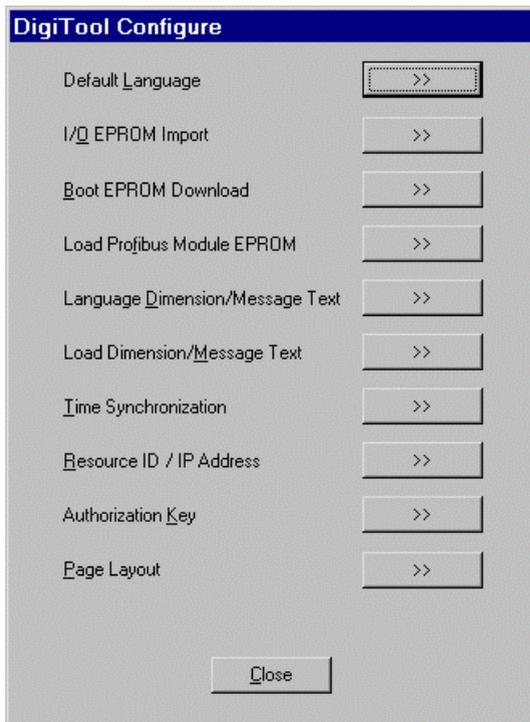
## B 4 Correcting the Installation

After installation has been completed, some parameters can be corrected without having to run through the full installation again.

Use the program “DigiTool Configure”.



Taskbar → Start → Programs → Freelance 2000 Tools → DigiTool Configure



di1718uk.bmp

**Default Language** Sets the default language for DigiVis, DigiTool and the supplementary programs DigiBrowse, DigiLock, etc.

**I/O EPROM Import** Loads I/O module firmware data into current installation.

Boot EPROM Download	Dialog to update the Boot EPROMs of the CPU modules. See also <b>page B-16, Boot EPROM download.</b>
Update Profibus Module EPROM	The current version can be loaded into the EPROMs of the profibus modules of all connected FieldControllers.
Language Dimension/ Message Text	Sets the desired language for message texts. See also <b>page B-15, Dimension and message text language.</b>
Load Dimension/ Message Text	<p>Loads the texts for other languages (from the Bonus disk) into the current installation. These texts are activated using Load Dimension/Message Text described above.</p> <p>WAV files for the selection list in DigiTool can be entered in the loaded file digitXX.str (XX stands for the language, e.g. GR = German, US = English). This is achieved by using an editor to add lines to the loaded digitXX.str file. Under the section [PARA] the names of the WAV files can be entered according to the following example:</p> <pre>WAVFILENAME_0=filename1 WAVFILENAME_1=filename2 WAVFILENAME_2=filename3 ...</pre> <p>The numbering should continue according to the same pattern. <b>filename</b> is the name of the WAV file that can be chosen from the selection list during configuration.</p>
Time Synchronization	Activates/deactivates time synchronization.
Resource ID / IP Address	Sets and changes the IP address, Resource ID and subnet mask. See also <b>page B-13, Resource ID and IP Address.</b>
Authorization Key	<p>New installation of the authorization key. After this you must restart DigiTool.</p> <p>See also <b>page B-15, Authorization key installation.</b></p>
Page Layout	Sets margins.

## B 5 Installing the DigiTool Help System

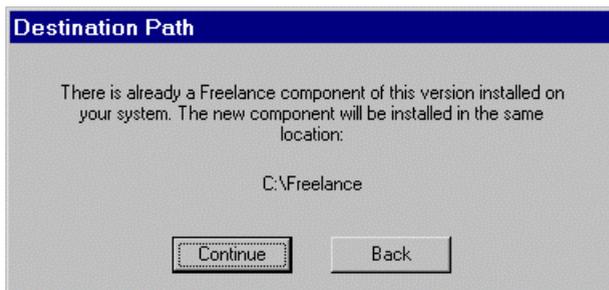
The Freelance 2000 help system provides quick, context-sensitive on-screen help while you are using the Freelance 2000 software.

 Do not install the Freelance 2000 help system until you have installed your **DigiTool Freelance 2000 software**.

To install the help system, insert the CD ROM or disk 1 of your Freelance 2000 help system in the appropriate drive and run the **SETUP.EXE** file.



di1742uk.bmp



di1743uk.bmp

When the Freelance 2000 help system is installed, the help files are copied to the directory specified for the installation of DigiTool, usually c:\freelance.



tb004us.bmp

OK      Completes the installation of the DigiTool help system.

The computer should be rebooted after successful setup.

## B 6 Checking the Installation with DigiCheck

With the DigiCheck test program, you can check whether certain conditions for a complete DigiTool installation are fulfilled.



Taskbar → Start → Programs → Freelance 2000 Tools → DigiCheck

```

DigiCheck V6.1 013 Check Results
Date: Friday, August 04, 2000
-----
Freelance 2000 components and version information:
Product           : DigiTool
Version           : 6.1 1201
Date              : 7/24/00 5:21:50 PM
Version Text      : BETA
Info Text         :
DigiToolUS       : 7-25-2000 17:18:24 ; 2938 Beta
DigiToolGR       : 7-24-2000 08:46:55 ; 2923 Beta
-----
Operating systems version information:
d:\Freelance\exe\MSR10S.BIN:
*** Freelance 2000 DPS Operating System
    For DCP02 (i960CA/CF), Vers. 6.04 (24 Jun 2000)
DigiPS_BS VERSION 006 001 279 Date: Tue Jul 25 10:58:59 2000 by quad Buildindex 1201$
d:\Freelance\exe\MSR20S.BIN:
-----

```

tb005us.bmp

The program can start a complete test with *Check* → *All Components*.

The items checked include the presence of the authorization key, the version of the operating system, etc.

It is possible to output a test log which allows you to compare the current and intended settings. The log can be output to a file or a printer.

## C Project Manager

Project name:	<input type="text" value="freelance"/>		
Project manager:	<input type="text"/>		
Project no:	<input type="text"/>		
Project orderer:	<input type="text"/>		
Project order no:	<input type="text"/>		
Project password:	<input type="text"/> <input type="button" value="Change"/>		
Project size:	71451K	Version:	06.10.1998 08:05:48
Project comment:	<input type="text" value="Manual DigiTool (1)"/>		



## Contents

<b>C 1</b>	<b>General Description - Project Manager</b> .....	<b>C-5</b>
<b>C 2</b>	<b>Starting the Project</b> .....	<b>C-6</b>
C 2.1	Starting DigiTool.....	C-7
C 2.2	Operation.....	C-7
C 2.3	Menu structure of project manager.....	C-8
<b>C 3</b>	<b>Editing Projects</b> .....	<b>C-9</b>
C 3.1	Creating new projects.....	C-10
C 3.2	General information on the project.....	C-11
C 3.3	Open a project.....	C-12
C 3.4	Hard disk memory requirements .....	C-14
C 3.5	Importing a project.....	C-14
C 3.6	Importing a backup.....	C-15
C 3.7	Exporting a project .....	C-16
C 3.8	Exporting a backup.....	C-17
C 3.9	Closing the project.....	C-18
C 3.10	Saving a project.....	C-18
C 3.11	Save project as.....	C-19
C 3.12	Deleting a project .....	C-19
C 3.13	Editing the project header .....	C-19
C 3.14	Editing a project comment.....	C-20
C 3.15	Setting up an online connection .....	C-20
C 3.16	Releasing an online connection .....	C-20
C 3.17	Exiting Freelance 2000 DigiTool .....	C-20
<b>C 4</b>	<b>Project Management Options</b> .....	<b>C-21</b>
C 4.1	DigiTool Version .....	C-21
C 4.2	Event log .....	C-21
C 4.3	DigiLock.....	C-22
C 4.4	Automatic execution of a command with PLC-Open-Export.....	C-23
C 4.5	Expanding tag names.....	C-23
C 4.6	Color table .....	C-25
C 4.7	DigiVis write access .....	C-25
<b>C 5</b>	<b>Appendix - Toolbar-Buttons</b> .....	<b>C-26</b>



## C 1 General Description - Project Manager

Dialog with the **DigiTool** software package starts with the **Project Manager** routine.

DigiTool is a **Configuration, Commissioning and Documentation** tool for the user programs and displays in a Freelance 2000 system.

Configuration using DigiTool is based on the IEC 61131-3 standard. You can use DigiTool to configure the entire process control level as well as create programs. This is facilitated by the fact that all components access a common database.

The system configuration is saved as a project file and is located in a preset project directory. To process a project, you must first open a project file. You then have unrestricted access to all project data within the project file.

Project name and file name of the project file can be selected independently from one another.

From the Project Manager function, you can save, rename or delete projects. The actual configuration, commissioning and documentation of the project take place in the project tree which you can access from the Configuration or Commissioning menus.

If there is an old project which you can use to base a new project on, it can be imported into the current database. You can also import project parts.

Refer to **chapter Project Tree**.

## C 2 Starting the Project

The Freelance 2000 DigiTool is started under Windows NT from the corresponding program group.



*Start* → *Program* → *Freelance 2000* → *DigiTool*

Once the software has finished loading, the version window is displayed. This window can also be called up at any time from the menu via *Help / About*.



di0927uk.bmp



To run Freelance 2000 a hard key and authorization diskette are required. These enable you to work with those components of DigiTool and/or DigiVis which you have ordered.

### Demo mode

DigiTool passes automatically to demo mode if no hard key has been found. Demo mode runs for 100 days and is only used for evaluation and presentation purposes. During this time, all operations such as save, export etc. are allowed. After the period of 100 days the software is no longer operational and must be re-installed if necessary.



Without hard key the usage of this software is not allowed for professional purposes.

## C 2.1 Starting DigiTool

After starting DigiTool, you access the Project Manager function by:

 → Continue

 → ENTER

From here, you can create new projects, open existing projects, and save or delete projects.

## C 2.2 Operation

DigiTool offers all the functions required to create, work on and commission a project. The various options for working on the project are provided on a menu.

All operator actions may be carried out by means of mouse or keyboard. For mouse operation the most important and frequently-needed functions are also made available through toolbars and shortcut menus. The buttons on the toolbars as well as the contents of shortcut menus are dependent on the current state of project processing. The toolbar buttons are used to initiate general, i.e. object-independent functions, while the functions on the shortcut menu relate to the object currently being processed. The shortcut menu is called up with a single click of the right mouse button.

For an explanation of the button symbols see **page C-2616, Appendix - Toolbar-Buttons**.

### C 2.3 Menu structure of project manager

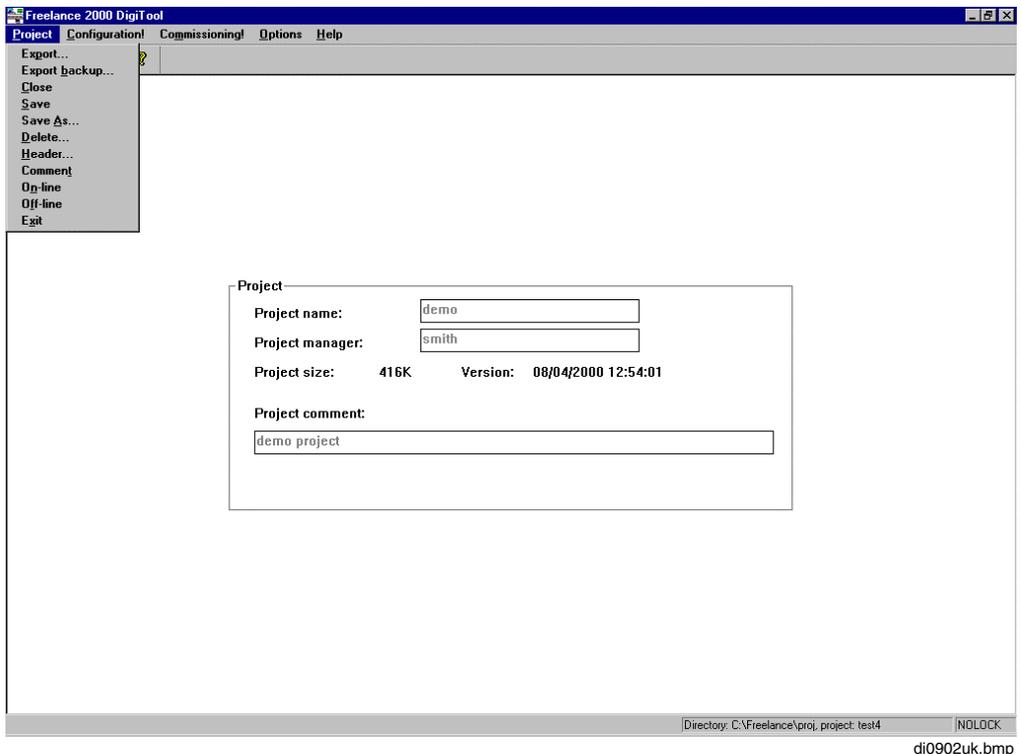
<b>Project</b>	<i>New</i> <i>Open</i> <i>Import</i> <i>Import backup</i> <i>Export</i> <i>Export backup</i> <i>Close</i> <i>Save</i> <i>Save as</i> <i>Delete</i> <i>Header</i> <i>Comment</i> <i>On-line</i> <i>Off-line</i> <i>Exit</i>
<b>Configuration!</b>	
<b>Commissioning!</b>	
<b>Options</b>	<i>Version</i> <i>Login (DigiLock)</i> <i>Logout (DigiLock)</i> <i>Change password (DigiLock)</i> <i>Run DigiLock (DigiLock)</i> <i>Tag names</i> <i>PLC-Open-Export</i> <i>Color table</i>
<b>Help</b>	<i>Contents</i> <i>Index</i> <i>Using Help</i> <i>About</i>

### C 3 Editing Projects

DigiTool can be operated in two modes. The first mode is Configuration in which you can structure, configure and document the project. Configuration can be processed off-line. This means that no Freelance 2000 system need be connected to configure all the project objects. The hardware can be ordered later or can be already installed while the user program is being configured. The project objects are assigned to the hardware structure as part of system configuration, and can then be downloaded to the stations when the connection is later on-line.

The second mode is **Commissioning**. When you select Commissioning, a connection is automatically set up to the process and operator stations via the system bus.

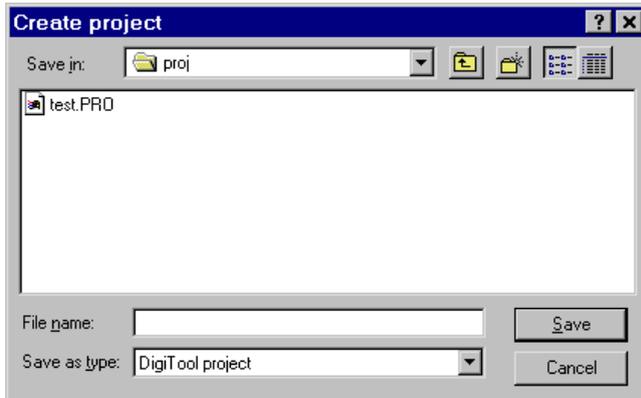
Also refer to **chapters Project Tree, Commissioning and Hardware Structure**.



### C 3.1 Creating new projects

 → Project → New

Specify a file name (max. 256 characters) in the create project window. The system appends the **.pro** file extension. The access to the directory is set by default. The previously active directory is called. The standard directory for projects on the hard disk is c:\Freelance\proj.



di0945uk.bmp

 → Choose SAVE button to call header

A project header contains general information on the project for use in project documentation, if necessary. This information is then documented in the drawing header or footer on a sheet.

See also **chapter Documentation**.

### C 3.2 General information on the project

di0909uk.bmp

Enter the following information which will apply to the entire project:

- Project name:* max. 12 characters independent from the file name (.pro)  
 You can only change the file name if you select *Save as*.
- Project manager:* max. 16 characters
- Project no.:* max. 6 characters
- Customer:* max. 27 characters
- Project order no.:* max. 12 characters
- Password:* max. 8 characters, minimum length 4 characters, to lock the project file.
- Project comment:* max. 34 characters

#### EDIT DRAWING

HEADER / FOOTER Assign specific system variables (F2) to the boxes in the drawing header. See also **chapter, Documentation**.

#### EDIT HEADER / FOOTER

TITLES You can change the titles in the headers of each window here. See also **chapter Documentation**.

OK Saves all entries in the database. The dialog box closes.

CANCEL Closes the dialog box without saving any entries.

### C 3.3 Open a project



→ *Project* → *Open*

Select a file name from the *Open Project* window and confirm by choosing the OK button. Access to the directory is defaulted during installation. The directory you selected previously is called. (The standard directory for projects is c:\Freelance\proj on the hard disk.)

When you open a project, the project file (**.pro**), a file with the extension **.log** and a file with the extension **.bak** are all created.

The file with the extension **.log** is a temporary file. Its purpose is to prevent the loss of changes made to the project during a session if Windows should crash. If this happens, you can restore the database after you restart DigiTool and reproduce the state at the time of the crash.



The temporary project file with the extension **.log** increases incrementally with every change in the project and is only reset to 0 Kbytes when the project is saved.

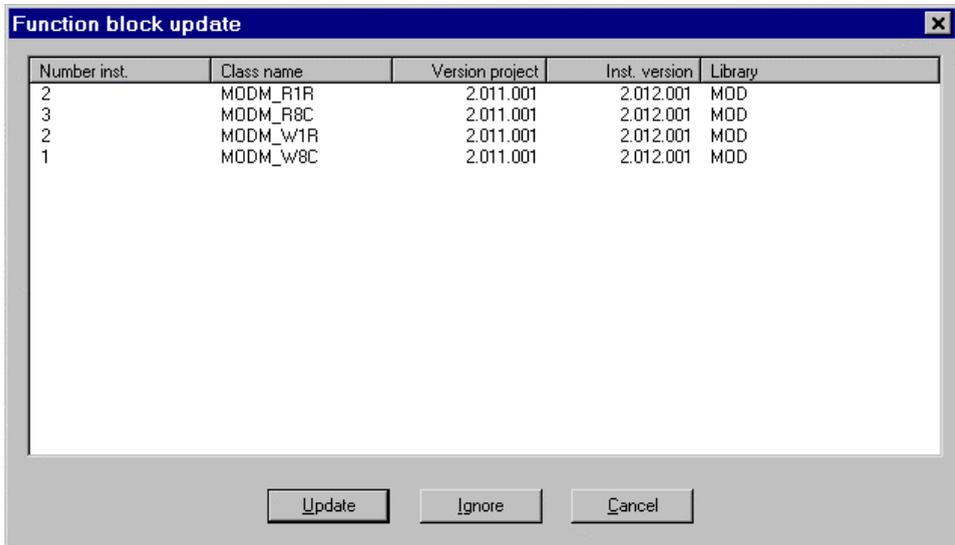
The file with the extension **.bak** is a compressed backup file of the saved project file. When you exit DigiTool and you select No from the *Save* menu item, the file is expanded and the extension **.bak** is replaced by the extension **.pro**.

When the project is opened the block versions in the project are compared with the versions of installed blocks. If any discrepancies are found, then a block update mask is opened, containing a list of all the block types with different version numbers.

A block type version number is composed of three levels. The individual version numbers are separated with a period.

#### Major version. minor version. code version

Major version	The major version no. identifies the external form of a block (e.g. its representation in the FBD program).
Minor version	The minor version no. identifies the parameters defined for a block (e.g. the layout of the parameters mask).
Code version	The code version no. identifies the stage reached in the internal processing of a block.



tc001us.bmp

The following parameters are shown for each block type:

Number inst.	Number of instances. Shows how many times the block type has been used in the project.
Class name	Name of the block type.
Version project	Version number of block type in the project.
Inst. version	Installed version number of the block type (DigiTool).
Library	Name of library to which the block type belongs.

The differences between versions can be updated or rejected.

**UPDATE** All the programs concerned are set to implausible. Differences in **major version** cannot be updated, and any blocks affected remain incompatible. In the case of differing **minor versions** an update function is called, which updates the blocks concerned. Block types with differing **code versions** are flagged for re-loading.

After the plausibility check and loading of changed objects it is possible to continue working with the project.

**IGNORE** All the programs concerned are set to implausible. All the blocks concerned remain unchanged in the project.



If you select IGNORE you shouldn't load any object in the process station.

**CANCEL** The project is left unchanged and is closed without being saved.

### C 3.4 Hard disk memory requirements

During a session on an open project, there are three files open: the project file (.pro), a temporary project file (.log) and a backup file (.bak).

The project file size is dependent on the size of the project and quickly reaches several 10 Mbytes of space requirements on your hard disk.

 We recommend at least 5 times the file size of a plausible project for the temporary requirements of free hard disk space.

A project which has been subjected to a plausibility check is about 3 times larger than the one which has not.

### C 3.5 Importing a project

 → *Project* → *Import*

This function imports the content of a project from a **csv file** previously generated with *Export*. The new project name is first displayed in the window. You can then save the project under this project name after the import routine has ended. When you confirm the file name of your choice, another window opens from which you can select the csv file you want from a directory of your choice.

Access to the import directory is by default. The previously active directory is called. The standard directory for Freelance 2000 import files on the hard disk is **c:\Freelance\export**.



Any changes in the csv file are not released and could possibly destroy your project database. You may cause a crash if you load a manipulated project file into the process station.



**Caution:** In projects created with version 3.3 and earlier the same name could be used for function blocks (tags) and variables.. As of version 4.1 conflicting names between tag names and variable names are not permitted anymore. Instead they will be neutralized! With import of a project all conflicting names are removed automatically by completing the current variable names with „\_var“. In the variable list changed variable names can easily be selected with the search function. If one further uses the function cross references, the places of use can be discovered easily and be changed if necessary.



The export of a project and the additional import of that project is only guaranteed within one main version and for any existing sub versions or between two sequential main versions. It is recommended that you go over the intermediate versions if there is a jump of more than one main version. In such cases, please contact our services.



As from version V3.1, CSV files are stored as Unicode files, and the only files that can be imported are CSV files in Unicode format. CSV files created using an earlier version of DigiTool must be converted from the ASCII character set with the help of the tool A2U\_CSV. This tool is only available with an English-language user interface and is called in program group *Freelance 2000 Tools*.



Before importing a project created under an earlier version than version 3.3 of DigiTool, the color table to be used should first be selected (see **page C-25, Color table**).

### C 3.6 Importing a backup

If you have no updated project file on the engineering station, you have the option of importing a backup of the project file.



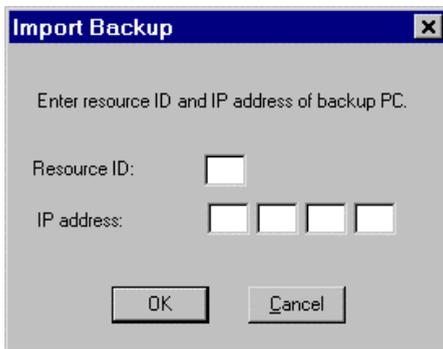
→ Project → Import backup



One condition for an import is that a backup was already exported to an operator station or that a backup file **backup.zip** exported via FTP-Export is located in the directory **c:\freelance\proj**.



No projects are allowed to be open. Any open projects must be closed before the import commences.



di0896uk.bmp

In the window, enter resource ID and IP address of the operator station where the project file backup is located.



→ OK

An on-line connection is automatically established and the project is transferred from the corresponding station via the DigiNet S system bus. The project file has been saved in compressed form in the c:\Freelance\proj directory under the file name backup.zip

 The IP address is part of the TCP/IP installation and can be changed in *system panel* → *network* → *protocols* → *TCP/IP protocol* → *IP address*. The station number is assigned with DigiVis setup and can be changed via *Freelance 2000 Tools* → *Configure DigiVis*.

See also **chapter Hardware Structure, Network**.

### C 3.7 Exporting a project

 → *Project* → *Export*

This function exports the entire content of the current project to a file. Either *Unicode export file (\*.csv)* or *PLCOpen file (\*.plc)* should be chosen as the file format. A **csv file** can then be reloaded using *Import*. A **PLC file** is used for transferring the project data to other systems, e.g. Maestro UX. You can assign a file name in the "Project Export" window that is opened.

Access to the directory is by default. The previously active directory is called. The standard directory for export files on the hard disk is **c:\Freelance\export**.



Where possible, only plausible projects should be exported. Only thus can it be ensured that all data are correct and consistent.



Warning: It is important with a PLC export that the names of tags and variables that are to be transferred to another system should conform to the naming conventions of the target system.

### C 3.8 Exporting a backup

You can save an additional project file to an operator station or via FTP export on a IP address in order to have a backup on the running machine. If necessary, maintenance personnel can access the backup.

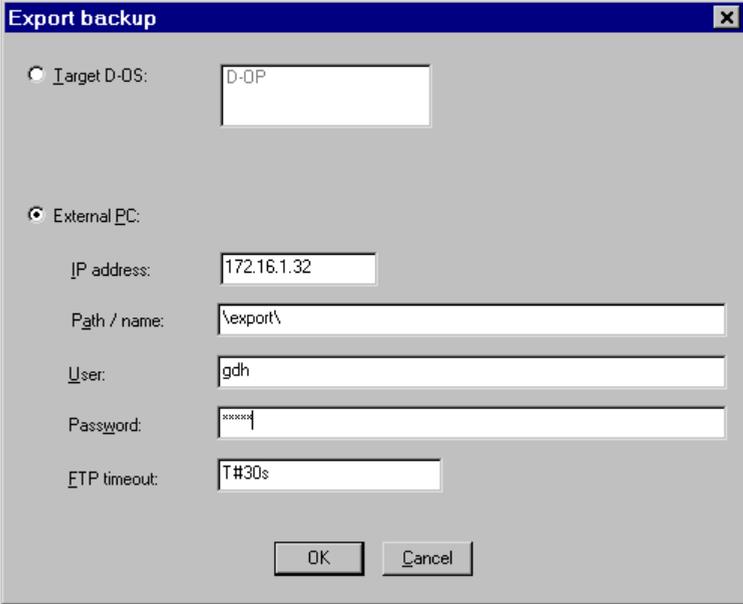
 → *Project* → *Export backup*

 Only saved projects can be saved as backups.

In the list box that opens, select one of the operator stations configured in the project or in the window enter the IP address of an external PC where the backup of a project file should be located.

 If no operator station is yet configured in the project, add a D-LS resource in the project tree and assign it to an IP address in the hardware structure.

See also **chapter Hardware Structure, Network**.



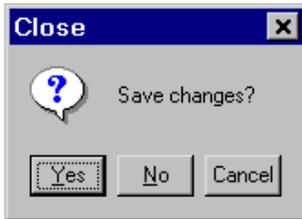
di0910uk.bmp

 → OK

Choose the OK button to set up an on-line connection automatically to transfer the project over the DigiNet S system bus to the operator station. The project file is then saved in compressed form in the c:\Freelance\proj directory under the file name backup.zip.

### C 3.9 Closing the project

 → *Project* → *Close*



di0897uk.bmp

 If you have made any changes to the project, you are normally asked whether you want to save these changes.

YES	Saves and closes an open project. The project file (.pro) is updated, and the temporary file (.log) and the backup file (.bak) are closed. The Freelance 2000 DigiTool user interface is then available for further projects.
NO	All changes made during the entire session are rejected. The backup file (.bak) is expanded and overwrites the project file (.pro)
CANCEL	Returns to the Project Manager

### C 3.10 Saving a project

 → *Project* → *Save*

When you select the *Save* command without giving any other instructions, you save all the changes which you have made after opening the project with *Open*. This includes the plausibility checks on project objects, loading the objects to a station or changing a configuration. The standard directory for projects on the hard disk is c:\Freelance\proj. The project remains open and can be edited.

 The save function resets the temporary file (.log) and releases memory.

### C 3.11 Save project as

If you want to change the project name, use the *Save as...* command.

 It is possible to change the file name of a project, e.g. in the File Manager. The project name continues to exist and can be changed in the project header.



→ *Project* → *Save as*

Select a file name from the "Save Project as" window and confirm by choosing the OK button. Access to the directory is by default. The previously active directory is called. The standard directory for projects on the hard disk is `c:\Freelance\proj`.

### C 3.12 Deleting a project



→ *Project* → *Delete*



When you choose the OK button, all the files associated with the project are deleted. The only exceptions are the files created using the *Export* or *Export block* commands.

### C 3.13 Editing the project header

The project header contains general information on the project which can be output with the project documentation.



Select Project Manager → *Project* → *Header*

The "Configuration: Project Header" window opens (described under creating new project). See also **page C-11, General information on the project**. The project size is also displayed in Kbytes as well as the date of creation and date when last saved.

See also **chapter, Documentation**, for the items referring to drawing header and footer.

### C 3.14 Editing a project comment



Select Project Object → *Project* → *Comment*

Use the Comment editor to create or modify a free text for the project. Additionally it is possible to import and export Unicode TXT files.

### C 3.15 Setting up an online connection

When you set up an online connection, you set up a communication link to the connected stations via the DigiNet S system bus.



The connection is automatically dialed when the function **Commissioning** is executed.

### C 3.16 Releasing an online connection

When you release the online connection, the communication link to the connected stations is released via the DigiNet S system bus.



If the bus is interrupted, a system message is displayed in the operator stations.

### C 3.17 Exiting Freelance 2000 DigiTool



→ *Project* → *Exit* → "Save changes?".  
→ OK

When you confirm this command, the open project is saved and closed. The project file is saved and DigiTool is terminated. The system returns to the Windows user interface.

## C 4 Project Management Options

### C 4.1 DigiTool Version

Using the menu option *Help / About* the window described on **page C-6, Starting the Project** can be opened.

 The DigiTool select license acquired with a Freelance select packet limits the number of tasks, control loops, trend acquisition blocks and the max. number of gateway stations. DigiTool select may be extended to standard DigiTool.

### C 4.2 Event log

The Event log is a log which can record operator or editor actions performed in DigiTool during **commissioning**. Since version 3 you can use the Window NT service. Therefore, start eventvwr.exe in directory c:\winnt\system32. If you select *Log* → *Application Freelance 2000* events are displayed. For detailed information double click on an event.

The following events can be recorded in a log, for example loading project objects in resources with:

- Date and time (h:min:s) of loading operation
- Loading state
- Object number
- Object type of loaded project object
- Name of loaded project
- Name of loaded project objects (task, program list, program)
- Name of loaded tag names (=function blocks) in this program

Commissioning interventions such as:

- D-PS Resource Start, Stop, Init/Reset and Boot.
- Task: Start, Stop, Reset and "Once" (single execution)

### C 4.3 DigiLock

DigiLock is an add-on package for DigiTool or DigiVis to permit the assignment of user rights and the definition of user groups.

Group specific rights can be awarded to the user groups for e.g.

- configuration
- commissioning
- operator interventions.

Every user identifies herself/himself by a password, which can be changed by her/him only.

If DigiLock is installed on an Engineering station, the user must enter her/his password before starting to edit or modify an project.

In the project tree, access rights edit or modify can be awarded to single or even several simultaneous project objects in an operator station. This will determine whether a DigiVis user may only view or also operate the specified displays. The same applies to logs or trend displays.

In the same way as displays and logs, access rights can be awarded to tag names in the tag list.

After installing DigiLock, you must first log in before you can work using DigiTool or DigiVis.

As long as no user has logged in, the rights assigned to the guest group are activated.

Standard users are:

<b>Nouser</b>	No license for DigiLock
<b>Guest</b>	No one has logged in, e.g. on starting DigiVis
<b>System</b>	Operator action by system (may appear in log)

 The LOGIN, LOGOUT, EXECUTE DIGILOCK and CHANGE PASSWORD actions can be recorded in the logbook file under DigiTool. The LOGIN, LOGOUT, and CHANGE PASSWORD menu options can only be executed if DigiLock has been installed.

See Manual **DigiLock**.

#### C 4.4 Automatic execution of a command with PLC-Open-Export

 → Options → PLC-Open export....

A Windows command can be automatically executed immediately after the creation of a \*.PLC file. After choosing the menu item → Options → PLC-Open export... a dialog box appears to enable a command and associated parameters to be entered.

Selecting *Execute automatically with PLC export* will result in the specified command being executed directly after the creation of a PLC file (via menu item → Project → Export → PLC file).

The example below will automatically create a backup of the PLC file:

Command: ***xcopy***

Parameter: ***c:\BACKUP\save.plc***

Every time a PLC file is created it will automatically be copied to directory ***c:\BACKUP\*** with the name ***save.plc***.

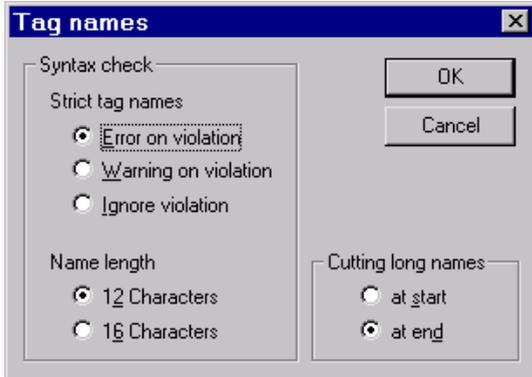
#### C 4.5 Expanding tag names

 → Options → Tag names

The syntax check in a project does not permit a tag name to start with a number (according to IEC 61131-3). In certain process sectors, e.g. in the power generation sector, this regulation must be bypassed.

It is possible to use 16 character long tag names (KKS) instead of 12 character long tag names. The standard is 12 characters. To be able to use 16 character long tag names this option must be ordered separately.

Special characters (! \$ % & / ( ) = ? B { } \ ' # + \* ≈ . : , ; |) and umlauts can be used in tag names only if check for IEC 61131-3 conform names is deactivated via Options → Tag names.



di0888us.bmp

### Strict tag names

- Error on violation*  All tag names must conform to IEC naming rules. (default setting)
- Warning on violation*  Tag names that do not conform to IEC are allowed, but a warning is issued.
- Ignore violation*  Tag names are not checked.

The setting chosen is stored both in the project database and the CSV export file. This ensures that the same name checking procedure is used each time a project is either opened or imported (e.g. to a different DigiTool PC).

### Name length

- 12 / 16 Characters  The length of tag names is limited to 12 or 16 characters. For 16 character tag name an extra license is necessary. When importing a project containing 16 character tag names into a DigiTool version without this license all names can be adjusted automatically. Without this adjustment the names have to be changed manually to get the project checked.

### Cutting long names

- at start / at end  If there is not enough space for displaying the whole tag name the name is cut in the display. The complete name can be shown by using the ToolTip.

## C 4.6 Color table

The colors available for graphic displays have been changed with version V3.3. From this version, color progressions are thus available, e.g. for 3D displays.

As of Version 6.1 not only 256 colors, but also True Color are supported.



Task bar → *Start* → *Settings* → *System panel* → *Display*  
→ *Settings* → *Color palette* → *Change*



If you have used bitmaps in the custom images for Digi-Vis, these will probably no longer have the same display as when 256 colors are used. You can choose to leave the setting at 256 colors, or update the bitmaps. When 256 colors are used, the use of other applications is not optimal. The applications of Freelance 2000 also work optimally with 256 colors.

When older projects are imported the colors previously available are converted as well as possible for the new color values. On the other hand, there is also an option to continue using the old color values. The color table can be selected **before a project is imported**.

### *Standard*

New color table is used (default).

Graphics created under an earlier version of DigiTool should be checked and, if necessary, corrected in respect of the colors used in them.

### *Compatible*

Old color table is used.

Graphics created under earlier versions of DigiTool are adopted without any modification. The new color values for displaying color progressions are not available.

## C 4.7 DigiVis write access

If only the process stations are loaded upon project re-configuration, a version conflict occurs between the operator stations and the process stations. Usually, write access from the operator stations is not useful then and, therefore, disabled. In special cases - and provided that the configuration changes made are exactly known - write access from the operator stations shall nevertheless be possible.



*Options* → *Enable DigiVis write access on version error*



Exclusively use this menu option in the commissioning phase. Otherwise, ignoring the version error may cause a system crash on the process station.

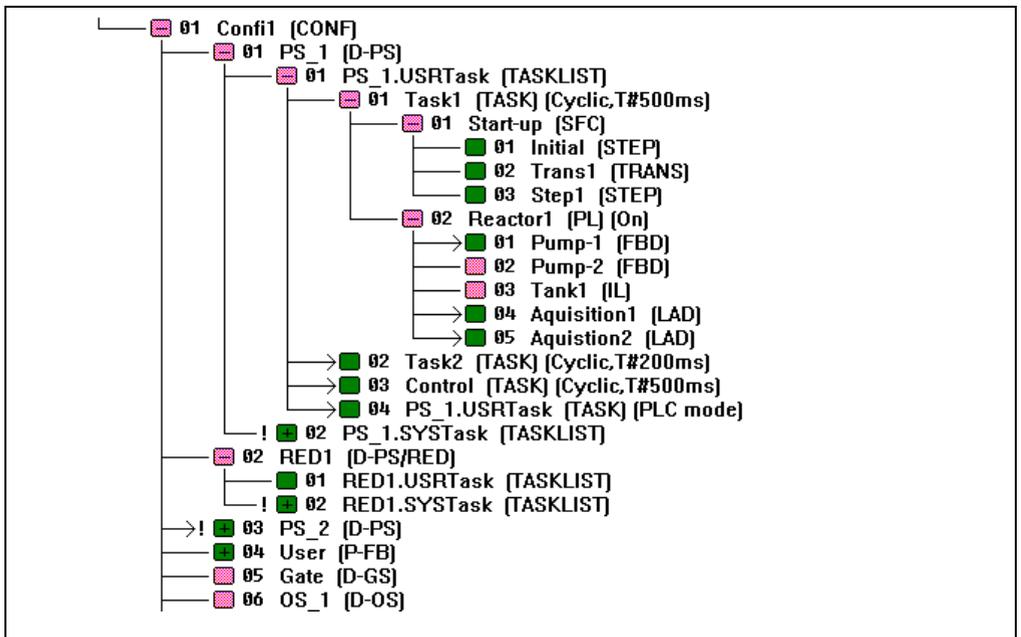
## C 5 Appendix - Toolbar-Buttons

-  Change to configuration mode
-  Change to commissioning mode
-  Check selected object with all accompanying sub objects
-  List all check messages of selected object and the accompanying sub objects
-  Save project or project part currently working on
-  Create a new project
-  Open existing project
-  Import a project file
-  Export of the current project
-  Call online help system
-  Edit the general data (header data) of the selected object
-  Call up the variable list
-  Call up the tag list
-  Call up the hardware structure
-  Call up the structured data types
-  Back to the program from which the current program has been called
-  Hardcopy the monitor content
-  Show value window
-  Show trend window
-  Define content for value and/or trend window
-  Insert a new object above the selected object
-  Insert a new object below the selected object
-  Insert a new object in next hierarchical level
-  Close all communication links

-  Load changed objects into selected station
-  Call up dialog editor
-  Insert following line
-  Show all cross references of selected variable or tag
-  Show interface declaration of current user defined function block
-  Show parameter mask of the accompanying task
-  Toggle display of drawing grid
-  Insert a column into a SFC program
-  Insert a row into a SFC program
-  Delete a column of a SFC program
-  Delete a row of a SFC program
-  Call up the operation dialog of on SFC program
-  Select all graphic objects of a graphic display
-  Return to the parameter mask of the graphic object
-  Change between graphic display and graphic pool
-  Call up the library functions for graphic macros
-  Edit a graphic macro
-  Activate or deactivate a station or a module
-  Show previous hardware object
-  Show next hardware object
-  Call up the I/O editor
-  Call up network editor



## D Project Tree





## Contents

<b>D 1</b>	<b>General Description - Project Tree</b> .....	<b>D-5</b>
<b>D 2</b>	<b>Structure of the Project Tree</b> .....	<b>D-8</b>
D 2.1	Project tree user interface .....	D-8
D 2.2	State line .....	D-8
D 2.3	Menu structure of the project tree .....	D-9
D 2.4	Display of project object states.....	D-10
D 2.4.1	Setting the node colors.....	D-11
D 2.5	Project objects .....	D-12
<b>D 3</b>	<b>Configuring the Project Tree</b> .....	<b>D-16</b>
D 3.1	Starting the project tree .....	D-16
D 3.2	Inserting project objects.....	D-17
D 3.3	Assigning project object names .....	D-17
D 3.4	Searching in the project tree .....	D-18
D 3.5	Expand, full expand, compress .....	D-18
D 3.6	Moving, deleting objects.....	D-19
D 3.6.1	Individual objects .....	D-19
D 3.6.2	Several objects (block) .....	D-20
D 3.7	Undo .....	D-20
D 3.8	Exporting and importing blocks .....	D-21
D 3.9	Pool for unneeded objects.....	D-22
D 3.10	Access rights and user groups.....	D-22
D 3.10.1	Access rights to DigiTool.....	D-22
D 3.10.2	Access rights to DigiVis in DigiLock .....	D-23
D 3.10.3	User groups .....	D-24
<b>D 4</b>	<b>Configuring the Project Objects</b> .....	<b>D-25</b>
D 4.1	Configuration (CONF) .....	D-25
D 4.2	Configuring the communication links .....	D-26
D 4.3	Resources.....	D-28
D 4.3.1	Process station .....	D-28
D 4.3.2	Operator station D-OS.....	D-32
D 4.4	Gateway station.....	D-33
D 4.4.1	Access rights of a gateway.....	D-36
D 4.5	Maestro configuration server MAESTRO-CS .....	D-37
D 4.5.1	Maestro RTDS.....	D-38
D 4.6	OPC server (OPC-S).....	D-39
D 4.7	User function block pool P-FB.....	D-40
D 4.8	Global display pool D-POOL .....	D-41

D 4.9	Task TASK and redundant task TASK/RED .....	D-41
D 4.9.1	Process image .....	D-44
D 4.9.2	Predefined system tasks.....	D-45
D 4.9.3	Program list PL.....	D-48
D 4.9.4	Programs on the operator station: Displays and logs .....	D-49
D 4.9.5	Structure node (STRUCT) .....	D-50
D 4.10	External timer server.....	D-50
<b>D 5</b>	<b>General Functions of the Project Tree .....</b>	<b>D-52</b>
D 5.1	Save .....	D-52
D 5.2	Documentation.....	D-52
D 5.3	Check, check all.....	D-52
D 5.3.1	Jumping directly to error points after plausibility check.....	D-53
D 5.3.2	Classes of plausibility errors.....	D-53
D 5.3.3	Structure of the plausibility check messages .....	D-54
D 5.3.4	Error list display.....	D-54
D 5.4	Show error list .....	D-55
D 5.5	Object header.....	D-55
D 5.6	Comments on a project object.....	D-55
<b>D 6</b>	<b>Project Tree Options.....</b>	<b>D-56</b>
D 6.1	Hardcopy.....	D-56
D 1.2	Long or short form in the state line .....	D-56
D 6.3	Colors in the project tree.....	D-57
D 6.4	Lock / unlock user function block.....	D-57

## D 1 General Description - Project Tree

The project tree provides an overview of the functions in a project. The individual elements or objects, generally known as **project objects**, are structured in accordance with IEC 6113-3.

The uppermost object in a project is the **Configuration CONF**, which is the sum total of all the project objects in a Freelance system.

The first structural level below configuration is formed by the resources, which represent the various different stations in a project. For the processing of the process itself there are the **D-PS (process station) resources**, for operation and observation of the process there are the **D-OS (operator station) resources** and **Maestro CS (Maestro NT configuration server)**, for interfacing to external systems there are the **D-GS (gateway station) resources**, and for integrating data from external systems there is the **OPC server** resource.

Data exchange between the process stations is performed by means of cross communication. Each process station is equipped with 10 connections for data exchange with operator stations and gateway stations. If more stations are configured, then the effective communication links of all the subscribers in a Freelance system are specified in a separate parameter definition dialog.

There are also additional structural elements, namely the **pool of user-defined function blocks P-FB** and the **global display pool D-POOL**.

According to the different tasks of these resources, additional project objects are available on the next levels for configuration:

### Process stations D-PS

Process stations are added with **resource types D-PS** and **D-PS/Red** (non-redundant and redundant stations respectively). Correlation to the physical stations is carried out in the hardware manager. Here, process stations (PS) can be selected for rack systems, and FieldControllers (FC and AC800F) can be chosen for connecting Fieldbus modules.

The execution of user programs within a process station is controlled by **tasks**. Within a task the sequence of user programs is defined either by **program lists** or by a structure - the sequence flow - that is configured using the **sequential function chart**. While the programs in a program list are run cyclically, sequential function chart programs are run only for specified periods of time according to their structure.

The user programs in the process station, both under a program list and under a sequence flow, are created using the IEC 6113-3 programming languages **function block diagram FBD**, **ladder diagram LD** or **instruction list IL**.

### Operator stations D-OS

For operation and observation of the process the **D-OS** resources can be selected for the operator stations with DigiVis software, while the **Maestro CS stations** can be selected for the Maestro NT operator stations.

Standard operating facilities, e.g. **faceplates**, are provided for all known tags and variables on the operator stations. In addition, **graphic displays**, **trend displays**, **SFC displays** and **logs** can be configured and structured for the DigiVis operator stations.

### Gateway stations D-GS

The gateway stations **D-GS** are used to make data from the Freelance system available to other systems. In principle, all the data from the Freelance system can be read and written via a gateway station. In addition to each gateway station in a Freelance system, the relevant server software from the add-on packages DigiDDE, DigiOPC or DigiCSO must also be installed on the network. (If, for example, a gateway station of type *OPC gateway* is configured in a Freelance system, then the *Freelance OPC server software* must be installed on a PC that is linked on the network with the Freelance process stations).

Parameters are set for each gateway station to specify which tags and variables are available to the other system for reading and/or writing via the gateway.

### OPC-Server server stations

Configuration of an **OPC-Server resource** in the project tree enables data from external systems to be integrated into a Freelance system.

According to the configuration of the gateway stations, parameters are defined for interfacing to an OPC-Server which determine which data from the external system is to be integrated into the Freelance project.

### Pool of user-defined function blocks P-FB

New classes of block are defined below this project object. These can then be used when configuring the user programs like the function blocks contained in the firmware.

### Global display pool D-POOL

Displays and logs set up under this project object should in principle be available on all operator stations. In fact the objects from the display pool are loaded only on those stations that have also been granted access to the process data needed in these displays and logs.

Alongside the Freelance project there is a project object **POOL**. This project pool can be used to temporarily store any project components as required. Any unchecked or unneeded configurations can be stored here and then completed or re-integrated into the project at a later point in time.

All the project objects are represented in a tree structure. A node is shown in front of each object. The color of the nodes is used to represent their processing state, and branches can also be recognized from the symbols. Sections of the project tree can be compressed as required, and this allows the overall structure to remain clear even in sizeable projects.

The DigiTool program contains two project processing states, **Configuration** and **Commissioning**. During configuration there must be no link in existence to the stations in the Freelance project. When switching to commissioning, a network link is established to all

Gross Automation, 1725 South Johnson Road, New Berlin, WI 53146, [www.ssacsales.com](http://www.ssacsales.com), 800-349-5827

configured stations. The configured and plausibility-checked user programs can be downloaded from the project tree to all connected process stations, operator stations and gateway stations.

The import and export functions that have been implemented can be used to exchange parts of the configuration with other projects. Checks are carried out when importing whole and part projects to ensure that all the labels within a project are unique.

If you want to award different **user rights** to various **user groups** to operate or configure the projects, DigiLock allows you to do this. **DigiLock** is an add-on package for both DigiTool as well as DigiVis. Refer to the **manual DigiLock**.

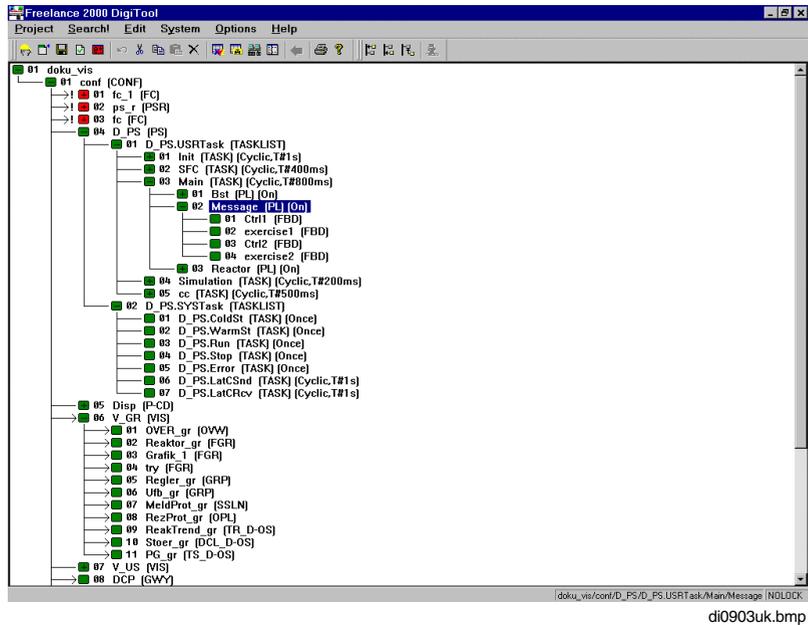
## D 2 Structure of the Project Tree

### D 2.1 Project tree user interface

menu line

project tree  
with nodes

state line



### D 2.2 State line

Depending on the user interface, the state line displays the following information:

- Operator hints
- Selected object in short or long form
- User or responsible operator logged into DigiTool. If DigiLock is not installed, the display "NOLOCK" appears.

## D 2.3 Menu structure of the project tree

### Configuration: Project tree

<b>Project</b>	<ul style="list-style-type: none"> <li>Save</li> <li>Documentation</li> <li>Check</li> <li>Check all</li> <li>Show error list</li> <li>Header</li> <li>Comment</li> <li>Commissioning</li> <li>Exit</li> </ul>
<b>Search</b>	
<b>Edit</b>	<ul style="list-style-type: none"> <li>Undo</li> <li>Program</li> <li>Insert above, Insert below, Insert next level</li> <li>Expand, Full expand, Compress</li> <li>Cut, Copy, Paste, Delete</li> <li>Export block, Import block, Import block as redundant</li>   <li>Access right (only on DigiLock)</li> <li>User groups (only on DigiLock)</li> <li>Display target stations</li> </ul>
<b>System</b>	<ul style="list-style-type: none"> <li>Variable list</li> <li>Tag list</li> <li>Structured data types</li> <li>Global message processing, Local message processing</li> <li>Hardware structure</li> <li>Display access</li> <li>Communication configuration</li> <li>Show global variables</li> <li>Show exported variables</li> <li>Show all objects</li> <li>Show selected objects</li> <li>Tele control signals (tele control blocks only)</li> <li>Display target stations</li> </ul>
<b>Options</b>	<ul style="list-style-type: none"> <li>Hardcopy</li> <li>Long state line</li> <li>Color settings</li> <li>Lock / unlock UFB</li> </ul>
<b>Help</b>	<ul style="list-style-type: none"> <li>Contents, Overview, Use help, About</li> </ul>

## D 2.4 Display of project object states

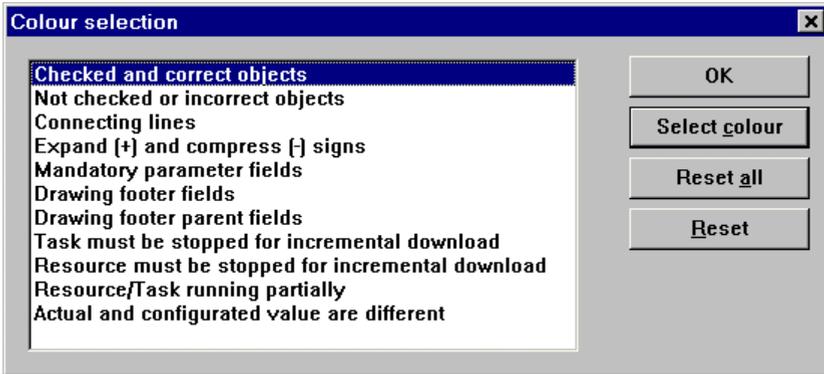
The following states which are generated by subjecting them to a plausibility check can be detected by displaying the nodes of the individual project objects.

-  The path is closed; there are no more branches.
-  The path is open.
-  The path is closed; there are side effects below the displayed level.
-  There are no more branches.
-  (Pink) Object has been changed; a plausibility check has not yet been performed or errors were found during the plausibility check.
-  Path with arrow: During the plausibility check, objects modified compared to the previous configuration state were detected.
-  (Flooded green) Plausibility check completed correctly.
-  (Flooded red) Plausibility check completed correctly; modified objects with side effects on the resource were detected. To load these objects, you must load the resource.
-  (Flooded yellow) Plausibility check completed correctly; modified objects with side effects on the task were detected. To load these objects, you must load the related task.

### D 2.4.1 Setting the node colors

The colors of the nodes are preset and you can modify them in the project tree using:

 → Options → Colors



di0894uk.bmp

OK	Adopts the color setting and exits the dialog.																						
SELECT COLOR	Opens a window for you to change the color state. Use DEFINE COLOR to compose your own colors.																						
RESET ALL	Resets all changed color settings to a default setting. The following colors can be set by default:																						
	<table> <tr> <td>Checked and correct objects</td> <td>green</td> </tr> <tr> <td>Not checked or correct objects</td> <td>pink</td> </tr> <tr> <td>Connecting lines</td> <td>black</td> </tr> <tr> <td>Expand + and compress - signs</td> <td>black</td> </tr> <tr> <td>Mandatory parameter fields</td> <td>red</td> </tr> <tr> <td>Drawing footer fields</td> <td>green</td> </tr> <tr> <td>Drawing footer parent fields</td> <td>pink</td> </tr> <tr> <td>Task must be stopped for incremental download</td> <td>yellow</td> </tr> <tr> <td>Resource must be stopped for incremental download</td> <td>red</td> </tr> <tr> <td>Resource/Task running partially</td> <td>red</td> </tr> <tr> <td>Actual and configured value are different</td> <td>yellow</td> </tr> </table>	Checked and correct objects	green	Not checked or correct objects	pink	Connecting lines	black	Expand + and compress - signs	black	Mandatory parameter fields	red	Drawing footer fields	green	Drawing footer parent fields	pink	Task must be stopped for incremental download	yellow	Resource must be stopped for incremental download	red	Resource/Task running partially	red	Actual and configured value are different	yellow
Checked and correct objects	green																						
Not checked or correct objects	pink																						
Connecting lines	black																						
Expand + and compress - signs	black																						
Mandatory parameter fields	red																						
Drawing footer fields	green																						
Drawing footer parent fields	pink																						
Task must be stopped for incremental download	yellow																						
Resource must be stopped for incremental download	red																						
Resource/Task running partially	red																						
Actual and configured value are different	yellow																						
RESET	Resets the currently selected color state to the default setting.																						

## D 2.5 Project objects

The figure beside the node shows the processing sequence of the project objects on the corresponding level. In the case of Tasks featuring different interval times, the project objects are processed according to their interval time irrespective of their processing number.

### General Process objects

	First line	The assigned project name appears here.
(CONF)	Configuration	The configuration level grouping all resources and permitting data transfer.
(D-PS or PS, FC, AC 800F)	Process station	<p>The process station contains the CPU module which processes the programs configured under the resource. The type of the process station is defined in the hardware structure by allocating the resource to the hardware object.</p> <p>The short label D-PS indicates that a station has not yet been assigned in hardware manager to a physical station. Following assignation, the assigned station type is displayed: <b>PS</b> for a rack system, <b>FC</b> or <b>AC800F</b> for a FieldController.</p>
(D-PS/RED or PSR , AC 800 FR)	Redundant Process station	<p>A redundant process station contains redundant CPU modules. The rack system uses two type DCP 10 CPU modules. A redundant FieldController consists of two AC 800F. The two controllers are connected by means of a redundant link and appear the same as a process station to the user. When the process station is loaded one of the two CPU modules becomes the primary CPU and the other becomes the secondary CPU.</p> <p>The short label <b>D-PS/RED</b> that a station has not yet been assigned in hardware manager to a physical station. Following assignation, the assigned station type is displayed: <b>PSR</b> for a rack system or <b>AC800F</b> for a FieldController.</p>
(D-OS or VIS)	Operator station	<p>An operator station is a resource permitting operation and observation on a PC using the DigiVis program package.</p> <p>The code <b>D-OS</b> indicates that the station has not yet been assigned to a physical station. After it has been assigned, the station type <b>VIS</b> is displayed.</p>
(MAESTRO-CS)	Maestro NT	A Maestro NT system can be used as an operator station for a Freelance system. With <b>MAESTRO-CS</b> the computer on which the configuration data for the Maestro NT system is stored is integrated into the Freelance project.

(D-GS GWY)	or Gateway station	<p>A gateway station enables data to be transferred to other systems via interfaces such as DDE or OPC. A gateway station is also set up in order to interface to a higher-level process control system such as Maestro.</p> <p>The code <b>D-GS</b> indicates that the station has not yet been assigned to a physical station in the hardware manager. After it has been assigned, the station type <b>GWY</b> is displayed.</p>
(D-GS/RED or GWY)	Redundant Gateway station	<p>The redundant gateway is only available for interfacing to a Maestro UX system.</p> <p>It must be equipped with two type DCP 10 CPU modules; one of these becomes the primary gateway, the other becomes the secondary gateway.</p> <p>The code <b>D-GS</b> indicates that the station has not yet been assigned to a physical station in the hardware manager. After it has been assigned, the station type <b>GWY</b> is displayed.</p>
OPC-S	OPC server	<p>An OPC server is used to import data from other systems into the Freelance project using an OPC interface.</p>
(P-CD)	Common display pool	<p>Certain displays and logs are set up below these operator stations; in principle these should be available on all operator stations. In fact the objects from the display pool should only be downloaded to those stations for which data access has been set up to the process data needed in these displays and logs.</p> <p>Trend displays and disturbance course logs which are linked with an acquisition module must be assigned to one specific operator station.</p>
(P-FB)	User function block pool	<p>New classes of block are defined below this project object; these can then be used in the configuration of the user programs like the function blocks contained in the firmware.</p>
Pool		<p>"Memory" of incorrect project objects or those no longer required for processing which you may want to return to the process.</p>

**Project objects of an operator station D-OS**

(STRUCT)	Structure node	Structural element of the operator station. To enable a better overview the displays and logs of an operator station can be grouped together with this project element.
(SFCP)	Sequential function chart display	Structured display presenting a program sequence configured in the sequential function chart.
(OVW)	Overview display	Display for the rapid selection of displays and/or logs. Up to max. 4 x 24 displays or logs can be entered in an overview display.
(GRP)	Group display	A group display is a group of several face plates. It offers the user the possibility of displaying associated tags in a display. Depending on the size of the face plates, up to 24 measuring points can be displayed simultaneously in a 4x4 grid.
(FGR)	Graphic display	Display of freely grouped static and dynamic display objects generated by the graphics editor.
(TR_D-OS)	Trend display	The trend display is for the graphic display of values across a time axis. A total of max. 6 trends can be displayed in one trend display.
(TS_D-OS)	Time scheduler display	The time scheduler display shows the state of a time scheduler and permits its operation. The display consists of a trend area for set point and actual value curves, a state field and the associated face plate.
(OPL)	Operation log	Cyclical, manual or event-related log containing max. 200 selected variables which are logged within a configurable text.
(SSL1)	Signal sequence log 1	Logging of system errors, fault messages, switching messages, operator instructions and operator actions with continuous printout.
(SSLN)	Signal sequence log N	Logging of system errors, fault messages, switching messages, operator instructions and operator actions in a log file. Printing is possible at the end of logging or manually.
(DCL_D-OS)	Disturbance course log	The disturbance course log belongs to the state logs. Its purpose is to log temporal sequences of selected analog and binary tags.

**Project objects of a process station D-PS or D-PS/RED**

(TASKLIST)	Task list	Object for separating the system tasks and the user tasks.
(TASK)	Task	Object which controls the processing of the subordinate program lists and sequential controllers within the resource. A distinction is made between cyclical tasks and those which are processed only once for specific events. In addition, a default task can be configured for each resource. This task is always executed when none of the other tasks is being executed (cyclically or once only).
(TASK/RED)	Redundant task	All subsidiary program lists and sequential function charts within this task are executed redundantly. All tasks can be in redundant format. All the variables in a redundant task must be written through the process image mode.
(SFC)	Sequential Function Chart program	This program is generated using the sequential function chart language. The assigned programs are executed using a structure.
(PL)	Program list	List of FBD, IL and LD programs which are processed according to the consecutive number in the project tree. Processing of the PLs can be switched ON or OFF.
(IL)	IL program	Program which was generated using the Instruction List (IL) language.
(FBD)	FBD program	Program which was generated using the Function Block Diagram language (FBD).
(LD)	LD program	Program which was generated using the Ladder Diagram (LD) language

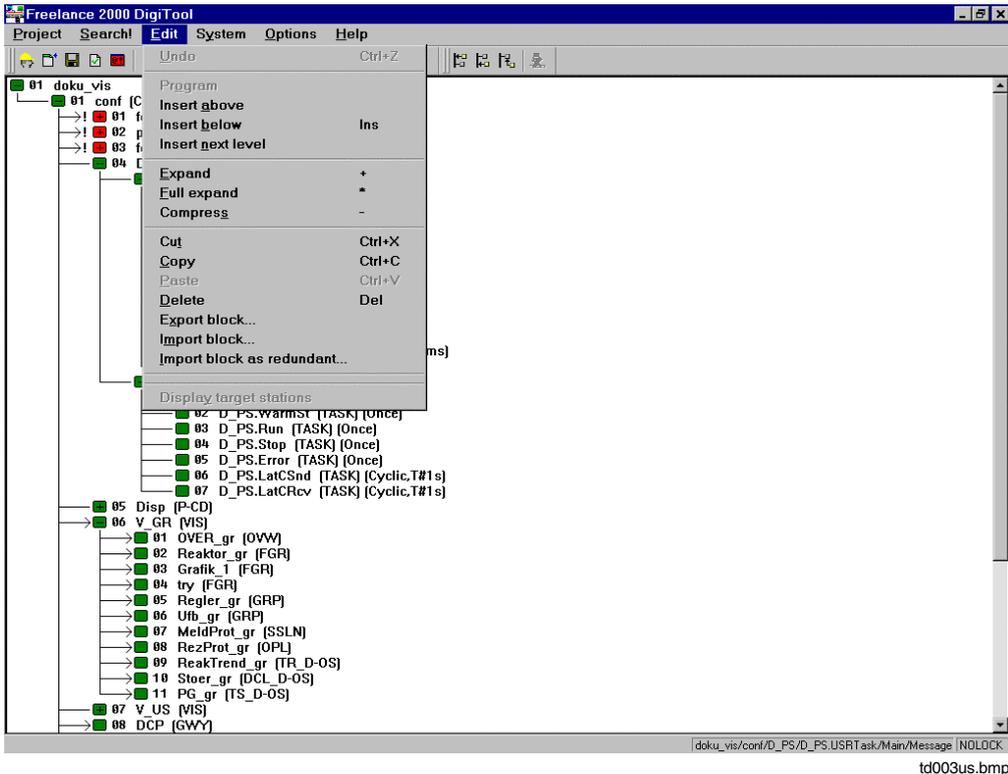
## D 3 Configuring the Project Tree

### D 3.1 Starting the project tree

After a new project has been created or an existing project has been selected in DigiTool, the project can then be configured or - if already configured - commissioned.

 → *Project* → *Open* → *Configuration*

The project structure is generated by selectively inserting the available project objects.



### D 3.2 Inserting project objects



Select the insertion position in the project tree

→ *Insert above*

Inserts a new object before the selected object on the same level in the hierarchy

→ *Insert below*

Inserts a new object under the selected object on the same level in the hierarchy

→ *Insert next level*

Inserts a new object one level lower in the hierarchy

Depending on the object selected in the project tree, the associated "Object selection" window opens.



→ Position pointer on object type and choose OK with the left mouse button

### D 3.3 Assigning project object names

Object names are assigned in the header of an object.

You must assign each object with a unique name. All upper-case and lower-case letters, figures and the special underscore character (    ) are permitted. An object name can be up to max. 12 characters long with the exception of the resources, where only 4 characters are allowed. The object header can be edited by selecting:



*Select Object* → *Project* → *Header* (see **Page D-25, Configuring the Project Objects** )



Using the comment editor, you can produce a free text several pages long for each object. This comment can then be output with the project documentation.



→ *Project* → *Comment*

See **chapter Project Manager**

### D 3.4 Searching in the project tree



*Search* → Specify a name or part of a name

The name to be searched for is entered either in full or in part in a dialog. The names of all objects in the project tree are checked. The first object found is marked in the project tree. If the SEARCH button is pressed repeatedly, all objects whose name contains the character string entered will be found. The search is case-sensitive.

### D 3.5 Expand, full expand, compress

To enhance the clarity of the project tree, individual project sections can be opened or closed. This is done by selecting the nodes ahead of the objects.

#### **Expand**

This is only possible if the node is displayed with  , the node is displayed by one level



Select node → *Edit* → *Expand*



Select node → Click left mouse button

#### **Full expand**

Only possible if the node is displayed with  , the node is fully expanded.



Select node → *Edit* → *Full expand*

#### **Compress**

Only possible if the node is displayed with  , the node is minimized to an object.



Select node → *Edit* → *Compress*



Select node → Click left mouse button

## D 3.6 Moving, deleting objects

Using the menus in the project tree, you can cut, copy, insert or delete single or blocks of objects and specify paths or subpaths. An exception to this rule is the project name and the system tasks. You can use the mouse to move objects without using the menus.

### D 3.6.1 Individual objects



Click the left mouse button on the name of the object you want to select.

#### Cut

Removes the object and saves it in the clipboard for you to insert later.



Select object → *Edit* → *Cut*

#### Copy

Saves the object in a clipboard for you to insert later at another position.



Select object → *Edit* → *Copy*

#### Paste

Before using the *Paste* command, you must first have copied or cut an object. If the insertion position is not permitted, the *Paste* command in the menu is dimmed (highlighted in gray).



Select insertion position → *Edit* → *Paste*

The Paste window opens for you to define the insertion position: *Above*, *Below* and, if necessary, *Level*. You must specify a unique name for each object which you copy or paste.

#### Delete

Deletes the objects you select from the project database. Any objects below the selected object are deleted along with it if they have also been selected; otherwise these objects are moved into the pool.



Select object → *Edit* → *Delete*



You are **not** asked if you really want to delete the objects if the node is displayed thus:



If the node is displayed thus , the delete window opens and you are asked: "Really delete object? Programs to be sent to pool?"

You can undo the delete operation with → *Edit* → *Undo*.

You cannot undo a delete operation after you have saved the project.

### Cut and Insert (Move)



Select object a second time by clicking the left mouse button and holding the mouse button down.

Move the mouse to the insertion position.

An icon appears, indicating whether insertion is permitted 

or not permitted .

Release the mouse button at the position you want.

The Move window opens for you to define the insertion position: *Above*, *Below*, and if necessary, *Level*.

### D 3.6.2 Several objects (block)

The objects you select are placed together in a frame and highlighted (in color) for further processing.

You can handle blocks in the same way as individual objects (see above). However, there is a difference in mouse and keyboard operation.



→ First select the object you want by clicking with the left mouse button and holding the left mouse button down.

→ Move the mouse to the next (second next, etc.)

→ Release the mouse button at the position you want

### D 3.7 Undo



→ *Edit* → *Undo*

Undoes only the last action you executed.

## D 3.8 Exporting and importing blocks

You can reuse project sections in the existing project or in other projects by exporting and importing blocks.

### Export block



Select block in project tree by dragging mouse → *Edit* → *Export block*

Exports the entire content of the block you selected to a PRT file which you can reload by using the Import Block command. You can specify the file name in the Export Project window that opens.

Access to the directory is by default. The previously active directory is called up. The standard directory for Freelance 2000 export files on the hard disk is C:\FREELANCE\EXPORT.

### Import block



Select block in project tree by dragging mouse → *Edit* → *Import block*

Imports the content of the block from a PRT file (which you previously generated by using *Export Block*) to the pool. From there, you can move the entire block or even individual objects to the position you want in the project tree. Access to the directory is by default. The previously active directory is called. The standard directory for Freelance 2000 import files on the hard disk is C:\FREELANCE\EXPORT.



Automatic rename for tag and variable names is controlled by two entries in the Windows NT Registry. Automatic rename for variable names is switched on if the entry ...SETUP\AutorenameEAM is set to 0, and switched off if it is 1. The entry SETUP\AutorenameMSR similarly controls renaming of tags. Both are preset to 0 by the Freelance 2000 setup. If rename is switched off, the import dialog asks the user, if he wants to rename or not. If the answer is yes, the names are extended by ...00. A further import would be extended by the number ...01, etc. If you choose NO, the variable names are retained. This function lets you link several project sections.



If 16 character tag names are used in a project but only 12 character tag names are set or licensed in DigiTool, the tag names will be cut at the beginning or end. See **chapter, Project Management, Expanding tag names.**

After importing, the imported variables are not assigned to any process station.

### Import block as redundant



Select block in project tree by dragging mouse → *Edit* → *Import block as redundant*

This function corresponds to the menu item Import block. In the import, all resources and tasks are converted into redundant resources or tasks. Access to all global variables in the imported programs can optionally be converted to an access via the process image.

This menu item is used to make existing projects “redundant”, in that the corresponding resources are exported as a block, then deleted, and then imported again.

### D 3.9 Pool for unneeded objects

The pool is a "memory" for incorrect objects or objects no longer required for processing and which you may want to return to the process. You can save individual programs or entire structures.

The processing options in the pool are identical to those in the higher-order process level.

-  If you import objects into the project, they are saved in the pool and you must then move them from there to the resource or task of your choice.

### D 3.10 Access rights and user groups

-  You can only assign certain access rights to various user groups if you installed the add-on package DigiLock. The separate software package DigiLock, is not part of DigiTool or DigiVis. It must be ordered separately.

See **manual DigiLock**.

#### D 3.10.1 Access rights to DigiTool

The following access rights are available to each of max. 16 user groups:

<b>No access</b>	The user cannot configure or commission the project.
<b>Configure</b>	The user can configure the project.
<b>Commission</b>	The user can execute all commissioning functions such as load user program sections or modify parameters.

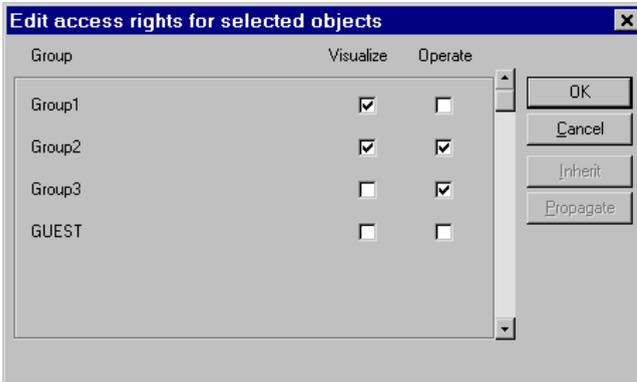
This definition applies to each project which is processed on the corresponding DigiTool PC. You can modify these rights in a dialog box in DigiLock.

See also **chapter Project Manager** and the **manual DigiLock**.

### D 3.10.2 Access rights to DigiVis in DigiLock

The following access rights are defined for the installed user groups for DigiVis:

 → *Edit* → *Access Rights*



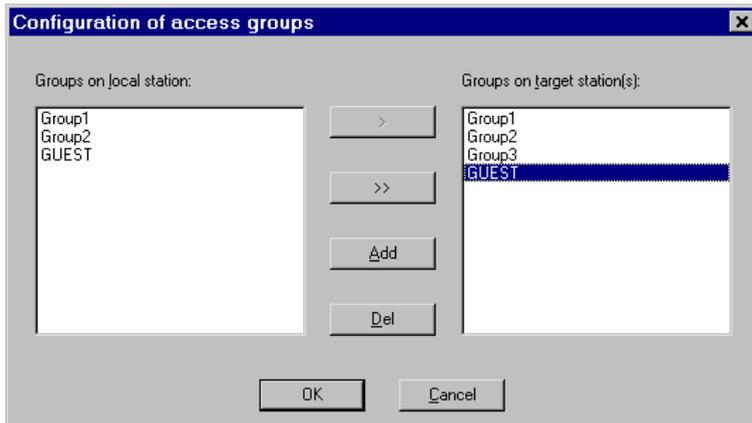
di4012uk.bmp

- |           |                                     |  |
|-----------|-------------------------------------|--|
| No access | <input type="checkbox"/>            | The user cannot operate or observe on the operator station.  |
| Visualize | <input checked="" type="checkbox"/> | The user may open displays and logs but not operate them.  |
| Operate   | <input checked="" type="checkbox"/> | The user can not only open displays and logs but also operate values, states of visualized variables and tags. |
| INHERIT   |                                     | Inherits the Access Rights setting from the higher-order objects.  |
| PROPAGATE |                                     | Propagates access rights to all lower-order objects.   |

### D 3.10.3 User groups

DigiLock can create up to max. 16 user groups (user profiles) each containing max. 1000 users. The user groups defined by DigiLock on the engineering station (local station) can be assigned to target stations in the project tree. A target station in this case means all operator stations in the project.

 → Edit → User groups



di4005uk.bmp

- > Adopts the selected group at the local engineering station for the operator stations.
- >> Adopts all the groups in the local engineering station for the operator stations.
- ADD Adds a new user group for the operator stations.
-  User groups assigned to target stations in the project tree must also be made known to each operator station using DigiLock.
- DEL Deletes user group from the selected operator station.
-  At the operator station (DigiVis), the user only sees the user's name of the access rights in the state line. The user's actions are also logged in the signal sequence log.

## D 4 Configuring the Project Objects

When you add a new object to the project tree, first assign a name to the object and, if necessary, a short comment. You can modify these parameters later in the menu. Depending on the object, you can display additional information, such as:

- Type of object
- Version (date, time of creation or last modification)
- Number of subordinate objects
- Processing sequence

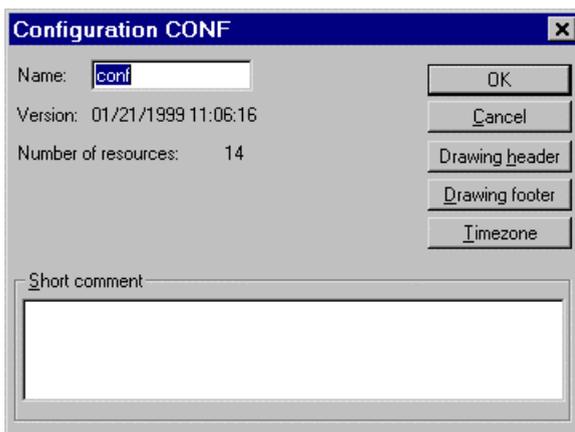
You can define a drawing header or footer for all objects.  
See also **chapter Documentation**.

 Select object → *Project* → *Header*

### D 4.1 Configuration (CONF)

The Configuration object summarizes information relating to the entire user program, and thus all subordinate project resources. The configuration contains all the station resources (process, operator and gateway stations) along with the pool of user-defined function blocks and the global display pool.

On this project object - or within the hardware manager - the communication links can also be configured for all stations in the Freelance project.



di0917uk.bmp

<i>Name:</i>	Max. 12 characters
<i>Version:</i>	Date and time of object creation
<i>Number of resources:</i>	Number of resources configured in the project tree
<i>Short comment</i>	Max. 159 characters
DRAWING HEADER / FOOTER	See also <b>Page D-52, Documentation</b> and <b>chapter Documentation</b> .
TIMEZONE	Set the time zone and summertime for the whole project.



The dialog for setting the time zone and summertime is structured in the same way as the Windows NT setting.

## D 4.2 Configuring the communication links

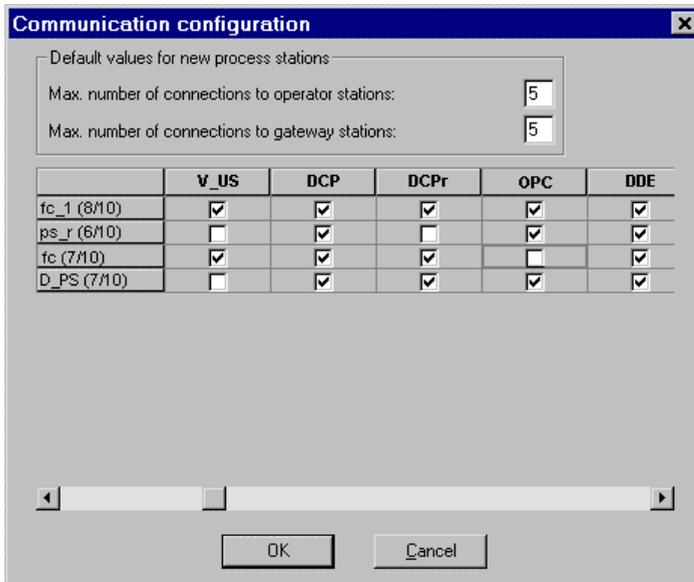
Each process station is equipped with 10 connections for data exchange with operator stations and gateway stations. If more stations are configured, then the effective communication links of all the subscribers in a Freelance system can be specified in a separate parameter definition dialog.

Double-clicking on the project object Configuration CONF or choosing from the menu:



→ *System* → *Communication configuration*

will call up the dialog.



td042us.bmp

### Default setting for new process stations

*Max. number of connections*

*to operator stations / gateway stations*

If a new process station resource is added to the project tree, the number of connections is pre-set using these values. Each process station can communicate with up to 10 operator stations and/or gateway stations. In other words, a value between 0 and 10 may be entered in each field but the sum of the two values must not exceed 10.

### *Matrix field*

All the process station resources are listed in the left column, and all the operator- and gateway station resources with their names as defined in the project tree are listed in the top row. Each field in this matrix represents a connection between a process station and an operator station or gateway station.

Connection in online mode should establish a link between the process station and the operator station.

Up to 10 links can be configured in each row of the matrix.

### D 4.3 Resources

The first structural level below configuration is formed by the resources, which represent the various different stations in a project. For the processing of the process itself there are the **D-PS (process station) resources**, for operation and observation of the process there are the **D-OS (operator station) resources** and **Maestro CS (Maestro NT configuration server)**, for interfacing to external systems there are the **D-GS (gateway station) resources**, and for integrating data from external systems there is the **OPC server** resource.

Data exchange between the process stations is performed by means of cross communication. Each process station is equipped with 10 connections for data exchange with operator stations and gateway stations. If more stations are configured, then the effective communication links of all the subscribers in a Freelance system are specified in a separate parameter definition dialog.

There are also additional structural elements, namely the **pool of user-defined function blocks P-FB** and the **global display pool D-POOL**.

 The resources in the project tree are assigned in the hardware structure of a station. Using the network configuration (menu item in the hardware structure: Hardware structure → Network...) the IP addresses and resource IDs are specified. They are thus accessible via the DigiNet S system bus on commissioning. Also refer to **chapter Hardware Structure**, Network and Assigning Stations, and **chapter Commissioning**, Loading the project.

It is also possible to configure more than one resource on a PC. For example, a D-OS resource and an OPC server can be operated simultaneously on the same PC, provided these two resources have different resource IDs so that they are addressable by the system.

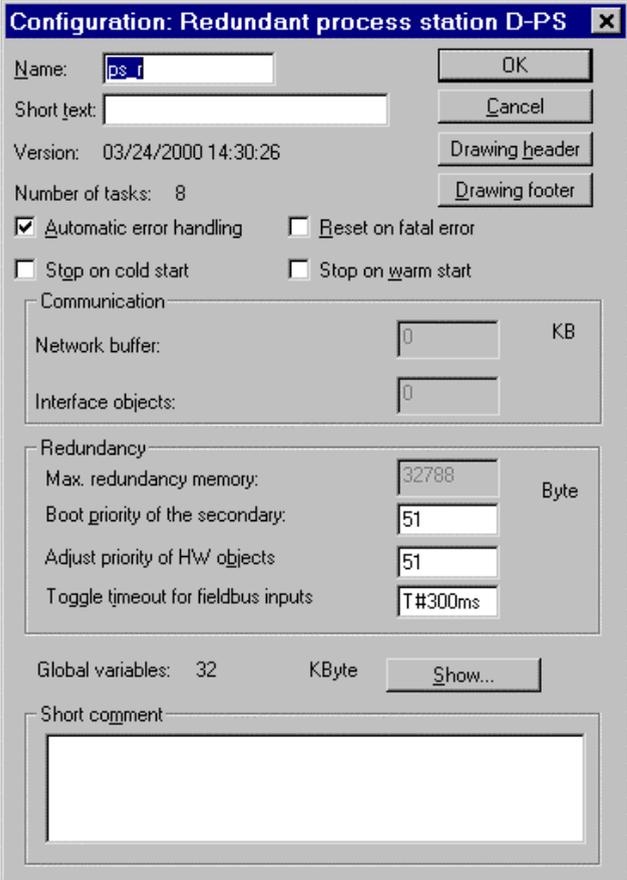
#### D 4.3.1 Process station

Process stations are added with **resource types D-PS** and **D-PS/Red** (non-redundant and redundant stations respectively). Correlation to the physical stations is carried out in the hardware manager. Here, process stations (**PS**) can be selected for rack systems, and FieldControllers (**FC** and **AC800F**) can be chosen for connecting Fieldbus modules. The network addresses and resource IDs of the stations are also defined within the hardware manager, and in commissioning mode the configured program modules are loaded from the project tree into the relevant stations. See also **chapter Hardware Structure**, Network and Allocation of Stations **chapter, Commissioning**, Loading the Project and Commissioning the Project Objects.

The execution of user programs within a process station is controlled by **tasks**. Within a task the sequence of user programs is defined either by **program lists** or by a structure that is configured using the **sequential function chart**. While the programs in a program list are run cyclically, sequential function chart programs are run only for specified periods of time according to their structure.

The user programs in the process station, both under a program list and under a sequence flow, are created using the IEC 6113-3 programming languages **function block diagram FBD**, **ladder diagram LD** or **instruction list IL**.

 Select object in the project tree → *Project* → *Header...* or double-click



**Configuration: Redundant process station D-PS**

Name:  OK

Short text:  Cancel

Version: 03/24/2000 14:30:26 Drawing\_header

Number of tasks: 8 Drawing\_footer

Automatic error handling  Reset on fatal error

Stop on cold start  Stop on warm start

Communication

Network buffer:  KB

Interface objects:

Redundancy

Max. redundancy memory:  Byte

Boot priority of the secondary:

Adjust priority of HW objects:

Toggle timeout for fieldbus inputs:

Global variables: 32 KByte

Short comment:

td005us.bmp

**Name:** Max. 4 characters

**Short description:** Max. 12 characters

**Version:** Date and time of object creation

**Number of tasks:** The number of tasks configured for this resource

*Automatic error handling:*

When error correction is on, errors occurring in the user program at run time (e.g. a division by zero) are automatically corrected. See **also chapter Commissioning, Error handling for the task.**

*Reset on fatal error*

If an internal error is detected in the sequence while the user programs are being processed in the resource, the CPU module will suspend processing. In the case of a non-redundant system, the outputs revert to the configured safety values, and the Failure LED flashes on the CPU module. This parameter allows you to set the CPU module to exit from this type of error state after 10 seconds by means of a cold start.

*Stop on cold start:*

After a cold start, the resource does not switch automatically to the 'running' state; it remains in the 'cold start stopped' state before computing the cold start task, and must be started from DigiTool.

*Stop on warm start:*

After a warm start, the resource does not switch automatically to the 'running' state; it remains in the 'warm start stopped' state before computing the warm start task, and must be started from DigiTool.

**Communication**

Communication management parameters. Every communication link requires internal system resources such as additional memory or system objects.

*Network buffer*

Area of memory in Kbytes that is reserved for the communication links.

*Interface objects*

Maximum number of objects that can be reserved for interfaces.

<b>Communication link</b>	<b>Required no. of interface objects</b>
Link module DCO01	1
Tele Control Library	4
Profibus Master	2
Profibus Slave	0
Modbus Master	2
Modbus Slave	2
Protronic / Freelance 100	1
Satorius	1
Send	1
Receive	1
Interbus Send	1
Interbus Receive	1

<b>Redundancy</b>	This information is available only for redundant resources.
<i>Max. memory req'd for redundancy</i>	Maximum amount of memory in bytes that the project requires for the transfer of redundancy data.
<i>Boot priority of the secondary.</i>	<p>In order to synchronize a redundant station, the bootstrapping of the second (secondary) CPU is initiated by the first (primary) CPU. This procedure is performed 'simultaneously' with the execution of the configured programs.</p> <p>This parameter controls the division of CPU utilization between normal program execution and bootstrapping the secondary. The higher this value is set, the more quickly the redundant process station will reach the state <i>sync</i>, and the more significantly normal program execution will be affected.</p>
<i>Adjust priority for hardware objects:</i>	<p>The objects configured in the hardware structure carry out the balancing of their redundancy data using a special mechanism.</p> <p>This parameter is used to specify the division of CPU utilization between normal program execution and this data balancing.</p>
<i>Toggle timeout for Fieldbus inputs:</i>	<p>Following a redundancy switchover, the initial data of the linked Fieldbus devices is acquired anew. The time required for this depends on the devices that are connected and the transfer rates used.</p> <p>This parameter determined the maximum length of time that the system should wait for new data. After the amount of time set here has passed, program execution will be initiated regardless. If not all of the data has been successfully re-acquired, a system message will be generated.</p>
Global variables	See <b>chapter Commissioning, Global resource variables</b>
Size:	Currently 32 KB
SHOW	Shows the memory assignment of the resource
<i>Short comment</i>	Max. 159 characters
DRAWING HEADER and FOOTER	See <b>Page D-52, Documentation</b> and <b>chapter Documentation</b> .

### D 4.3.2 Operator station D-OS

The **D-OS** resource is provided for operation and observation of the process by an operator station using DigiVis software. Assignment to a physical station is indicated by the code **VIS**.

Standard operating facilities, e.g. **faceplates**, are provided for all known tags and variables on the operator stations. In addition, **graphic displays**, **trend displays**, **SFC displays** and **logs** can be configured and structured for the DigiVis operator stations.

External process stations can be time-synchronized by an operator station.



Select the operator station object in the project tree → *Project* → *Header*

td004us.bmp

- Name:** max. 4 characters.
- Short description:** max. 12 characters.
- Version:** Date and time the object was created.
- Number of displays:** Number of displays and logs configured in the resource.
- D-OS password:** Enter a password that allows the operator to exit from DigiVis. If no password is specified, any operator can close DigiVis.

<i>Diag. password:</i>	Enter a password that allows the operator to switch to the DigiVis diagnostic mode. If no password is specified, the default password "diag" can be used to switch into diagnostic mode.
<i>Cycle time:</i>	The cycle time determines the frequency with which the data in the message page, system display and the faceplates on the operator station is updated.
<i>Flash rate:</i>	The frequency with which flash colors are updated in the displays.
<b>Overview display</b>	
<i>OVW</i>	● The configured overview display is displayed as an overview display in DigiVis.
<i>FGR</i>	● A graphic display is displayed as an overview display in DigiVis. Enter the name of the graphic display directly or choose it from the list.
<i>Ext. time server</i>	External stations with IP addresses 1 / 2 / 3 are time-synchronized by the operator station.
	 See also <b>Page D-50, External timer server</b>
<i>Enable</i>	Switch on time synchronization
<i>Short comment</i>	max. 159 characters
DRAWING HEADER and FOOTER	See <b>Page D-52, Documentation</b> and <b>chapter Documentation</b> .

#### D 4.4 Gateway station

The gateway stations **D-GS** are used to make data from the Freelance system available to other systems. In principle, all the data from the Freelance system can be read and written via a gateway station. In addition to each gateway station in a Freelance system, the relevant server software from the add-on packages DigiDDE, DigiOPC or DigiCSO must also be installed on the network. (If, for example, a gateway station of type *OPC gateway* is configured in a Freelance system, then the *Freelance OPC server software* must be installed on a PC that is linked on the network with the Freelance process stations).

Parameters are set for each gateway station to specify which tags and variables are available to the other system for reading and/or writing via the gateway.

Correlation to a physical station is carried out in the hardware structure. An assigned gateway station is represented by the label GWY. In the course of commissioning, the data specified is made available to the other stations by loading the gateway station.

Also refer to **chapter Hardware Structure**, network and assigning stations, **chapter Commissioning**, loading a project and loading objects, and the **manuals DigiDDE** and **OPC** and **DigiCSO**.



Select object GS in the project tree → *Project* → *Header...* or *double-click*

td006us.bmp

<i>Name</i>	Max. 4 characters
<i>Short description</i>	max. 12 characters
<i>Version</i>	Date and time of object creation
<i>Type</i>	Type of gateway. One of the following types may be selected: <ul style="list-style-type: none"> <li>DCP    A DCP gateway is used for interfacing to the Maestro UX process control system. It runs on a DCP02 CPU module and is installed in one of the process stations. Only the gateway type DCP can be assigned to a redundant gateway resource, and two DCP10 CPU modules must be used.</li> <li>DDE    A DDE gateway is needed to implement a link to another system via the DDE interface. This runs on the PC on which the DigiDDE server software is installed.</li> </ul>

- OPC An OPC gateway is needed to implement a link to another system via the OPC interface. This runs on the PC on which the Freelance OPC server software is installed.  
The linking of a Freelance system to Maestro NT is implemented via the OPC interface. In order to interface to Maestro NT an OPC gateway must be configured. A redundant interface can be achieved by configuring two OPC gateways in the project tree.
- TRN A TRN (trend) gateway is required when trend displays are used in the system without trend data acquisition (see DigiTool manual 1C, Standard Displays). The trend server software must also be installed. Trend gateways and trend servers run on the same PC.
- UNI A UNI gateway is required if data from the Freelance system is being made available via a DMSAPI interface (DataMessageApplicationInterface).

Ext. time server External stations with IP addresses 1 / 2 / 3 are time-synchronized through the gateway station.

 See also **Page D-50, External timer server.**

*Reset on  
fatal error...*

If an internal error in the sequence system is detected during processing of the user programs in the resource, the CPU module will suspend processing.

This parameter enables the system to be set up so that the CPU module can extricate itself from this type of error condition by executing a cold start.

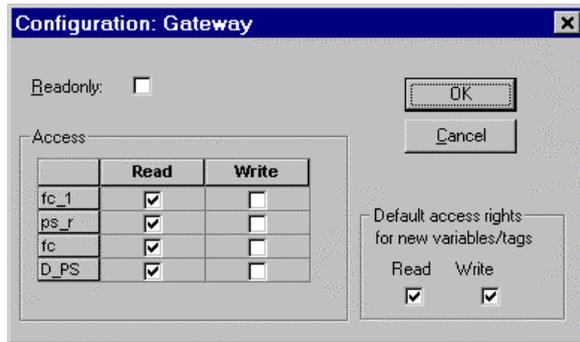
*Short comment*

Max. 159 characters

DRAWING HEADER  
and FOOTER

See also **Page D-52, Documentation** and **chapter Documentation.**

## D 4.4.1 Access rights of a gateway



td007us.bmp

*Read only*

- The gateway can access the process variables only in read-only mode. Any attempts to write to process stations and to variables and tags are ignored.

*Access*

For each process station already configured in the project, the type of access (read-only or read/write) is specified. If a process station is configured here as read-only, any write accesses to the variables and tags are ignored. If *read-only* mode is specified for the complete gateway, this overrides any write access defined here.

*Standard access for new variables/tags*

All newly-installed variables and tags are assigned by default the access rights configured here for this gateway.



Variables that are written via a gateway may not be written simultaneously through the process image. This results in the values of these variables not being capable of redundancy. If these variables are used in a redundant task they should be mirrored to other variables.

## D 4.5 Maestro configuration server MAESTRO-CS

A Maestro NT system can be used as an operator station for a Freelance system. The configuration data from a Maestro NT system is stored centrally in a database. To enable the Freelance data to be transferred to the Maestro NT system the MAESTRO-CS project object, which represents the configuration database, must be configured.

See also the **Interfacing to Maestro NT** manual.

td045us.bmp

<i>Name</i>	max. 4 characters
<i>Short description</i>	max. 12 characters
<i>Computer name</i>	Name of the PC in the network containing the Maestro NT configuration database.
<i>Project name</i>	Name of the Maestro NT project into which this Freelance system is to be integrated.
<i>Language</i>	Choice of language in which the Freelance project is to be installed, generally the language of the Maestro NT system.
<i>Short comment</i>	max. 159 characters.

### D 4.5.1 Maestro RTDS

The database of the Maestro NT system is configured by means of the project object Maestro-CS. RTDS (**RealTimeDataServer**) software packages are provided in a Maestro NT system for the data link. This data acquisition software does not normally run on the same computer on which the Maestro NT database is installed. Below a Maestro CS object, therefore, at least one RTDS object must be configured.

See also the **Interfacing to Maestro NT** manual.

td0451us.bmp

<i>Name</i>	max. 4 characters
<i>Short description</i>	max. 12 characters
<i>Computer name</i>	Name of the PC in the network containing the RTDS software for a Freelance system.

#### OPC gateway

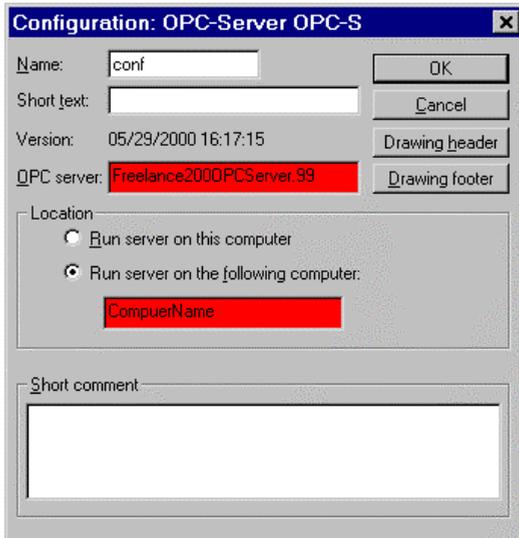
*Gateway 1 / 2* Enter the name of a type-OPC gateway station or select from a list after pressing the F2 key.  
A second OPC gateway only needs to be entered if a redundant link is to be established.

**MATCH** The configuration of the second gateway is matched with the configuration of the station entered under *Gateway 1*. Any access rights previously set up for this gateway will be lost.

*Short comment* max. 159 characters.

## D 4.6 OPC server (OPC-S)

The OPC server enables a Freelance 2000 system to be connected to other OPC servers. These OPC servers are third-party servers, and from the point of view of the other system they form the gateway to Freelance 2000; they perform this role in the same way as an OPC-type gateway station is the Freelance 2000 system's OPC server for an external system. For further details see the manual **OPC for Freelance 2000**.



td002us.bmp

<i>Name</i>	max. 4 characters
<i>Short description</i>	max. 12 characters
<i>Version</i>	Date and time the object was created
<i>OPC server</i>	max. 256 characters, name of OPC server
	 Example: Freelance2000OPCServer.92 is the default name of the Freelance 2000 OPC server, "92" is the station number of the OPC server or the gateway station of type "OPC server".
<b>Location</b>	Specifies the computer on which the OPC software is installed.
<i>Server running on ...this computer</i>	The OPC-Server software of the other system is running on the DigiTool PC.
<i>...the following computer</i>	Name or IP address of the PC on the network containing the other system's OPC server software
<i>Short comment</i>	Max. 159 characters

DRAWING HEADER  
and FOOTER

See **Page D-52, Documentation** and **chapter Documentation**.

Gross Automation, 1725 South Johnson Road, New Berlin, WI 53146, www.ssacsales.com, 800-349-5827

### D 4.7 User function block pool P-FB

New block classes are defined below this project object; these classes can then be used when configuring the user programs like the function blocks contained in the firmware. This means that the range of function block types in Freelance 2000 can be supplemented by user-defined function blocks (UFB). These function blocks are configured in the same way as programs and are displayed as function blocks. The resulting compressed display of programs permits the following:

- A clearer display of programs
- The creation of typical solutions
- Enhanced and reduced documentation

For working with User Function Blocks a distinction is made between classes and instances. The UFB class determines the functionality and visual appearance of a user function block. The UFB class also contains the entire program generated by the user - complete with its functions, function blocks and variables - along with the faceplate and parameters mask.

The user function block program is configured in the project tree under *pool of user-defined function blocks, P-FB*. Each UFB class receives a freely-assigned class name, under which it is called in other programs. The program can be configured in Function Block Diagram FBD, Ladder Diagram LD or Instruction List IL.

The faceplates are configured in a dedicated faceplate editor, corresponding to the graphics editor.

User function block are not made available before the plausibility check under the project tree. They can then be accessed through: *Blocks* → *User blocks*.

From there, user function blocks can be integrated into other programs or called from any other programming languages: from IL, LD, FBD and SFC programs.

User function blocks can be locked by the user with a password. Following this, they only still appear through their outer form. Embedded function blocks are invisible.

The configuration of the user-defined function blocks is described in **Engineering Manual, IEC 61131-3 Programming, User Function Blocks**.

## D 4.8 Global display pool D-POOL

Displays and logs set up under this project object should in principle be available on all operator stations. In fact the objects from the display pool are loaded only on those stations that have also been granted access to the process data needed in these displays and logs.

 Any modifications to displays in the general display pool must be loaded separately into all the operator stations.

Once an object from the display pool has been selected, *Show target stations* can be chosen from the shortcut menu to display a list of operator stations that this program has been loaded to.

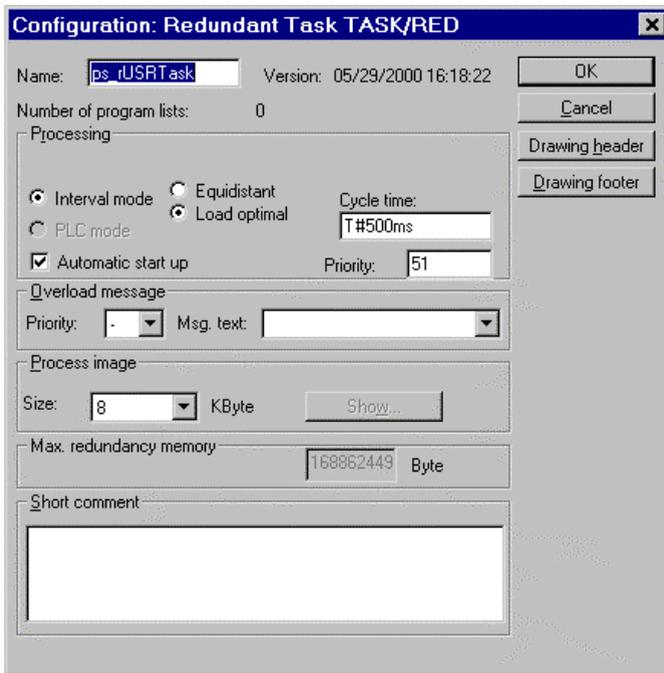
## D 4.9 Task TASK and redundant task TASK/RED

The task object comprises all the subordinate program lists and SFC programs. The tasks determine how quickly the programs are processed on the resource. A distinction is made between user and system tasks.

In user tasks, the programs are processed cyclically; in system tasks, certain events in the resource or commissioner actions determine whether a certain system task is executed. The programs within a task are controlled by a program list or by an SFC program.

The number of user tasks is limited to 9 tasks per Process Station/FieldController (FieldController default setting: 3 tasks) . A maximum of 8 cyclic tasks and one default task can be configured. The default task is always executed if none of the other tasks is executed (either cyclically or once only).

Also refer to **chapter Commissioning, Loading the project** and **commissioning the objects**.



di0919uk.bmp

Name:	Max. 12 characters
Version	Date and time of object creation
Number of program lists	The number of assigned program lists is displayed.
<b>Processing</b>	
<i>Interval mode</i>	The task is performed in time intervals. You can select between <i>equidistant</i> or <i>load optimal</i> .
<i>Equidistant</i>	The task is triggered at regular intervals, i.e. the first time the task is computed, a series of equidistant times is specified for triggering task computation.
<i>Load optimal</i>	In contrast to equidistant triggering, the times at which the task computes are continually recalculated during execution. Each task computation decides the time of the next computation, as the configured interval time is added to the current system time. In normal operation equidistant tasks do not differ from those which are load optimal in execution. The distinction only becomes relevant when brief load peaks in the system cause a task to be delayed in execution. Equidistant tasks will keep to the defined time scale, while load-optimal tasks stretch the intervals slightly in the event of overload, thus relieving the system.

<i>Cycle time</i>	Input format: e.g. T#4s500ms (=4.5s) d = days, h = hours, m = minutes, s = seconds and ms = milliseconds. The interval time must be at least 5 ms and a multiple of 5 ms.
<i>PLC mode</i>	At most one user task can be made the <b>default task</b> , in that it computes in PLC mode. It has the lowest priority of all cyclic tasks, and starts itself again at once after a run. This leads to the fastest possible cyclic processing of the underlying programs. Because of its low priority (50) it can be interrupted by all other tasks. The default task is thus only in execution when no other task is ready to compute.
<i>Automatic start up</i>	<input checked="" type="checkbox"/> Defines whether the task starts automatically (thus causing the state "Running") after the loading or the starting of the resource.  The default is automatic start.
<i>Priority</i>	The priority (not to be confused with message priority) controls the processing of several tasks which are in the state "running" at this time. The multitasking of several tasks of the same interval time is controlled by the priority. The lowest priority that can be assigned for user tasks is 51; the highest is 99.
<b>Overload message</b>	A task triggers an overload message if it detects that an interval time is overshoot. An interval time overshoot takes place if a task has not been executed within double its interval time.
<i>Priority:</i>	None, no overload message is generated. <b>1 to 4</b> message priorities, also refer to <b>Engineering Manual, Operator Station, Messages and Hints</b>
<i>Msg. text:</i>	Max. 8 characters
<b>Process image</b>	See Page D-44, Process image and <b>chapter Commissioning</b>
<i>Size</i>	4 and multiples of 4, up to 32 KB
<i>SHOW...</i>	Shows the process image of a task. See <b>Page D-44, Process image and chapter Commissioning</b>
<i>Max. redundancy memory required</i>	Maximum size of memory required by the redundant task for the transfer of redundancy data. The redundancy memory is only displayed in redundant and plausible tasks.
<b>Short comment</b>	Max. 159 characters
DRAWING HEADER and FOOTER	See also <b>Page D-52, Documentation</b> and <b>chapter, Documentation</b> .

### D 4.9.1 Process image

The process image is an intermediate storage memory for task variables. All variables written via the process image are identified by @ in the programs.

Access via process image (indirect)                      @Engine\_E1243\_On

Access bypassing process image (direct)              Engine\_E1243\_On

The process image is read at the start of task calculation. Then the entire user program, the sum of all program lists and programs belonging to this task are calculated once. The current states and sizes of the task variables are then rewritten from the process image to the global variables. In the next interval or task run, the variables from the last task calculation are available. In practice, indirect access via the process image is undesirable since the aim is to achieve the shortest possible reaction times. For this reason, you can decide for each variable whether access is permitted via the process image.

-  Access via the process image increases clarity of configuration. This path should always be preferred instead of the configuration bypassing the process image.
-  In a redundant task the variables may only be accessed via the process image. Write accesses which are not made via the process image are not transferred to the secondary, so that the variable would change value in a toggle.
-  Variables that are written via a gateway should not be written through the process image as this state can be overwritten by the task.

### D 4.9.2 Predefined system tasks

All system tasks are grouped in the project tree under \*.SYSTask node. The predefined Tasks let you start and stop processing sections of a user program as a reaction to system state changes. The event which triggers the task, e.g. a certain state transition of the resource, or an error in the user program, is always defined. All predefined system tasks are created in the resource after initial program loading. After the resource is initialized, the system tasks are automatically started, but do not compute until the event assigned to them occurs.

The following tasks are predefined:

- \*.**ColdSt** [Task] [Once]
- \*.**WarmSt** [Task] [Once]
- \*.**Run** [Task] [Once]
- \*.**Stop** [Task] [Once]
- \*.**Error** [Task] [Once]
- \*.**LatCSnd**[Task] [Cyclic,T#1s]
- \*.**LatCRcv**[Task] [Cyclic,T#1s]
- \*.**RedSt** [Task/Red][Once] (only for redundant process stations)



No other user task is calculated during the computation of a once-only task. The maximum permitted computation time for this task (i.e. the time from beginning of execution to end of execution) is therefore limited to 10 seconds. Operations which take more time, e.g. waiting for process data in a loop, should therefore be configured in one of the cyclic user tasks.

Further information may be found in the relevant **Engineering Manual, Process Station**.

#### **Cold start task ColdSt**

The cold start task is executed once when the resource performs a cold start. The task is executed on transition from *cold start* or *cold start stopped* to *running* state.

During a cold start, all the variables and working data in the function blocks are initialized. The initial variable values can be specified singly in the variable list for each variable.

The I/O module and device outputs are set to safety values. After the computation of the cold start task, the outputs are then rewritten depending on the programs.

Also refer to **Engineering Manual, IEC 61131-3 Programming, Variables** and **chapter Commissioning, State diagram or Resource**.

### Warm start task WarmSt

The warm start task is executed once when the Resource performs a warm start. The task is executed on transition from *warm start* or *warm start stopped* to *running* state. It is computed before the cyclic user tasks are started.

The process station always performs a warm start when power is restored after a power failure. A cold start may be forced, depending on the duration of the power failure. (Trigger Cold Start function)

When a power-fail signal is received, the output modules first hold their value. If the voltage restabilizes after a specific period of time (15 ms), task processing resumes where it was interrupted. If not, a warm start is executed on the modules and the configured safety values are output.

Also refer to **chapter Hardware Structure, Parameters of I/O modules, Start and shut-down characteristics of modules** and **chapter Commissioning, Loading entire station and Task**.

All system variables are retained so that the functions and function blocks can continue operating with the same values as before the warm start.

In the function blocks, the inner states are retained and the function blocks resume after the warm start from where they were interrupted. In addition, special block-specific features are included in some function blocks.

- Blocks which use the task interval time, e.g. controllers, continue working smoothly.
- For blocks with monitoring times, the time of the voltage failure is counted. This means that, e.g. a timer set to 30 min. will run a total of 30 min. even there is a power-fail of 5 min.
- The gap is marked in acquisition blocks such as trend or disturbance course log acquirers.
- Analog filters such as INTEG (I component) and PT1 components continue their calculation from where they were interrupted.

Also refer to **Engineering Reference Manual, Functions and Function Blocks, General, Warm start characteristics of function blocks**.

 The number of warm starts are counted in a system variable (xxxx.NoPowerFail) which is reset to zero after a cold start. Another system variable (xxxx.PowerOffTim) contains the time of the last power-fail signal.

Also refer to **chapter Commissioning, State diagram of Resource**, and **Engineering Manual, IEC 61131-3 Programming, Variables, System variables**.

### Run task Run

The Run task is executed once when the resource changes state from *Stopped* to *Running*. All user tasks are only started after the run task ends.

Also refer to **chapter Commissioning, State diagram of resource**.

**Stop task Stop**

The stop task is executed once when the resource changes state from *Running* to *Stopped*. The stop task is only triggered if all the other tasks have stopped. A restart of the resource is only possible if the stop task was fully executed.

Also refer to **chapter Commissioning, State diagram of resource**.

**Error task Error**

The error task has the highest priority (100) and is used for handling errors in user programs caused by user programs. It is executed once if an error is detected in a user program. If errors cannot be eliminated, the error-producing task reverts to the "*unrunnable*" state; if the errors can be eliminated, the error-producing task can continue, provided the automatic error correction is switched on for the resource. Due to its high priority, it cannot be interrupted by other tasks. Execution of the error task can be suppressed.

Also refer to **chapter, Commissioning, Error Handling**

**Lateral communication send task LatCSnd**

All variables are automatically transferred to other resources if they are released to be sent with the Export = Yes attribute in the corresponding (local) resource.

The lateral communication required for sending is automatically executed by the **LatCSnd task** (Lateral Communication Send task).

Also refer to **chapter Commissioning, Export-Variables of resource** and **Engineering Manual, IEC 61131-3 Programming, Variables**.

 The total quantity of variables the resource can make externally visible is limited to 1400 bytes.

**Lateral communication receive task LatCRcv**

All variables can be read automatically in other resources if they were released by the attribute Export = Yes in the resource to be sent.

The lateral communication required to read and receive is automatically performed via the **LatCRcv-Task** (Lateral Communication Receive-Task).

Also refer to **chapter Commissioning, Export-Variables of resource** and **Engineering Manual, IEC 61131-3 Programming, Variables**.

**Redundancy start task RedSt (for redundant resource only)**

The RedSt task is executed once directly after a redundancy toggle before the user task's first computation. It can cause execution of user programs which should react especially to the redundancy toggle.

### D 4.9.3 Program list PL

The Program List object contains all subordinate programs and controls the processing sequence of these programs. The programs are processed depending on their consecutive number in the project tree. The higher-order task determines how fast the programs are processed on the CPU module of the process station. The programs are written in the Function Block Diagram (FBD), the Ladder Diagram (LD) or Instruction List (IL) language.

See **chapter Commissioning**, Loading the Project and Commissioning the objects.



di0921uk.bmp

<b>Name:</b>	Max. 12 characters
<b>Version</b>	Date and time of object creation
<b>Number of programs</b>	The number of the subordinate programs is displayed
<b>Processing sequence</b>	Indicates the position within the task at which this program list is executed
<b>DRAWING HEADER and FOOTER</b>	See also <b>Page D-52, Documentation</b> and <b>chapter Documentation</b> .
<b>Processing</b>	
<input checked="" type="checkbox"/> ON	The subordinate programs are processed.
<input type="checkbox"/> ON	The subordinate programs are not processed.
	 The default is: <i>On</i> .
<b>Short comment</b>	Max. 159 characters.

#### D 4.9.4 Programs on the operator station: Displays and logs

In principle, displays and logs are the programs in an operator station. Like the programs on the process stations, the displays are updated cyclically.



Select object in the project tree → *Project* → *Header...*

td008us.bmp

Name:	Max. 12 characters
Version	Date and time of object creation
Type	Dependent on the project object selected:
	SFCP Sequential function chart display
	OPL Operator plant log
	FGR Graphic display
	GRP Group display
	TR_D-OS Trend display
	TS_D-OS Time scheduler display
	SSL1 Signal sequence log 1
	SSLN Signal sequence log N
	DCL_D-OS Disturbance course log
	O VW Overview display
	SFC SFC program
	IL IL program
	FBD FBD program
	LD LD program
Cycle time	Time interval at which the values in a display or log are updated
<b>Short comment</b>	Max. 159 characters

### D 4.9.5 Structure node (STRUCT)

Structural element of the operator station. To provide a clearer overview the displays and logs from an operator station can be grouped together with this project element.

The name must be no longer than 12 characters.



Select operator station in the project tree → *Edit* → *Insert next level* → *Structure node* → Specify a name

### D 4.10 External timer server

Within Freelance 2000 systems all internal stations are time-synchronized by the first process station or by a timer (radio clock).

External stations that are connected by means of a router or modem can be time-synchronized by an operator station PC, a gateway PC or a DCP gateway. To enable this facility the function "Ext. time server" must be activated in the resource configuration (select operator station or gateway in the project tree, → *Project* → *Header*). By specifying the IP address, up to three stations can be time-synchronized by a resource.

The following forms of time synchronization are possible:

<b>Resource type</b>	<b>can synchronize</b>
Operator station (D_OS) →	external process stations D_PS
DDE gateway →	external process stations D_PS
OPC gateway →	external process stations D_PS
UNI gateway →	external process stations D_PS
DCP gateway →	external DCP gateway stations and Maestro UX workstations

### Deactivating time synchronization

If several Freelance 2000 applications are running on a Freelance 2000 PC with external time synchronization activated, time synchronization continues until all the Freelance 2000 applications have finished running.

Time synchronization is not finished if only one application is completed or if on-line mode is switched off.

td004us.bmp

- Ext. time server** External stations with IP addresses 1 / 2 / 3 are time-synchronized through the operator station.
- Enable** Switches on time synchronization
-  If the *Enable* flag is not set, then the time synchronization messages will not be generated.
- IP address** Enter the IP addresses of the stations to be time-synchronized by means of this function.  
*If the Onet broadcast address is entered when connecting to a Maestro UX system, then all DCP gateways linked in this network will be synchronized.*

## D 5 General Functions of the Project Tree

### D 5.1 Save

 → *Configure* → *Project* → *Save*

This function saves the project during configuration. The project is saved to the project file.

### D 5.2 Documentation

 Select level → *Configuration* → *Project* → *Documentation*

The project documentation function is started. Depending on the position in the project tree, all the selected objects or all the subordinate objects are documented.

 Compilation of documentation jobs and the scope of documentation defined are described in **chapter Documentation**.

### D 5.3 Check, check all

**Check all** is used to check the selected object and all objects positioned below it for correctness (or plausibility). **Check** is used to check only those objects in the project tree that are not yet correct. Program code is generated automatically for all correct objects, and this can be loaded to the relevant stations during the subsequent commissioning phase. Incorrect objects are displayed accordingly in the color set for that purpose.

In the case of long error lists it is advisable to perform plausibility checking in smaller sections. Each node of the project tree can be checked separately. Individual blocks within a program can also be checked.

 Select level → *Configuration* → *Project* → *Check*

The selected object and the subordinate incorrect objects are checked for plausibility, i.e. for errors, missing entries or contradictions. The errors are listed in a window and must then be eliminated.

 Select object → *Configuration* → *Project* → *Check all*

The object selected and all the subordinate objects are checked for plausibility. The errors are displayed in an error list and must then be eliminated. The plausibility check can take a few minutes.

### D 5.3.1 Jumping directly to error points after plausibility check

After a plausibility check list has been newly created with *Check* or *Check all*, all errors that have been detected are displayed for the user in a list box.

- Double-clicking the left mouse button on a marked message or the CURRENT ERROR and NEXT ERROR buttons triggers a jump to the object causing the error.
- A marked object is identified through being depicted in the "system selection" color (light blue) or being framed with a border.
- In so far as is possible for the editor concerned, the marked object is positioned in the middle of the screen.

The destination of a jump is exactly the same whether the plausibility check was called up in the project tree or the editor. If jumping to an editor page which previously contained a selection (only possible after a plausibility check within an editor), this selection is lost by the plausibility check jump.

### D 5.3.2 Classes of plausibility errors

Three classes of plausibility errors are distinguished:

#### Error

- Some problem will occur.
- The project cannot be loaded in this state.
- The error must be corrected.

#### Warning

- The configuration is inconsistent, but the project can be loaded.
- The person responsible for configuration must decide whether the error needs to be rectified or whether the project can run effectively despite this message.

#### Hint

- Contains information for the user which does not necessarily need to be followed.
- No action is generally required on the part of the person carrying out the configuration process, s/he is merely being informed that certain inaccuracies have been discovered by the system or that minor adjustments have been carried out.

### D 5.3.3 Structure of the plausibility check messages

The messages are formed as follows:

<Number> <Label for class of error> <Message text>

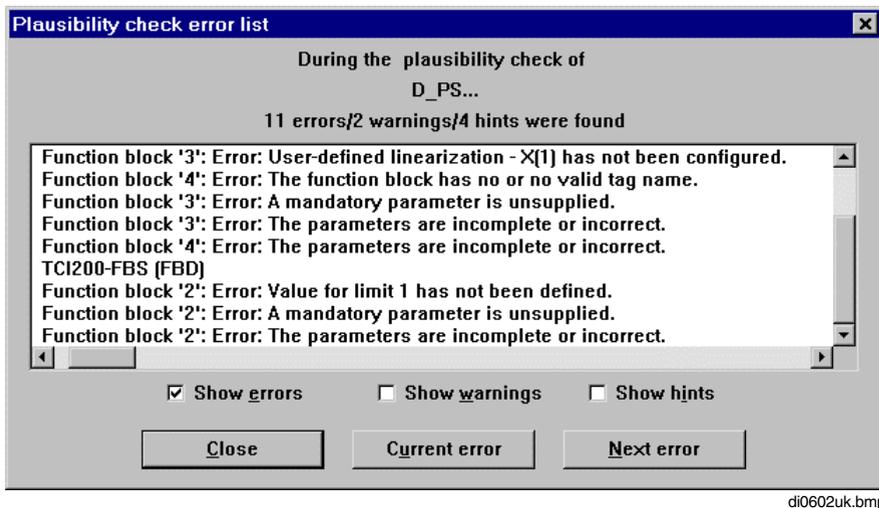
The following arrangements apply to the various different error classes:

[xxxx.2.xxxxx] Error: <Message text>

[xxxx.3.xxxxx] Warning: < Message text >

[xxxx.4.xxxxx] Hint: < Message text >

### D 5.3.4 Error list display



di0602uk.bmp

- The classification of an error can be recognized both from its number and the introductory text.
- The different message classes can be selectively faded in or out.
- For the whole time the project is being processed the most recent setting of the check boxes is preserved.

## D 5.4 Show error list



Select level → *Configuration* → *Project* → *Show error list*

The error list contains all the errors in the selected objects occurring after a plausibility check. The list can be printed out via the Documentation function. See **chapter Documentation**.

## D 5.5 Object header

The general parameters of an object can be modified at any time.

It is the same as specified when the object was added. Choose an object use:



*Configuration* → *Project* → *Header...*



In some objects such as resource, task or program list, selection is by double-clicking the mouse. On other objects such as FBD program, graphic display or trend display, you can access the editor or extended parameter-definition dialog directly.

See **Page D-25, Configuring the Project Objects**.

## D 5.6 Comments on a project object



*Configuration* → *Project* → *Comment*

Use the comment editor to create or modify a free text for each object.

This multi-page text can also be used to describe the program or the process technology and can be output together with the object using the Documentation function.

See **chapter Project Manager, Comment**.

## D 6 Project Tree Options

### D 6.1 Hardcopy



*Options → Hardcopy*

Dumps the screen contents to the printer.

### D 1.2 Long or short form in the state line



*Options → Long state line*

**Long form:** If you select object E3122:  
the state line displays: **project/E1/E300/E3100/E3120/E3122**  
i.e. the names of the objects are displayed.

**Short form:**

- C Configuration level
- R Resource
- T Task
- P Program list
- B Program, Display
- A Sequential function chart
- S Step
- T Transition
- S Structure node

The **short form** is the default setting.

### D 6.3 Colors in the project tree

 *Options* → *Colors*

The colors of nodes, connecting lines, mandatory parameters and fields can be modified in the drawing footer. The default settings for the colors can be restored if necessary.

Also refer to **Page D-11, Setting the node colors**

### D 6.4 Lock / unlock user function block

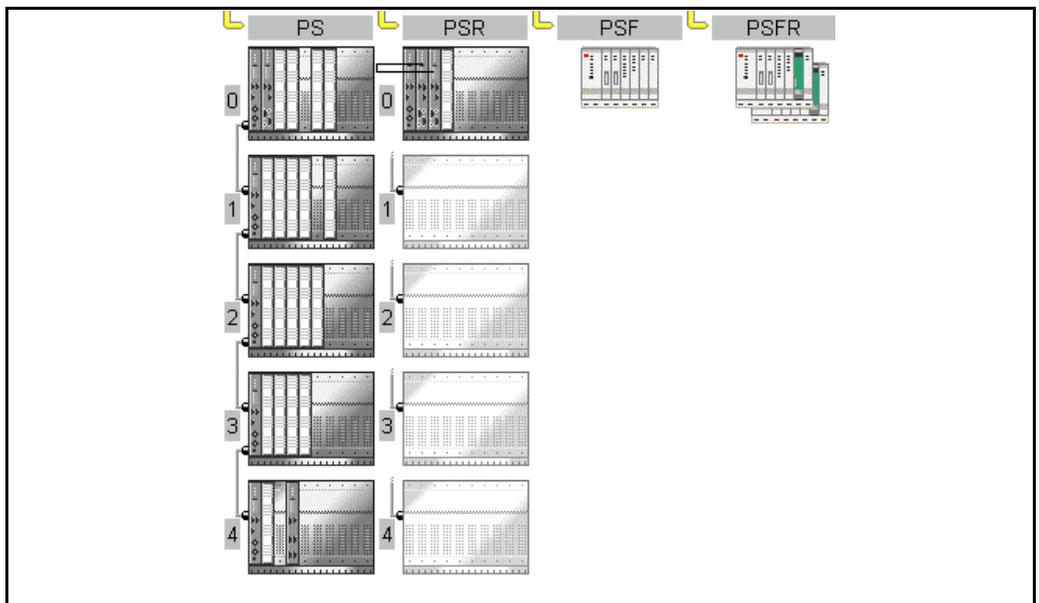
 *Options* → *Lock / Unlock UFBClass*

A user function block class (UFB-CLASS) can be locked or unlocked.

For details see **Engineering Manual, IEC 61131-3 Programming, User Defined Function Blocks.**



# E Hardware Structure





## Contents

<b>E 1</b>	<b>General Description - Hardware Structure .....</b>	<b>E-5</b>
<b>E 2</b>	<b>Hardware Structure User Interface .....</b>	<b>E-8</b>
E 2.1	Tree view .....	E-8
E 2.2	Object list of the hardware structure.....	E-8
E 2.3	Graphic view.....	E-11
E 2.3.1	System view .....	E-12
E 2.3.2	Station view of the process station as Rack system .....	E-13
E 2.3.3	Station view of the process station with FieldController.....	E-14
E 2.3.4	Station display of a process station with redundant FieldController .....	E-15
E 2.4	Detail view of the Profibus master.....	E-16
E 2.4.1	Detail view of a device.....	E-17
E 2.4.2	Detail view of a module .....	E-18
E 2.5	Menu structure of the hardware structure .....	E-19
<b>E 3</b>	<b>General Functions of the Hardware Structure .....</b>	<b>E-20</b>
E 3.1	Context menu .....	E-20
E 3.2	Cut, Copy and Paste .....	E-20
E 3.2.1	Cut.....	E-20
E 3.2.2	Copy.....	E-21
E 3.2.3	Insert.....	E-21
E 3.2.4	Paste .....	E-21
E 3.2.5	Delete .....	E-21
E 3.3	Clipboard viewer.....	E-22
E 3.4	Export block.....	E-22
E 3.5	Import block.....	E-23
E 3.6	Check all.....	E-23
E 3.7	Check .....	E-23
E 3.8	Search .....	E-23
E 3.9	Cross references.....	E-24

<b>E 4</b>	<b>Editing the Hardware Structure.....</b>	<b>E-25</b>
E 4.1	Insert new objects.....	E-25
E 4.1.1	Insert PC components in the operator station.....	E-26
E 4.1.2	Insert gateway stations.....	E-27
E 4.1.3	Insert process stations.....	E-29
E 4.1.4	Default names of the objects.....	E-30
E 4.1.5	Default names of the modules.....	E-30
E 4.1.6	Default names of the master objects.....	E-30
E 4.1.7	Default names of the slave objects.....	E-30
E 4.1.8	Default names of the slave modules.....	E-30
E 4.1.9	Standard names of the Hart Channels.....	E-31
E 4.1.10	Default names of the modules.....	E-31
E 4.1.11	Default name of the serial interface.....	E-31
E 4.1.12	Default name for the remote process controller.....	E-31
E 4.2	Resource allocation.....	E-32
E 4.3	Activate/deactivate objects.....	E-33
E 4.4	Configuration of the communications connections.....	E-34
<b>E 5</b>	<b>I/O Editor.....</b>	<b>E-36</b>
E 5.1	Call the I/O Editor.....	E-37
E 5.2	Edit components.....	E-40
E 5.2.1	Insert component.....	E-40
E 5.2.2	Delete component.....	E-42
E 5.2.3	Copy.....	E-42
E 5.2.4	Paste contents.....	E-42
E 5.2.5	Move position.....	E-43
E 5.2.6	Change data type.....	E-43
<b>E 6</b>	<b>Network Configuration.....</b>	<b>E-44</b>
E 6.1	IP address and resource ID.....	E-44
<b>E 7</b>	<b>General Parameters of the Hardware Structure Objects.....</b>	<b>E-46</b>
E 7.1	HWSYS system object.....	E-46
E 7.2	VIS operator station.....	E-47
E 7.3	GWY gateway-station.....	E-49
E 7.4	Process station.....	E-50

## E 1 General Description - Hardware Structure

Within the hardware structure the resources defined in the project tree are allocated to the hardware actually required. A system essentially consists of the process, operator and Gateway stations. These stations are allocated to resources according to IEC 61131-3. The resources serve as structuring elements in the project tree for allocation of the application program parts and displays to the hardware actually required.

The D-PS resources which are configured in the project tree are assigned to the process stations in the hardware structure. A process station can have a rack-based set-up, that is, conventional with input and output modules assembled in module assembly frames for process control, or by the FieldController through one or more field buses with intelligent field devices and/or remote I/O.

In the conventional process station, the I/O modules are mounted in modules supports (Racks). The CPU module processes all programs of this resource and is simultaneously the communication module to the system and the I/O modules. The **Process station** consists of the central unit and 4 I/O units max. The central unit is always the station with the CPU module. The I/O units do not need an own CPU module: they can accept other I/O modules so that their numbers can be expanded (from 8 for the central unit) to a maximum of 44 for a process station. All units need a link module for feeding in the power supply and an identical rack for holding the modules.

Depending on the FieldController, the process station is now provided in two versions: the FC and the redundant Industrial<sup>IT</sup> Controller AC 800F. The FieldController takes the fieldbus modules and makes it possible to connect various fieldbuses. The FieldController basic unit consists of the case and the main board, which together form a unit which can be equipped with various modules. The module for the power supply and an Ethernet module for connection to the system bus are absolutely necessary. Both modules are available in various designs. A FieldController can be equipped with a maximum of 4 fieldbus modules selected from CAN, Profibus and serial modules.

The CAN module allows the connection of a maximum of 5 I/O units and thus the connection of 45 I/O modules in the way in which they are also used in the conventional process station.

Each Profibus module allows the connection of a Profibus line, i.e. the connection of a maximum of 125 slaves. Each of these slaves can also be modular, i.e. contain a maximum of 64 modules. The serial module has 2 interfaces which can be occupied at option with the Modbus master interface protocol, the Modbus slave interface protocol, the telecontrol interface protocol, the Protronic interface protocol or the Sartorius scale interface protocol.

The number of process and operator stations is not limited. The information given in the boot parameters of the process station determines how many operator stations or gateways will be supported by the process station.

The **Operator stations** are commercial PCs in which the software program DigiVis has been installed for process visualization. All displays and logs are configured with the DigiTool software program and loaded into the operator station. Data acquisition for recording and trend display takes place in the process station, and all display and archiving in the operator station. Each process station can communicate with up to 10 operator or gateway stations.

D-GS resource stands for a **Gateway** connecting the system to other systems. Actually, there exist gateways for coupling Maestro NT, **Maestro UX**, for a free number of **DDE clients**, such as Microsoft Excel or Visual Basic applications, for any **OPC clients** and for individually developed applications (**DMS API**).

D-ES resource stands for an **Engineering station**, i.e. the configuration tool DigiTool. It is also displayed in the hardware structure; this display is only for documentation purposes.

Each object can subsequently be parameterized. In addition to the object name given by default setting, a short text and a long text can be allocated.

There are no switches on the I/O module: all module settings can be configured in the hardware manager. To each channel of an output module a safety value can be allocated; this is output in the case of a CPU module power fail. The update times can be set individually and depend on the module type (analog or digital), the transmission rate set on the process station (100/500 kbit/sec) and the number of units connected to the central unit.

The hardware can be configured in two different view areas. In the tree view area the complete system can be configured in a tree structure.

In the graphic view area the hardware can be configured in several display levels:

The **system view** displays the complete hardware structure. In the system view the individual stations like the process station, FieldController or operator station are activated and allocated to the project tree resources.

The **station view** displays the complement of a process station or a FieldController. In the station view the modules or FieldController modules are allocated to slots.

The **detail view** displays further information on the modules of a process station or the modules of the FieldController so that they can be identified more easily. The detail view of the Profibus master displays a bus line with the slaves configured on the bus. The slaves are represented using bitmaps which are specified in the device database file. If no bitmap files are specified in the device database file of the respective vendor, standard bitmaps are displayed. However, it is possible subsequently to assign vendor-specific bitmaps. The detail view of the Profibus slave displays a device view in the form of an individual bitmap file which can be assigned in the parameter dialog of the slave.

The variables which are to be linked with the process via the I/O modules are entered in the I/O editor. A number of I/O components is provided by each module or slave according to the number of channels. Via these I/O components it is possible to directly use the information in programs and graphic displays. If the I/O component is also to appear in the variable list, a variable name must be indicated. The I/O component name is composed of the object name (16 characters) and the component name (16 characters), thus making available a total of 32 characters.

Diagnosis components are available in addition to the I/O components.

Through the **network configuration**, communication addresses, so-called resource IDs (previously station numbers), are assigned to the resources. IP addresses of the units assigned to the resources are also indicated in the network setting. When the project objects are loaded into the stations, the corresponding application program parts (as allocated in the project tree of the individual resources) are loaded into the resources. The resources or individual parts of the application program are loaded with DigiTool commissioning.

Below the tree view, it is possible to superimpose another tree from which **templates** which have already been created can be transferred into the project. The templates are configured hardware objects, the specific parameter settings of which are to be reused. An application might, for example, involve preserving an already created slave configuration together with the parameter and I/O configuration created for certain devices, or pertaining to all slave modules, in order that the dialog created, based on the information from the device database file, can be used again and again.

The Windows NT operating system allows several resources to run simultaneously on one PC. Hence, parallel DigiVis and DigiDDE operation on one and the same unit is possible. These two resources have the same IP address in network setting but they can still be addressed because of their different resource IDs. From the point of view of the system, they do not differ from two resources running on different PCs. Even if several resources have been assigned to the same physical unit, each resource has an individual representation in the hardware structure.

As well as specifying the communication connection, the configuration of the hardware also serves the purpose of documentation and plausibility checking. Thus a comparison can be made between what was actually mounted and what was configured. Incorrectly inserted or missing objects are marked in different colors in Commissioning Mode of the hardware structure.

## E 2 Hardware Structure User Interface

### E 2.1 Tree view

The hardware structure tree view displays all hardware structure objects. Beginning with the system object, other hierarchical levels can be entered down to the device level.

The stations are entered via a position number showing the respective station in the graphic view display.

Position query is made every time a station or a gateway is inserted.

Each object features object parameters such as name, short text and long text. The parameters relating to the objects are described in the Engineering manuals of the process stations. See **Engineering Manual, Process station**.

Individual object names are allocated automatically so the user does not need to allocate names. Names can be subsequently configured.

### E 2.2 Object list of the hardware structure

Abbreviation	Object Description
HWSYS	System Object
HWSTR	Structural element within the template manager
PS	Process Station
PSR	Redundant Process Station
AC 800F	Industrial <sup>IT</sup> Controller
AC 800FR	Industrial <sup>IT</sup> Controller Redundant
FC	FieldController
FCSELECT	FieldController <i>select</i>
VIS	Operator Station
GWY	Gateway
SR_SNDEV	Send and Receive, send interface
SR_RNDEV	Send and Receive, receive interface
PROFI_M_DEV	Profibus master
PROFI_S_DEV	Profibus slave
Hart	Hart-Device
MODS_DEV	Modbus Slave
MODM_DEV	Modbus Master
FWK_DEV	IEC 870-5 Telecontrol module
FWK_DEV_M	IEC 870-5 Master telecontrol connection
FWK_DEV_S	IEC 870-5 Slave telecontrol connection

Gross Automation, 1725 South Johnson Road, New Berlin, WI 53146, [www.ssacsales.com](http://www.ssacsales.com), 800-349-5827

Abbreviation	Object Description
SCALE_DEV	Sartorius Scale
FLRC_MSL	Freelance Remote Control / Modbus Slave (Protronic)
DNETP	CAN master
SA 801F	115 V AC / 230 V AC Power supply
SD 801F	24 V DC Power supply
AM 801F	Battery module
AM 802F	Debug module
EI 801F	10Base2 (Cheapernet) Ethernet module
EI 802F	AUI Ethernet module
EI 803F	Ethernet module, twisted pair
FI 810F	CAN-3 module
FI 820F	Serial module
FI 830F	Profibus module
SA 801FR	115 V AC / 230 V AC Redundant power supply
SD 801FR	24 V DC Redundant power supply
AM 801FR	Redundant battery module
AM 802FR	Redundant debug module
EI 801FR	10Base2 (Cheapernet) Redundant Ethernet module
EI 802FR	AUI Redundant Ethernet module
EI 803FR	Redundant Ethernet module, twisted pair
FI 810FR	CAN-3 redundant module
FI 820FR	Redundant serial module
FI 830FR	Redundant Profibus module
DFP 01	115 V AC / 230 V AC power supply
DFP 02	24 V DC power supply
DFA 01	Battery module
DFE 01	10Base2 (Cheapernet) Ethernet module
DFE 02	AUI Ethernet module
DFE 03	Ethernet module, twisted pair
DFM 01	Profibus module
DFM 02	Serial module
DFM 03	CAN- 3 module
DAI 01	Analog Input 16 x 0/4...20 mA, Ri=50 Ohm
DAI 02	Analog Input 16 x 0..10 V DC
DAI 03	Analog Input 16 x 0/4 ... 20 mA, Ri=250 Ohm
DAI 04	Analog Input 8 x PT100 / mV
DAI 05	Analog Input 16x0/4...20 mA, MU infeed

<b>Abbreviation</b>	<b>Object Description</b>
DAO 01	Analog Output 16 x 0/4...20 mA
DCO 01	Communication Module 4 x RS 485/RS 422 or RS 232 C
DCP 02	CPU Module
DCP 10	CPU Module for Redundancy
DCP 02GWY	DCP Gateway CPU for Maestro UX Gateway
DCP 10GWY	DCP Gateway CPU for Maestro UX Gateway
DDI 01	Digital Input 32 x 24 V DC
DDI 02	Digital Input 16 x 24...60 V AC/DC
DDI 03	Digital Input 16 x 90...230 V AC
DDI 04	Digital Input 28 x 24 V DC NAMUR initiators
DDI 05	Digital Input 32 x 115/230 V AC
DDO 01	Digital Output 32 x 24 V DC, 0.5 A
DDO 02	Digital Output 16 x 24...230 V AC/DC
DD O03	Digital Output 16 x 24...60 V AC/DC, read back
DDO 04	Digital Output 16 x 115...230 V AC, read back
DFI 01	Frequency Input 4x f <= 45 kHz
DLM 01	Link Module for Supply
DLM 02	Link Module for Redundant Supply

### E 2.3 Graphic view

The hardware structure graphic view features several views with different information within their detailed levels.

<b>System view</b>	Rapid overview over the global structure, e.g. number of operator and process stations, number of FieldControllers, number of gateways.
<b>Station view</b>	Immediate information concerning equipped/free slots and plugged-in module types. Intuitive equipment by simple double click.
<b>Detail view</b>	The detail view differs depending on the object selected and displays a detailed display of the object. Displayed are the module type (status information only online), the slaves connected, their vendor, model name and bus address (status information and diagnostic information only online) or only the slave connected in its device view with some information such as vendor, model name and bus address (status information and diagnostic information only online).

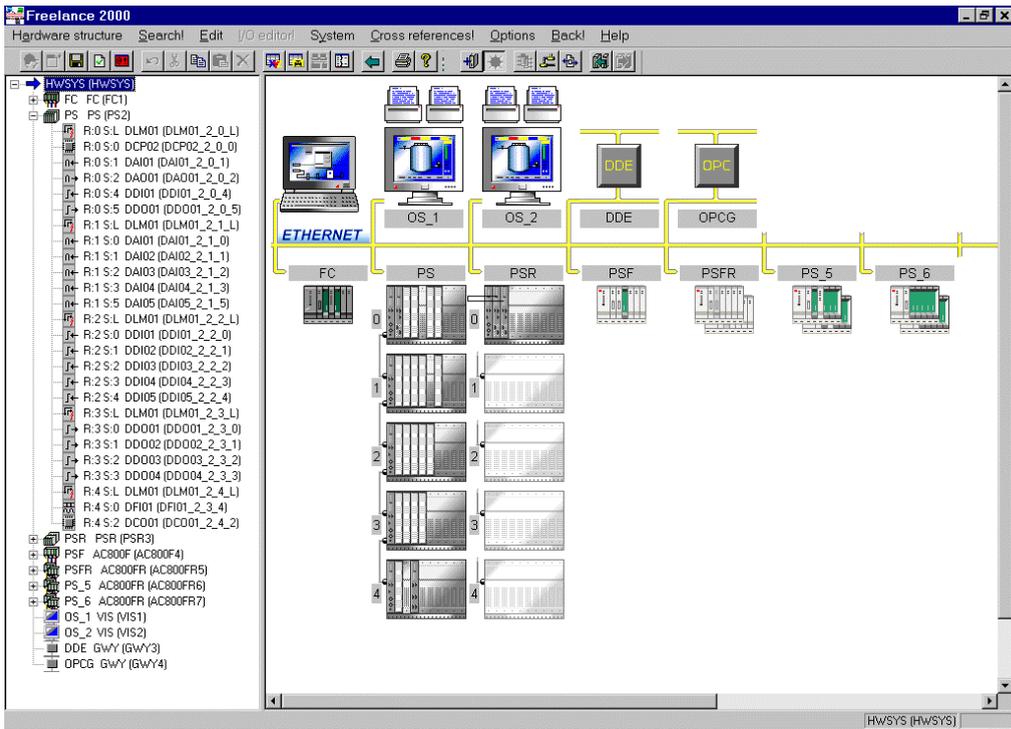
### E 2.3.1 System view

The system view shows the graphic view of the hardware structure. A system consists of any number of process and operator stations. The process stations consist of a central unit and up to four I/O units. For the FieldController, only the basic unit without the field busses or connected I/O units is displayed in the system view.

Up to two printers may be connected to each operator station. The engineering station need not be separately activated, and is only used for displaying the network address. Definite positions (1-10) are allocated to relevant stations either directly by selection (double click) of a definite position in the graphic view or by indication by means of a dialog during insertion in the tree view.



→ System → Hardware structure → Select system object (HWSYS)



### E 2.3.2 Station view of the process station as Rack system

In the station view modules can be equipped by simple double click; doing this, only those modules are offered in the relevant selection list which actually can be equipped. The station view of the process station shows the central unit with the CPU module and the I/O units. If other I/O units are activated, the station bus DigiNet P is shown connected between the stations. For each rack, a container name and a rack bay name can be allocated which can then be used as location indicator in the project documentation.



→ Select a station object (PS) in the tree view or system view



→ Select a DigiNet P master (DNETP) in the tree view or module view for the CAN module (FI 810F)

Cabinet name	Rack bay name
J4	A
J4	C
J4	E
J5	A
J5	C

di0931us.bmp

### E 2.3.3 Station view of the process station with FieldController

In the station view, the modules can be equipped with a simple double click. Only those modules which can actually be equipped are offered in the corresponding selection list.

The station view of the FieldController displays the basic unit with integrated CPU, the slot P for the power supply module, slots E1 and E2 for the Ethernet modules and slots F1 – F4 for the bus modules.



→ Select a FieldController object (ex. AC 800F) in the tree view or system view

The screenshot shows the Freelance 2000 software interface. The title bar reads "Freelance 2000" and the menu bar includes "Hardware structure", "Search", "Edit", "I/O editor", "System", "Cross references", "Options", "Back", and "Help".

The left-hand tree view shows a hierarchy of hardware objects under "HW/SYS (HW/SYS)":
 

- FC FC (FC1)
- PS PS (PS2)
- PSR PSR (PSR3)
- PSF AC800F (AC800F4) - Selected
- S:P SA801F (SA801F\_4\_P)
- S:E1 EI801F (EI801F\_4\_E1)
- S:E2 AM801F (AM801F\_4\_E2)
- S:F2 FI830F (FI830F\_4\_F2)
- S:F3 FI830F (FI830F\_4\_F1)
- S:F4 FI830F (FI830F\_4\_F4)
- PSFR AC800FR (AC800FR5)
- PS\_5 AC800FR (AC800FR6)
- PS\_6 AC800FR (AC800FR7)
- OS\_1 VIS (VIS1)
- OS\_2 VIS (VIS2)
- DDE GWY (GWY3)
- OPCG GWY (GWY4)

The main station view displays a rack with slots P, E1, E2, F1, F2, F3, and F4. Slot P contains a power supply module (SA801). Slots E1 and E2 contain Ethernet modules (EI801 and AM801). Slot F1 contains a bus module (FI830), which is highlighted in green. Slots F2, F3, and F4 are empty. The rack is labeled "115VAC/230VAC" and "10Base2".

The right-hand panel shows configuration options:
 

- Cabinet name: J6
- Rack bay name: A

The status bar at the bottom right indicates "PSF AC800F (AC800F4)".

di0932us.bmp

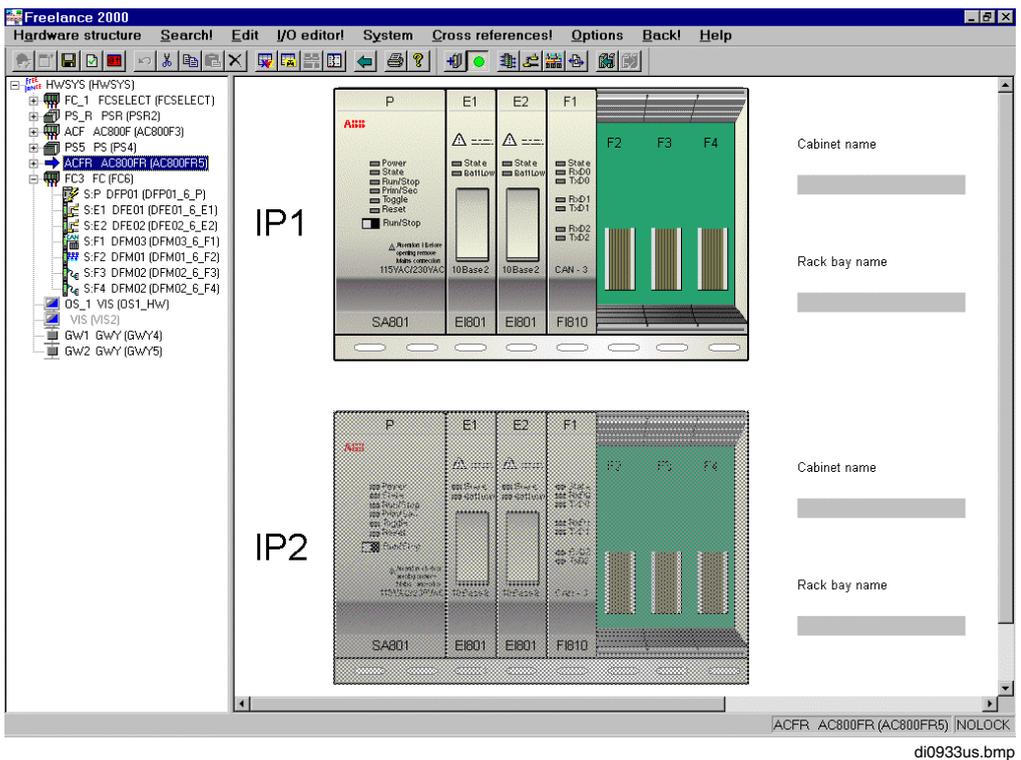
### E 2.3.4 Station display of a process station with redundant FieldController

In the station display of the redundant FieldController AC 800F, both modules can be assembled with a simple double-click. Therefore, the assembly of the modules is possible only in the upper base unit. In the selection list of the redundant AC 800F, only the redundancy objects are shown, designated by an R at the end, e.g. FI 830FR. The modules for the redundant AC 800F are set up in the same way as those for the non-redundant AC 800F. However, the objects in the hardware structure are made for redundancy, thereby enabling the configuration of both modules at the same time.

The station view of the Industrial<sup>IT</sup> Controller AC 800F shows two basic units, where only the upper one can be assembled. Otherwise, the display is identical with the non-redundant FieldController AC 800F, with the slots P for the power supply module, E1 and E2 for the Ethernet modules, and F1 to F4 for the field bus modules.



→ Select an AC 800FR from the tree view or system view.



## E 2.4 Detail view of the Profibus master

In the Profibus view, the slaves can be equipped with a simple double click. Firstly, the required view position and bus address must be defined. In the corresponding selection list, only the free positions and bus addresses are offered, i.e. only those which can actually be equipped. Then a dialog is started in which a slave object is generated by selecting a device database file. The Profibus view of the FieldController displays the master and the configured slaves on a stylized bus and the slave objects in the form of 70x40 pixel bitmaps which are either included in the device database file or can be assigned to the object later. If no bitmap has been assigned, the standard display for a slave is displayed.



→ Select a Profibus master (PROFI\_M\_DEV) in the tree view or module view for the Profibus module

Vendor:	ABB
Model name:	Freelance 2000 Profibus DPV1 Master
Tag name:	PROF_1_F4
Bus address:	1
Vendor:	FESTO AG&Co.
Model name:	FESTO Typ 02 FB9
Tag name:	SL_1F3_4
Bus address:	3
Vendor:	WIELAND
Model name:	ricos BC DP
Tag name:	Ricos
Bus address:	6
Vendor:	ABB Automation Products AB
Model name:	C1830
Tag name:	ABB_S800
Bus address:	14
Vendor:	R. STAHL IS1
Model name:	IS1 DP
Tag name:	Stahl
Bus address:	4
Vendor:	ABB-DRIVES
Model name:	ABB-ACS400-NPBA-02
Tag name:	ACS400
Bus address:	2

te078us.bmp

The most important information for the device such as vendor, name, tag name and bus address is displayed directly in the graphic view. Further information can be found in the parameter dialog of the object.

### E 2.4.1 Detail view of a device

The detail view displays detailed information on the device. It is a realistic view which is represented by a bitmap which can be assigned to the slave object. A list of equipped modules appears below the device view if the slave is modular.



→ Select a station object in the tree view or master view

Vendor: ABB Automation Products  
 Model name: S900  
 Tag name: SL\_4F2\_1  
 Bus address: 2

Position	Module name	Module type
0	M4F2_1_0	RII0_GTW ohne Nutz...
1	M4F2_1_1	AJ 4H Ex ohne HART ...
2	M4F2_1_2	DIO 8 Ex ohne Status ...
3	M4F2_1_3	AO 4 Ex
4	M4F2_1_4	AO 4 Ex
5	M4F2_1_5	AO 4 Ex
6	M4F2_1_6	AO 4 Ex
7	M4F2_1_7	AO 4 Ex
8	M4F2_1_8	AO 4 Ex
9		
10		
11		

Node: 3 BusAddr.: 5 PROFIL\_S\_DEV (AF800Series)

te079us.bmp

The most important information for the device such as vendor, name, tag name and bus address is displayed directly in the graphic view. Further information can be found in the parameter dialog of the object.

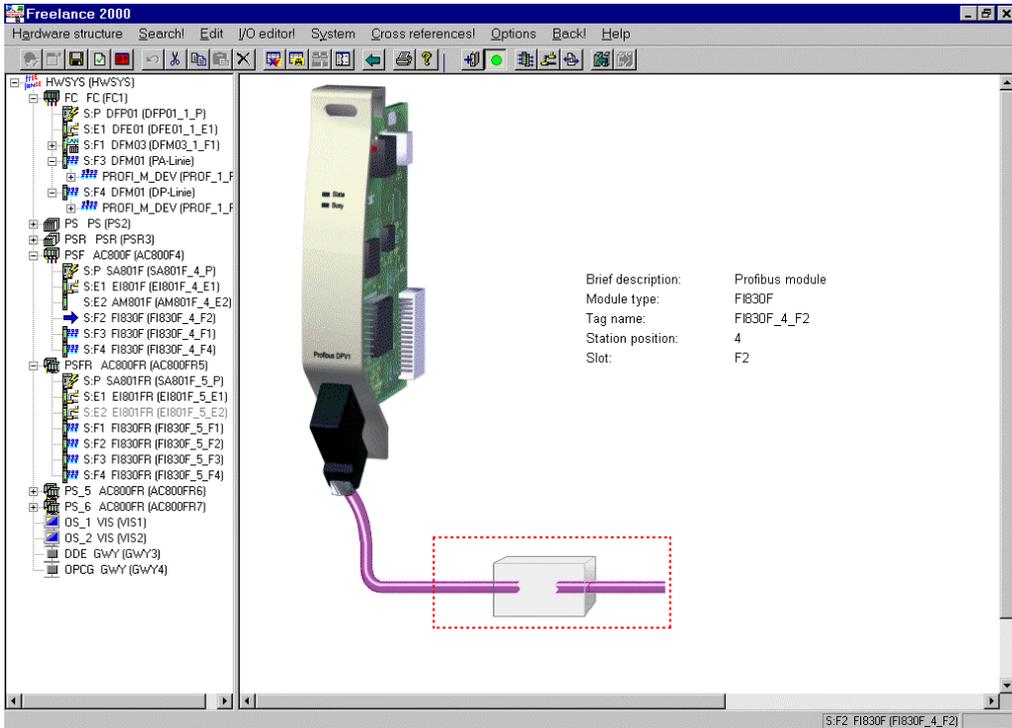
For the modules, the slot position, module name and module type are tabulated.

### E 2.4.2 Detail view of a module

The detail view displays a stylized display and detailed information on a module or the FieldController module.



→Select a module in the tree view or station view



te072us.bmp

## E 2.5 Menu structure of the hardware structure

<b>Hardware structure</b>	<ul style="list-style-type: none"> <li>Save</li> <li>Documentation</li> <li>Check all</li> <li>Check</li> <li>Network</li> <li>Connection configuration</li> <li>Exit</li> </ul>	
<b>Search</b>		
<b>Edit</b>	<ul style="list-style-type: none"> <li>Undo</li> <li>Cut</li> <li>Copy</li> <li>Paste</li> <li>Delete</li> <li>Clipboard viewer</li> <li>Templates</li> <li>Insert</li> <li>Resource allocation</li> <li>Parameter</li> <li>Dialog editor</li> <li>Block export</li> <li>Block import</li> <li>Activate/Deactivate</li> <li>Object</li> </ul>	<ul style="list-style-type: none"> <li>Change bitmaps (only context menu)</li> </ul>
<b>I/O editor</b>		
<b>System</b>	<ul style="list-style-type: none"> <li>Variable list</li> <li>Tag list</li> </ul>	
<b>Options</b>	<ul style="list-style-type: none"> <li>Hardcopy</li> </ul>	
<b>Return!</b>		
<b>Cross references</b>		
<b>Back</b>		
<b>Help</b>	<ul style="list-style-type: none"> <li>Contents</li> <li>Overview</li> <li>Use help</li> </ul>	

## E 3 General Functions of the Hardware Structure

### E 3.1 Context menu

The context menu provides a context-specific selection of currently used functions.



→Select an object in the tree view or graphic view.



te070us.bmp

### E 3.2 Cut, Copy and Paste

#### E 3.2.1 Cut



→Select station or I/O module to be cut

→ *Edit* → *Cut*

The selected objects are cut and deposited in the Windows clipboard. From there they can be repositioned with *paste*.



All parameters and I/O channel reservations of the I/O module are moved with it.

### E 3.2.2 Copy



→ Select station or I/O module to be copied

→ *Edit* → *Copy*

The selected objects are deposited in the Windows clipboard. From there they can be positioned as often as required with *paste*.



All parameters except the module name and all I/O channel reservations of these I/O modules are copied with them.

### E 3.2.3 Insert

New objects can be inserted at the selected position depending on the context. See **page E-25, Insert new objects.**



→ Select position

→ *Edit* → *Insert*

### E 3.2.4 Paste

The objects previously selected with cut or copy can be re-inserted when a destination has been specified.



→ Select position where the content of the clipboard are to be pasted

→ *Edit* → *Paste*

### E 3.2.5 Delete



→ Select station or I/O module → *Edit* → *Delete*

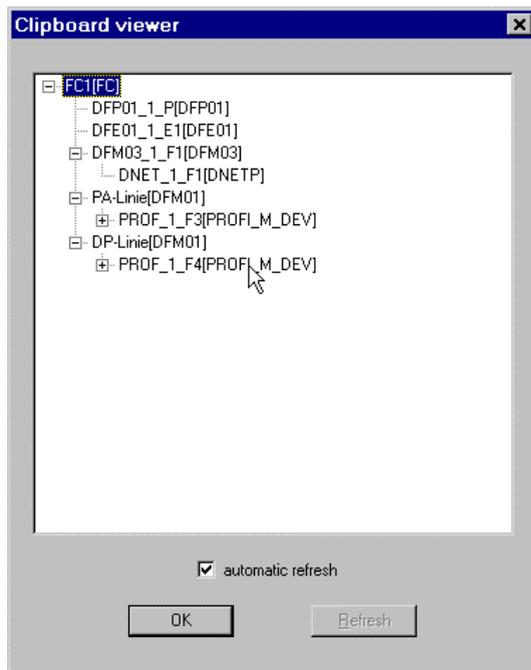
The parts selected are deleted from the process station. The module function block remains in the tag list and can be reallocated to a module of the same type with F2.

### E 3.3 Clipboard viewer

The dialog displays the objects on the clipboard. In each case a node can be selected and inserted at the desired position. The object(s) remain(s) on the clipboard and can be copied again.



→ *Edit* → Clipboard viewer → Select object → Drag to the destination position holding the left mouse button down



te104us.bmp

### E 3.4 Export block

To reuse parts of the configuration of the hardware structure at a different location, a block of objects or an object with all subordinate objects can be exported.



→ Select block/object → *Edit* → Export block  
→ State the file name of the export file (\*.hwm).

### E 3.5 Import block

After the import, a window is displayed with the objects which are on the clipboard. In each case a node can be selected and inserted at the desired position. The object(s) remain(s) on the clipboard and can be copied again.



→ *Edit* → Import block

→ Select object → Drag to the destination position holding the left mouse button down

### E 3.6 Check all

All objects below the selected object are checked for plausibility and for errors with all activated stations, units, modules and parameter entries, missing entries and inconsistencies. Errors are listed in a window and can be recovered later.

### E 3.7 Check



→ *Hardware structure* → *Check*

Only not plausible objects below the selected object are checked for plausibility.



A plausibility check is also offered in parameter windows and detail views; it then refers in each case to the system stations or modules.

### E 3.8 Search



→ *Hardware structure* → *Search*



An \* at the beginning and an \* at the end is internally added onto the text which is to be searched. Wildcards cannot be used.

### E 3.9 Cross references

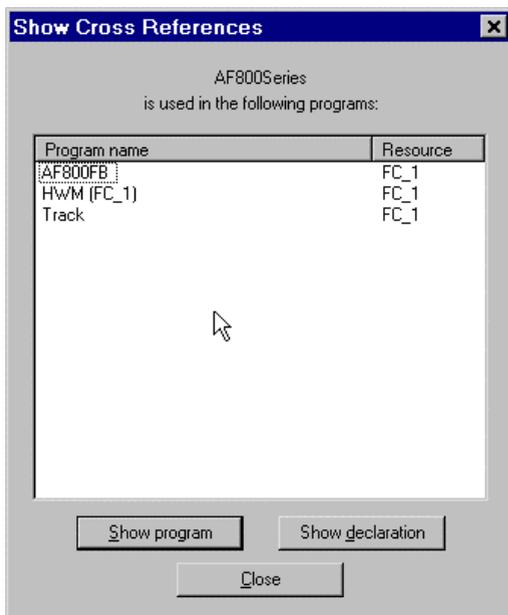
Cross references are references relating to an object in programs, displays, lists, etc., i.e. locations where I/O components of this object are used.

All cross references of an object can be displayed in a list by means of 'Cross references'.



→ Select object → *Cross References*

→ A window displays the names of the relevant programs in which the I/O components are used. Programs can be selected from the list with the arrow keys or via cursor click.



te050us.bmp

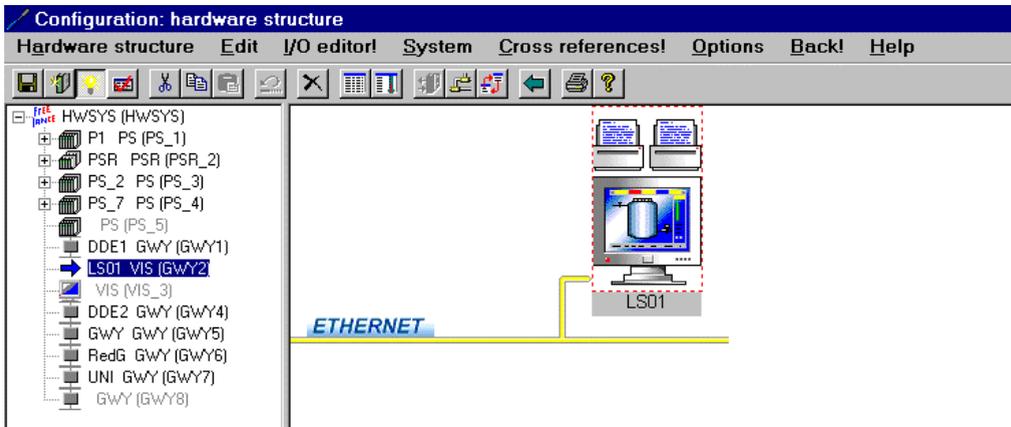
Filter	Displays only those I/O components or variables which are edited in <b>read</b> (sink) or <b>write</b> (source) mode.
SHOW PROGRAM	Call the program which uses an I/O component or a variable assigned to the I/O component.
SHOW DECLARATION	Step change to the corresponding I/O module in the tag list.
CLOSE	Window is closed.



-  Selection of the INSERT function is possible via menu, context menu or toolbar button.
-  Per FieldController, it is possible to connect only one CAN line with maximum 5 racks. The slot of the CAN module is preset to F1 and cannot be changed. If a CAN/3 module is installed in that slot, then the I/O bus (CAN) must be activated here. If any other module is installed in slot F1, then the CAN bus must be deactivated.

### E 4.1.1 Insert PC components in the operator station

-  → Select a position in the graphic view or the operator station node (VIS) in the tree view
- select parameter mask 2/3 or 3/3
-  → Double click with left mouse button on operator station position
- select parameter mask 2/3 or 3/3



di0952uk.bmp

### E 4.1.2 Insert gateway stations

Generally, gateways are always viewed at the control level next to the operator stations even if they run on a CPU module mounted to a process station (DCP-Gateway).

With 6.1 version IP address, resource ID and the boot parameter are configured in the hardware structure for **DDE gateways**, **OPC gateways** and **UNI gateways**.

Within the boot parameters is defined how many objects, how much memory and how many communications connections must be made available to other stations.

For **DCP gateways** no boot parameters are configured, since DCP gateways run on a CPU module. Therefore a module, type DCP 02GWY or DCP 10GWY, is mounted at the corresponding location. For a redundant DCP gateway two DCP 10GWY modules must be used.

At control level, the name of this module can be entered at the corresponding DCP gateway. Entering of the module name serves exclusively for slot documentation and is not needed for the gateway function.



→ Select system object (HWSYS) in the tree view → *Insert*

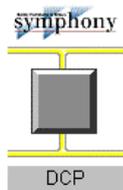


→ Double click with left mouse button on a operator station position.

The gateway type is defined with resource allocation. See also **page E-32, Resource allocation**.

The following gateway types are available:

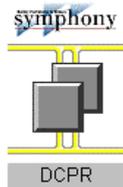
DCP gateway



te060.bmp

A DCP gateway is used for coupling the Maestro UX control system. It runs on a CPU module DCP 02GWY or DCP 10GWY installed in one of the process stations.

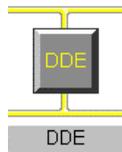
Redundant DCP gateway



te061.bmp

A redundant DCP gateway is used for coupling the Maestro UX control system. It runs on two CPU modules DCP 10GWY installed in one of the process stations.

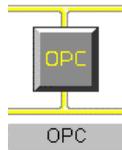
DDE gateway



te062.bmp

A DDE gateway runs on a PC, on which DigiDDE is installed.

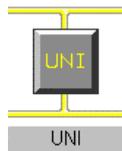
OPC gateway



te063.bmp

The OPC gateway runs on a PC, on which an OPC-Server is installed.

UNI gateway



te064.bmp

The gateway type UNI gateway is intended for DMSAPI use.

### E 4.1.3 Insert process stations

Process stations can be inserted in the system in either the tree view or the system view. After INSERT, the object appears in both the tree view and the system view. The object is shown at the corresponding station position. According to the selection the following is visualized:

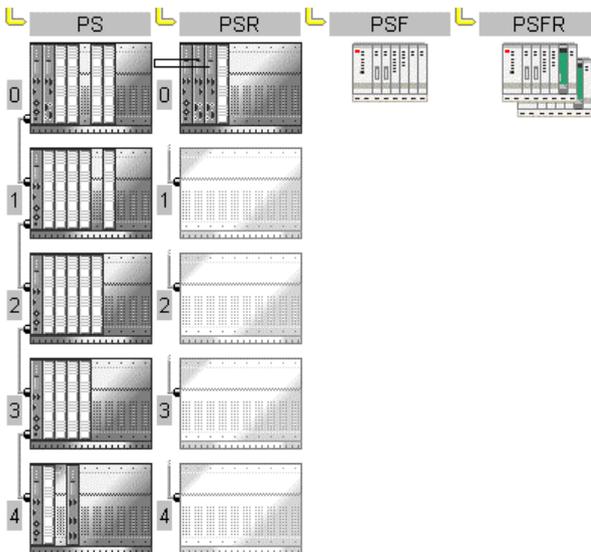
- Rack-based process station PS
- Rack-based process station redundant PSR
- FieldController FC
- FieldController *Select*
- Industrial<sup>IT</sup> Controller AC 800F
- Industrial<sup>IT</sup> Controller AC 800F Redundant



→ Select system object (HWSYS) in the tree view → *Edit* → *Insert*.  
 → Select station type → Select an insertion position



→ Select station position in the system view → Double click



te080us.bmp

#### E 4.1.4 Default names of the objects

With inserting a new object this one gets a default name which represents the installation point. With copying a object the name will adapt to the new position. With moving the name will consist.

#### E 4.1.5 Default names of the modules

S:F1 FI810F (FI810F\_4\_F1)

S:	Slot
FI810F	Module type, CAN-3 module here
((FI810F_4_F1)	Default name with station position 4, slot F1

#### E 4.1.6 Default names of the master objects

PROFI\_M\_DEV (PROF\_4\_F1)

PROFI_M_DEV	Profibus DPV1 master object (device)
(PROF_4_F1)	Default name with station position 4, slot F1

#### E 4.1.7 Default names of the slave objects

Note: 0: BusAddr.2 PROFI\_S\_DEV (SL\_4F1\_2)

Note:0	Node 0, the same as the bus address as standard
BusAddr.2	Bus address 2

 If the bus address is changed later, the number of the node remains the same. Therefore, it is possible to sort the Profibus devices in the tree and graphic view in a different way than by bus address. The number of the node can be changed in that way that the slave object will move onto the master object.

PROFI_S_DEV	Profibus slave object (device)
(SL_4F1_2)	Default name with station position 4, slot F1 and bus address 2

#### E 4.1.8 Default names of the slave modules

ModAddr0: MODUL (M4F1\_2\_0)

ModAddr0:	Module address (= module position) 0
MODUL	Profibus slave module
(M4F1_2_0)	Default name with station position 4, slot F1, bus address 2 and module position 0

**E 4.1.9 Standard names of the Hart Channels**

Ch 1: HART (HART\_1\_001)

Ch 1:: Channel address 1  
 HART Hart channel object  
 (HART\_1\_001) Consecutive numbering from 001 to 999, valid for the whole project.

**E 4.1.10 Default names of the modules**

R:1 S:L DLM 01 (DLM01\_4\_0\_L)

R: Rack  
 S: Slot  
 DLM 01 Module type, e.g. link module  
 (DLM01\_4\_0\_L) Default name with station position 4, rack 0, slot L

**E 4.1.11 Default name of the serial interface**

Below the CPU modules (DCP 02, DCP 10), the communication module (DCO 01) or the serial module DFM 02, the serial interface assignments are shown as follows:

Ser1: MODM\_DEV (MODM\_2\_0\_1\_2)

Ser1: Serial interface 2

 CPU modules only feature one serial interface; four interfaces max. are available for DCO 01.

MODM\_DEV Modbus Master Device (unit)  
 (MODM\_2\_0\_1\_2) Default name with station position 2, rack 0, slot 1, serial interface 2

**E 4.1.12 Default name for the remote process controller**

BusAdr 3: FLRC\_MSL (FLRC\_2\_0\_254)

BusAdr 3: Bus address 1-254  
 FLRC\_MSL Freelance Remote Control as Modbus Slave  
 FLRC\_MSL\_001 Default name with class name for Freelance Remote Control as Modbus Slave, consecutive numbering from 001 up to 999, valid for the whole project.

## E 4.2 Resource allocation

D-PS resources, D-GS resources and/or D-LS resources configured in the project tree must be associated to the respective hardware stations.

In this way the software parts of the user program configured in the project tree are allocated to the hardware stations.



- Left click on the gray text area of the station
- Edit* → *Resource allocation*
- Select the corresponding resource from the list



- Double click on the gray text area of the station
- Select the corresponding resource from the list

Allocation of a resource to a station automatically activates the station; i.e. in DigiTool commissioning mode the station is searched for in the system bus.

### Delete the resource allocation

Deleting the resource allocation deactivates the stations.



- Left click on the gray text area of the station
- Edit* → *Resource allocation*
- Select "No allocation to resource"



- Double click on the gray text area of the station
- Select "No allocation to resource"



- Data and parameters previously entered by the user are not lost by deleting a resource allocation. When allocation is continued they reappear and can be re-edited.

### E 4.3 Activate/deactivate objects



A deactivated object is not loaded into the process station or the FieldController or one of its subordinate devices. If the deactivation of an entire station is saved in the hardware structure, the entire station must be reloaded, i.e. initialized, after reactivation.

A deactivated object appears in gray in the tree view. In the graphic view, the object is displayed in gray or in a lighter shade within or below the FieldController.

In the graphic view of a process station, the CAN cable connection between the racks is represented open and in the activated state closed. The frame of the CPU or I/O unit and/or a deactivated module are also shown in gray.

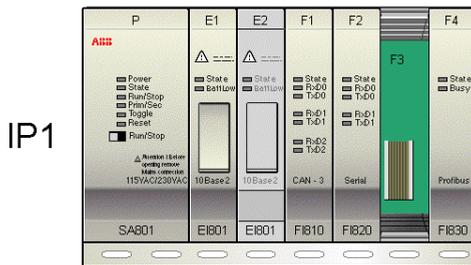


→Left click on the object → *Edit* → *Activate/Deactivate*

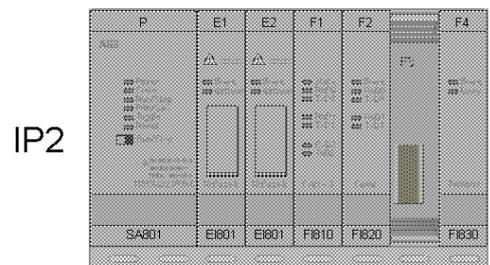
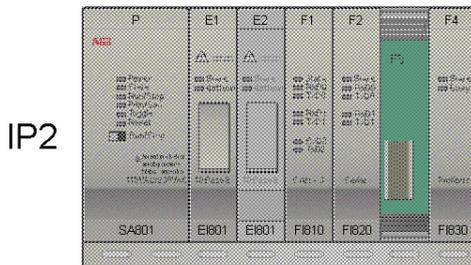
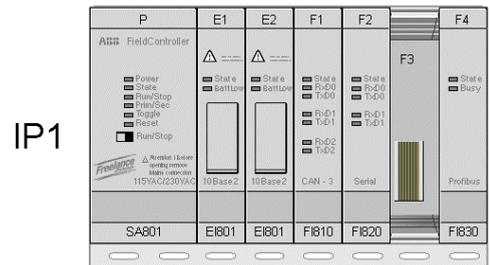


→Right click on the object → *Activate/Deactivate*

activated:



deactivated:



te107us.bmp

te108us.bmp

## E 4.4 Configuration of the communications connections

For data exchange with operator and gateway stations, 10 connections are available in each process station. If more stations are configured, the operative communications connections of all users of a Freelance system can be defined in a separate parameter dialogue.

The dialogue is called up with a double click on the project element configuration KONF or by selection from the menu.



→ Hardware structure → Connection configuration

Communication configuration

Default values for new process stations

Max. number of connections to operator stations: 2

Max. number of connections to gateway stations: 2

	OS_2	OS_1	DDE	OPGC
FC (4/4)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
PSF (4/4)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
PS (4/4)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
PSR (4/4)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
PSFR (4/4)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
PS_5 (4/4)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
PS_6 (4/4)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

OK Cancel

td042us.bmp

### Standard setting for new process stations

*Max. no. of connections to operator stations*

*/ gateway stations*

If a new process station resource is created in the project tree, the number of connections is set up with these values. Each process station can communicate with up to 10 operator and/or gateway stations. That is, a value between 1 and 10 can be entered in each field, but the sum of the two values cannot exceed 10.

*Matrix field*

In the left-hand column, all process station resources, in the upper line, all operator and gateway station resources, are listed with the names which were configured in the project tree. Each field of this matrix represents a connection between a process station and a operator or gateway station.

The connection In Online Mode if a connection between the processing station and the control station is to be created.

Up to 10 connections can be configured in each line of the matrix.

See also **chapter project tree**.

## E 5 I/O Editor

Each module provides I/O components with respect to the number of channels. The channel of an I/O module can be used directly in programs and graphic displays via the I/O component. If the I/O component shall also appear in the variable list, it is possible to also indicate a variable name.

-  I/O components without a variable association do not show up in the variable list and therefore can not be exported for lateral communication and can not have an initial value.
-  If the access to an I/O component is programmed by the <TagName>.<ComponentName> syntax, deleting the referenced I/O device sets all programs to incorrect. To get a correct program it is necessary to create a new I/O device instance with the name <TagName> which supports the referenced component <ComponentName>. If the access to an I/O component is programmed by a variable, deleting the corresponding I/O device sets all programs to incorrect. To get a correct program the programs must be checked again.
-  Both I/O components and variables can be called in programs and displays via the function key F2.

The I/O component name is composed of the module name (16 characters) and the component name (16 characters), so that 'module name.component name' provide a total of 32 characters. The 'Module name' is the tag name of the module, the 'Component name' is the name of the I/O or diagnosis component.



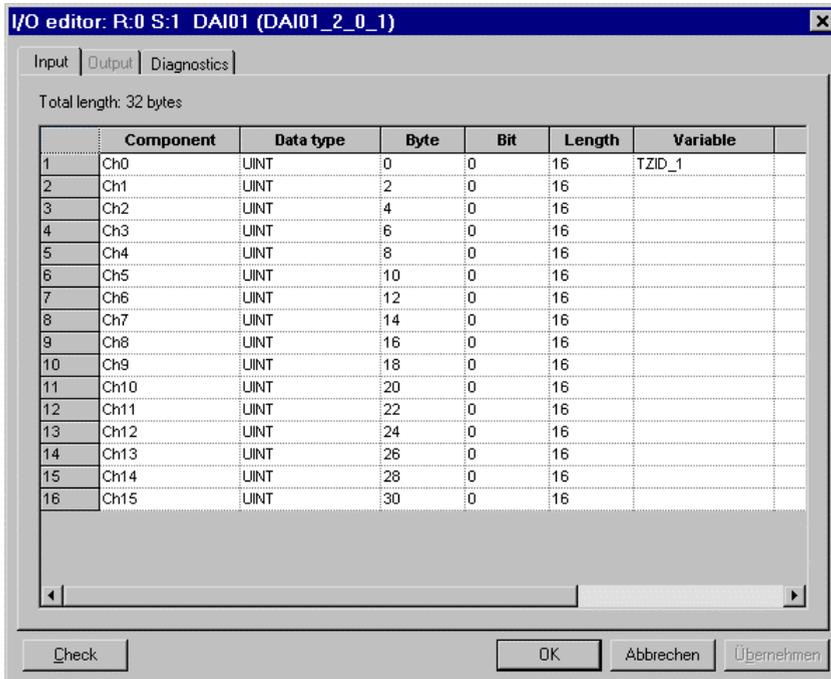
Variable names cannot have the same name as tag identifications. Such a name allocation is stopped by the check for plausibility. For existing projects conflict situations are automatically eliminated during project import by attachment of 'var' to the variable name. So the respective variables can be subsequently modified.

## E 5.1 Call the I/O Editor



→ Left click on the module in the tree view

→ I/O Editor



te048us.bmp

The I/O editor is composed of several register cards indicating inputs, outputs and diagnosis components. The columns on the tabs can be sorted via mouse click on the column head.

### Component

This column shows the names of the components. Depending on the type of the module, the module channels are marked Ch0 to Ch7/15/31 by default setting. It is possible to change the component name (16 characters max.) but within a module the name must be unambiguous.



By renaming a component, all programs referring to this component become incorrect.

- Data type* The variables for analog modules are processed as variables of data type UINT from the analog I/O module, except for the temperature module DAI 04 (REAL). The useful signal is 12 bits wide. These variables can be converted into a REAL format for further processing in the programs, e.g. in order to switch them to analog monitoring. This is done with converter modules, in which the ranges are configured for further processing or given default values, e.g. in case of wire break. These converter modules transpose the process signal (e.g. 20 mA) to a physical value, or conversely the physical value to an output signal.  
An exception is the DAI 04 module, which carries out this conversion on the module by linearizing the signals over a corresponding characteristic. The signals of the digital modules are supplied to the system with variables of data type BOOL. The variables do not require a converter, as their states can be processed directly.
-  By changing the data type, all programs referring to this component become incorrect.
- Byte* Zero-based byte offset of the component in the actual section. The value of the component starts at this offset and ends at offset plus the size of the component data type. Only byte values equal to or larger than 0 are allowed.
-  Changing the byte offset will not adjust the byte offset of other components. If the byte value is changed all references to components of this I/O module or device in user programs or wherever variables can be referenced will be set to incorrect.
- Bit* Zero-based bit offset of the component in relation to the corresponding byte. Only bit values in the range 0...7 are allowed.
-  Changing the bit offset will not adjust the bit offset of other components. If the bit value is changed all references to components of this I/O module or device in user programs or wherever variables can be referenced have to be set to incorrect.
- Bit Length* Bit length of the corresponding data. This parameter must be specified.

Variable	<p>Different name for the I/O component. Enables access to I/O components not only by the new syntax described above but also in the same manner as in earlier DigiTool versions. This is an optional parameter. If this field is left empty, the I/O component can only be accessed by the new syntax. The variable name must be unique for all variables. Pressing F2 shows a list which contains all 'unmapped' variables of the project which are located on the same process station and which have the correct data type. All variables show up in the variable list with the same behavior as any other variable. Variables which have not yet been configured in the project, can be declared directly in the I/O editor.</p>
	<p> Only I/O components with allocated variables can perform lateral communication, i.e. can be read in other resources.</p>
	<p> If the variable name is changed all references to the old and to the new variable will be set to incorrect.</p>
Comment	<p>Comment for the I/O component. This is an optional parameter. The maximum length of this entry is 31 characters. Changing the comment for references to an I/O component has no effect on the check state.</p>
CHECK	<p>By pressing the 'Check' button the I/O module checks the I/O definition (overlapping bytes, gaps, data types, etc.).</p>

## E 5.2 Edit components



→ Left click on the object → Right click → Context menu

I/O editor: Node 20: PROFIBUS\_SL (ABB\_S800)

Input | Output | Diagnostics

Total length: 14 bytes

	Component	Data type	Byte	Bit	Length	Variable
1	FCL_RPA	BOOL	0	1	1	6
2	FCL_RPB	BOOL	0	2	1	6
3	FCL_CDS	BOOL	0	3	1	6
4	FCL_GE	BOOL	0	6	1	6
5	FCL_GW	BOOL	0	7	1	6
6	FCL_ACT	BOOL	1	0	1	6
7	FCL_SE	BOOL	1	1	1	6
8	<b>Insert component</b>	<b>Ins</b>		2	1	6
9	<b>Delete component</b>	<b>Del</b>		3	1	6
10				4	1	6
11	<b>Cut</b>	<b>Ctrl+X</b>		5	1	6
12	<b>Copy</b>	<b>Ctrl+C</b>		0	4	6
13	<b>Paste</b>	<b>Ctrl+V</b>		4	1	6
14				5	1	6
15				7	1	6
16	<b>Move position</b>			0	16	6
17	<b>Undo</b>	<b>Ctrl+Z</b>		0	8	6
18				0	8	6
19	M1_IN_Not_Used1	WORD	8	0	16	6

Check OK Abbrechen Übernehmen

te101usbmp

### E 5.2.1 Insert component

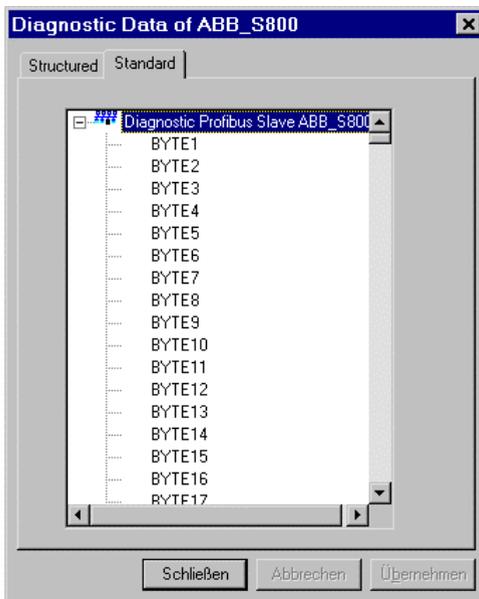


→ Right click on a field or line number → Context menu → *Insert component*

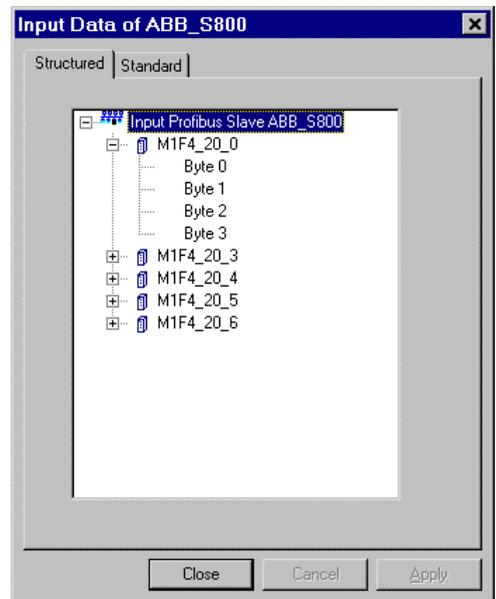
A new component is inserted. All lines below are moved down. “New0000” is defined as the first component name and then “New0001”, etc.

 If a component for the modular slave is inserted, a special dialog appears which makes it possible to access individual items of information on the field device. A distinction is made between structured access and standard access. With structured access, the structures of the device data are defined by the user or device vendor. With standard access, the device data is listed under the slave object unstructured. Both views display all device data which has not yet been assigned to a component.

 → Right click on a field or line number → Context menu → *Insert component* → Select file card STRUCTURED/STANDARD → Select component → Right click → Select data type



te105us.bmp



te106us.bmp

### E 5.2.2 Delete component

 → Right click on line number, possibly drag block → Context menu → *Delete component*

The selected components are deleted. All lines below are moved up.

 Components can only be deleted if the entire line has been marked beforehand.

### E 5.2.3 Copy

 → Right click on line number, possibly drag block → Context menu → *Copy*

The selected components are placed on the clipboard. The lines/components to be copied can be inserted, after selecting the destination position/line, using paste contents.

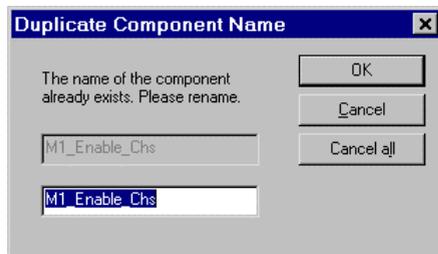
 Components can only be copied if the entire line has been marked beforehand.

### E 5.2.4 Paste contents

 → Right click on line number → Context menu → *Paste contents*

The components previously placed on the clipboard are inserted.

 The contents of the clipboard can only be inserted if the entire line has been marked. If it is discovered, during insertion, that the corresponding component already exists, a new component name can be entered.

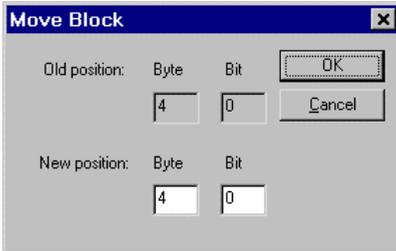


te103us.bmp

### E 5.2.5 Move position



→ Right click on line number → Context menu → *Move position*



te102us.bmp



A position of a component can only be moved if the entire line has been marked.

### E 5.2.6 Change data type



→ Right click on data type field

The data type for digital module channels can be set (BOOL, BYTE, WORD, DWORD). The data type for analog modules is permanently set to UINT. For field devices, the data type can be set freely.

## E 6 Network Configuration



→ Hardware structure → Network

Resource type	Resource name	Resource ID	IP address 1	IP address 2
D-ES		32	172.16.1.32	
D-GS	DDE	82	172.16.1.32	
D-OS	OS_1	21	172.16.1.21	
D-OS	OS_2	59	172.16.1.39	
D-PS	PS_1	10	172.16.1.10	
D-FC	FC_2	1	172.16.1.1	
D-FC	FC_1	2	172.16.1.2	

di0944uk.bmp

### E 6.1 IP address and resource ID

All stations are linked by the system bus DigiNet S. The system bus is based on the Ethernet standard with the TCP/IP communications protocol. The word Ethernet denotes local networks complying with DIN ISO standard 8802, part 3 and the now valid standard IEEE 802.3.

The Freelance 2000 system makes a difference between **Resources** and **Stations**. Stations are units connected to the Ethernet with a definite IP address.

Resources are software parts loaded to the stations. In the project tree they are displayed as resource nodes and identified in the network by a unique **resource ID**.

The differentiation between resource ID and IP address is necessary as there are stations in which several resources can be loaded (e.g. one PC with DigiVis and DigiDDE); but resources can also be addressed via two different IP addresses (e.g. a redundant process station).

In the example "DigiVis and DigiDDE on one PC", two different resource IDs must be allocated for installation on the PC under which the two software parts can be addressed. For a project configuration, the same IP address (the address of the PC in which the resource is loaded) is allocated in the hardware structure for both resources and the two different resource IDs are entered.

In the example "**Redundant Process Station**", two IP addresses are entered for the process station resource.

### IP addresses

The IP (Internet Protocol) addresses are unique worldwide and are allocated within the scope of the installation of DigiTool or DigiVis. For local networks, which are not linked over a router with the outside world, it is possible to allocate 172.16.x.y with  $x = 0.5$  and  $y = 0.255$ . If Freelance 2000 stations are in a network with other users, please have corresponding IP addresses allocated by your network representative.

 The IP addresses permanently set at present in Freelance 2000 (172.16.1.1) are exclusively allowed for local Freelance 2000 networks. If other network users should be connected, the IP addresses set in the CPU modules should be altered through the CPU module's diagnostic interface.

Various types of cable are available as transmission media. They need not be considered when configuring the hardware structure, as the IP addresses are independent of them.

See also manual **Installation and Maintenance Instructions, Setting the CPU Modules** for setting of IP addresses on the CPU modules.

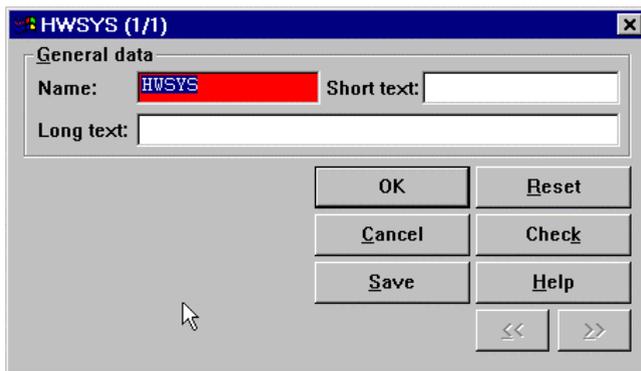
## E 7 General Parameters of the Hardware Structure Objects

All stations such as process stations, FieldControllers or operator stations and their subordinate objects such as modules are elements/objects in the hardware structure. These objects have an object name and a parameter dialog with specific object parameters.

### E 7.1 HWSYS system object

HWSYS System object is only a structure element comparable to the project element "Configuration" in the project tree. As the project element configuration stands for the total of all software resources, the system object stands for the total of all hardware stations such as process, operator or gateway stations.

With the system object the system view is selected in the graphic part of the hardware structure.



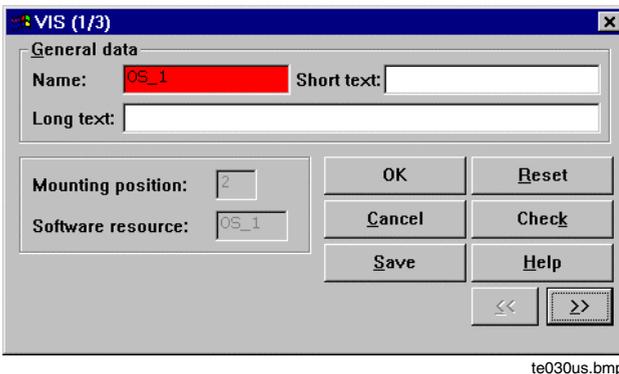
te028us.bmp

Please see **page E-5, General Description - Hardware Structure**.

## E 7.2 VIS operator station

The **operator stations** are usually PC's in which the DigiVis software program is installed for process visualization. All displays and records are configured with the software program DigiTool and loaded in the operator station. Data acquisition for recording and trend display is done in the process station; the complete display and archiving is done in the operator station.

Each process station can communicate with up to 10 operator and/or gateway stations. The display of the communications connections is done in the *Connection Configuration*. See also **chapter Project Tree Configuration (CONF)**. See also **page E-32, Resource allocation**.

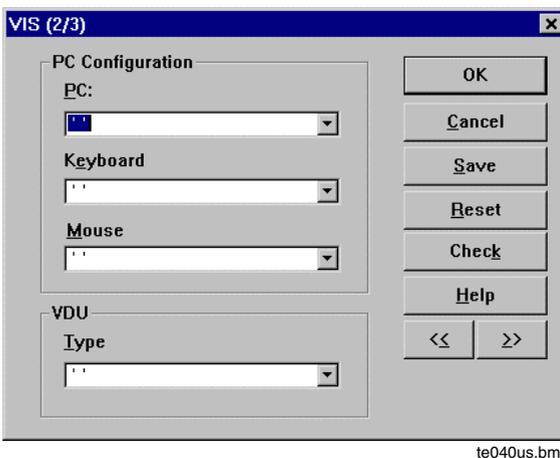


### General data

*Mounting position* Position in the graphic view of the System Display.

*Software resource* e.g. process station in the project tree

With >> turn to the next parameter definition mask.



**PC Configuration (only for documentation)**

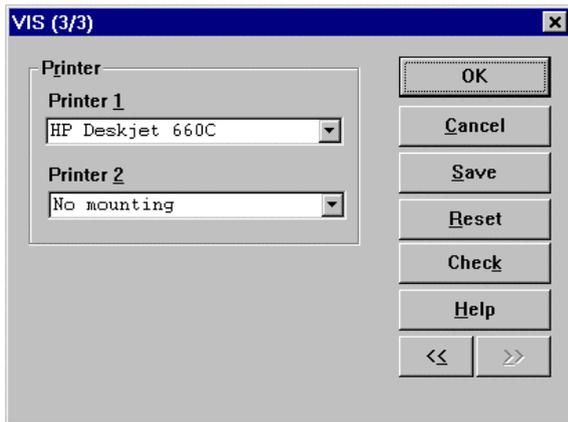
*PC:* PC for Freelance 2000 as OS

*Keyboard:* Type of Keyboard.

*Mouse:* Type of Mouse.

*VDU:* Type of Monitor.

With >> turn to the next parameter definition mask.



te041us.bmp

**Printer**

*Printer 1* Operator station printer

*Printer 2* Operator station printer

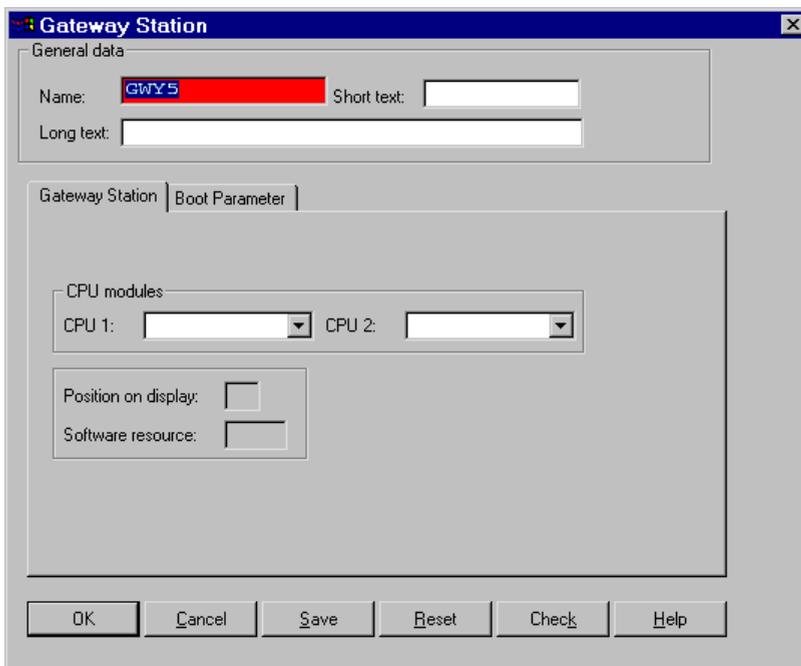
 Only for printer documentation, not for printer setting.

### E 7.3 GWY gateway-station

Gateways currently exist for the connection of **Maestro UX**, for any **DDE clients** such as Microsoft Excel or Visual Basic applications, for any OPC clients and for individually created applications (**DMS API**).

The max. number of gateway stations is defined in the project tree element Configuration. See also **chapter Configuration (CONF)**. The hardware configuration must be allocated to a resource (project tree), because the gateway type is only defined by means of such an allocation. For each gateway station configuration the number of configurable control stations is reduced by one.

See also **page E-32, Resource allocation**.



te059us.bmp

#### Gateway station tab.

##### CPU modules

- |                          |   |
|--------------------------|---|
| <i>CPU1</i>              | Name of the corresponding CPU module (DCP gateways only)          |
| <i>CPU2</i>              | Name of the corresponding CPU module (redundant DCP gateway only) |
| <i>Mounting position</i> | Position in the graphic view of the system display.               |
| <i>Software resource</i> | Allocated resource (project tree)                                 |

**Boot parameter tab****Memory**

These numbers influence the partitioning of the memory within a DCP gateway. If errors occur in loading of the objects when there is a very large number of objects in the gateway, you can try to make the project loadable by changing these settings.

*Max. number  
of objects*

Maximum number of objects which can be loaded on the gateway.

*Configuration data  
(PRAM)*

Storage area, in KB, which is reserved for the configuration data. This area is cold-start stable. (only with DCP gateway)

**System limits**

*Number of  
connections*

Maximum number of configured communications connections. Also see **page E-34, Configuration of the communications connections.**

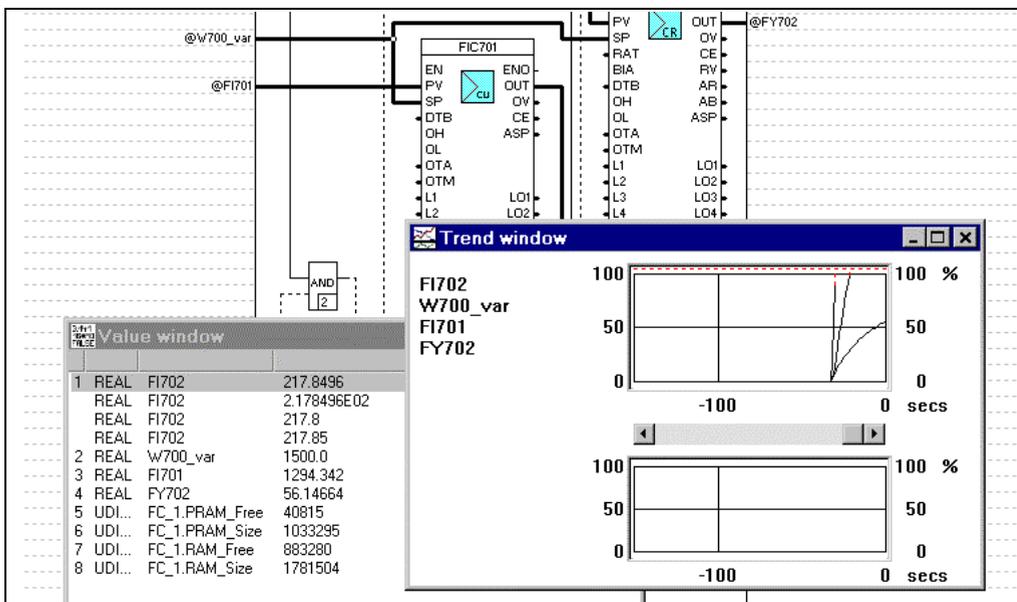
**E 7.4 Process station**

The process station is defined by the controller or the CPU. In essence, the process stations can be divided into two classes: the conventional process station where the I/O modules are assembled in module assembly frames and the process is carried out in one or two (redundant) CPU modules, and the FieldController where several field buses as well as rack-assembled modules can be connected.

The configuration of the process station is described in specific manuals.

The station types PS and PSR are described in the **Engineering manual, Process station, Rack system**. The station FC is described in the **Engineering manual, Process station, FieldController**. The stations AC 800F and AC 800FR are described in the **Engineering manual, Process station, ABB FieldController 800**.

# F Commissioning





## Contents

<b>F 1</b>	<b>General Description - Commissioning .....</b>	<b>F-5</b>
<b>F 2</b>	<b>Starting Commissioning .....</b>	<b>F-7</b>
F 2.1	Commissioning procedure.....	F-8
F 2.2	Switching between commissioning and configuration.....	F-9
F 2.2.1	Switching from commissioning to configuration .....	F-9
F 2.2.2	Switching from configuration to commissioning .....	F-10
F 2.3	Faster switch from commissioning to configuration mode .....	F-11
<b>F 3</b>	<b>Commissioning User Interface .....</b>	<b>F-12</b>
F 3.1	Project tree .....	F-12
F 3.2	Differences between commissioning and configuration modes .....	F-13
F 3.2.1	Configuration functions.....	F-13
F 3.2.2	Commissioning functions .....	F-14
F 3.3	State displays in the project tree .....	F-15
F 3.4	Representation of the side effects caused by configuration changes .....	F-15
F 3.5	Project version.....	F-16
F 3.5.1	Version check.....	F-16
F 3.5.2	Adjust version error .....	F-17
F 3.5.3	Show global variables of the resource .....	F-17
F 3.5.4	Show exported variables of the resource .....	F-18
F 3.5.5	Show all objects .....	F-19
F 3.5.6	Show selected objects.....	F-20
F 3.5.7	Program information.....	F-21
<b>F 4</b>	<b>Time Settings .....</b>	<b>F-22</b>
F 4.1	System time, local time and time zone.....	F-22
F 4.1.1	Set time zone .....	F-22
F 4.1.2	Synchronize system time .....	F-22
F 4.1.3	Set system time .....	F-23
F 4.1.4	Writing DT variables .....	F-24
F 4.2	Daylight saving time .....	F-24
<b>F 5</b>	<b>Loading the Project.....</b>	<b>F-25</b>
F 5.1	Load whole station .....	F-26
F 5.2	Load variables .....	F-27
F 5.3	Load message configuration .....	F-27
F 5.4	Load selected objects.....	F-28
F 5.5	Load changed objects .....	F-28
F 5.6	Uploading current block parameters .....	F-29
F 5.7	Delete project from the process station.....	F-32
F 5.8	Ignore version error .....	F-32

<b>F 6</b>	<b>Changes with Side Effects on Other Project Objects .....</b>	<b>F-33</b>
F 6.1	Display of changes in the project tree .....	F-33
F 6.2	Changes with no side effects.....	F-34
F 6.3	Changes with side effects on the task.....	F-34
F 6.4	Changes with side effects on the resource .....	F-34
<b>F 7</b>	<b>Displaying and Writing Values in a Window .....</b>	<b>F-35</b>
F 7.1	Enter variable .....	F-37
F 7.1.1	Define debug windows .....	F-38
F 7.1.2	Trend options.....	F-40
F 7.2	Show value window .....	F-41
F 7.3	Show trend window .....	F-42
F 7.4	Writing a value.....	F-43

## F 1 General Description - Commissioning

Commissioning is an operating mode of DigiTool which offers a range of other functions in addition to the loading of project objects. However, in contrast to the configuration operating mode, the user programs cannot be changed structurally. Only when the user programs have been configured and checked for plausibility, can they be loaded into the process station or the FieldController and started as part of the commissioning process.

When a project is commissioned for the first time, the complete project must be loaded into the respective station. Later, only changes need to be loaded. Changed project objects are loaded and started in accordance with selections made in the project tree. The selection of single or multiple user program objects is made in the project tree, as already described under configuration.

As well as **starting, loading, stopping** or **initializing** project objects, such as a resource, a task or a program list, it is also possible to edit individual function blocks of programs. This enables the operator to change operating modes, switch to a specified operating state and set parameters for function blocks without having to reload the modified program. There are two methods of making these changes known to the system:

- **Write** loads the changes without saving them in the project file.
- **Correct** saves the changes in the project file, as well as loading them.

Although they share a common user interface, configuration and commissioning are two separate processes. For commissioning, a DigiNet S (Ethernet) system bus **connection** must be established from the engineering station to the process station(s)/FieldController(s) and the operator station(s). It is then possible to switch directly between configuration and commissioning modes.

After loading a project with **Load whole station**, additional configuration changes may be loaded incrementally. A configuration change consists of the creation of, deletion of or a change to a project object. A configuration change which effects other project elements is said to have **side effects**.

During loading, the existence of side effects will cause the changed project object and any objects affected by it to be stopped. For this reason, changes with side effects should be loaded during operations only with the greatest of caution. The user is made aware of the occurrence of side effects by the way the object nodes concerned are displayed in the project tree.

Version checking ensures that the project running on the commissioning station and the process station are one and the same. As regards the loading of changes, configuration changes which have not been downloaded are indicated by arrows on the objects concerned.

Commissioning of the programs is facilitated in all cases by a global **value and trend window**. In it the user can track analog and binary values from one program or another.

In **FBD** and **LD programs**, binary values are displayed directly with their logical state of 1 or 0. The state of the binary signal is recognized by a different line type. In the **IL programs**, the current contents of the accumulator are displayed in their own column.

In **SFC programs**, the processing of the transitions and steps can be controlled. That language allows the transitions to be blocked or forced. The steps can be processed in permanence or permanently switched off. In addition, parameters such as the step wait time (TWA) and the step monitoring time (TMO) can be altered.

Binary and analog values can be set on a one-time basis in order to test the reaction or functionality of the program.

Values can be **forced** to remain constant (for debugging purposes) at the I/O modules and field devices. This is achieved by switching to the hardware structure and selecting the requisite object.

## F 2 Starting Commissioning

Unlike configuration, commissioning can only be performed once a connection to the process station has been established via the DigiNet S (Ethernet) system bus. Communication is set up automatically when *Commissioning* is selected.



Call project manager → *Commissioning!* or

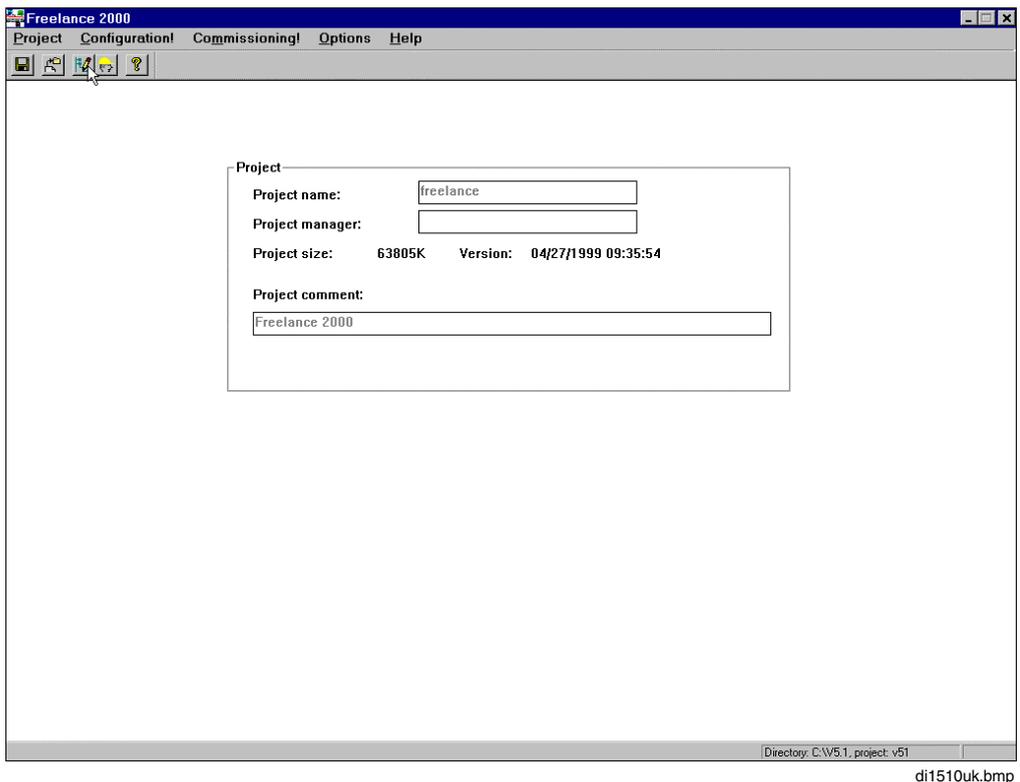


Call project tree → *Project* → *Commissioning!*



It is possible to switch between configuration and commissioning modes directly from the project tree.

Detailed information on installation of the DigiNet S system bus connection can be found in the **Mounting and Installation Instructions** manual.



**Connection Requirements**

- Ethernet card correctly installed in the engineering station DE-S
- TCP/IP driver loaded
- System bus cable connected correctly to the stations to be commissioned
- Station numbers and IP addresses entered correctly in the stations
- Station numbers (IP addresses) unique
- Following an upgrade/update, current operating system downloaded to process station
- Current EPROM's downloaded to modules

**F 2.1 Commissioning procedure**

- The project parts to be loaded into the process station must have satisfied plausibility checks.
- The start-up characteristics of the project must be known. Since it is possible to configure an autostart for tasks, autostarts should be rechecked before starting.
- Call up Commissioning
- Select a resource, then the project objects to be loaded from the project tree.
- Initiate processing in the function blocks, switch on the program list, the Start Task and start the resource.

## F 2.2 Switching between commissioning and configuration

Direct switching between commissioning and configuration modes is possible from the project tree.

### F 2.2.1 Switching from commissioning to configuration

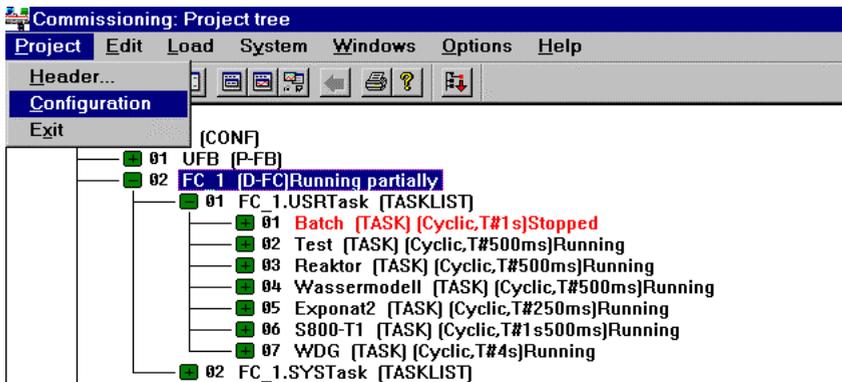


*Project → Configuration*

or



*Program → Configuration*



di1513uk.bmp

It is necessary to switch to configuration when, for example:

- Objects are to be added or changed
- Variable changes are to be performed, such as adding new variables or changing data types
- Extra tags or variables are to be configured in a resource

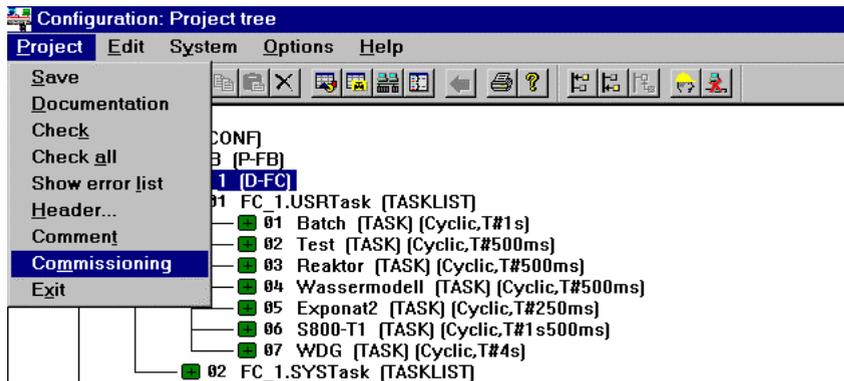
### F 2.2.2 Switching from configuration to commissioning



*Project → Commissioning*



After a switch from commissioning mode into configuration mode, it is not permissible to switch from the program directly back to commissioning. The reason is possible side effects which any changes made might have on other objects, such as programs or displays. Visible indication of side effects is provided only in the project tree display; hence a switch to the project tree is required before a return to commissioning.



di1514uk.bmp

It is necessary to switch to commissioning when, for example:

- The user program is to be loaded
- After plausibility checks, objects are flagged for loading with an arrow symbol
- Tag parameters are to be corrected on a running system
- Processing of individual programs is to be monitored
- Individual signals from or to a field object are to be debugged (forcing)
- DVP1 parameters of a Profibus field device are to be loaded.

### F 2.3 Faster switch from commissioning to configuration mode

A quick switch from commissioning to configuration mode has been implemented through the new menu option **Configure** available in the following editors/list displays:

- FBD program,
- LD program,
- IL program,
- SFC program,
- Hardware structure (system structure and station review),
- Variables list, tag list.

The mode change always applies to the entire DigiTool system.

Thus, changing from commissioning to configuration mode within an editor means that if one subsequently moves to the project tree, it will also be in configuration mode.

 When changing mode, the current selection, the current block selection, the current display section, the current search filter and the current sort sequence are all retained within the editors or lists.

 If dialogs such as the 'Define debug window' or the operator dialogs in the SFC program are opened, then the values and trend window will be closed.

 The 'Back' path stored is likewise retained when the mode is changed.

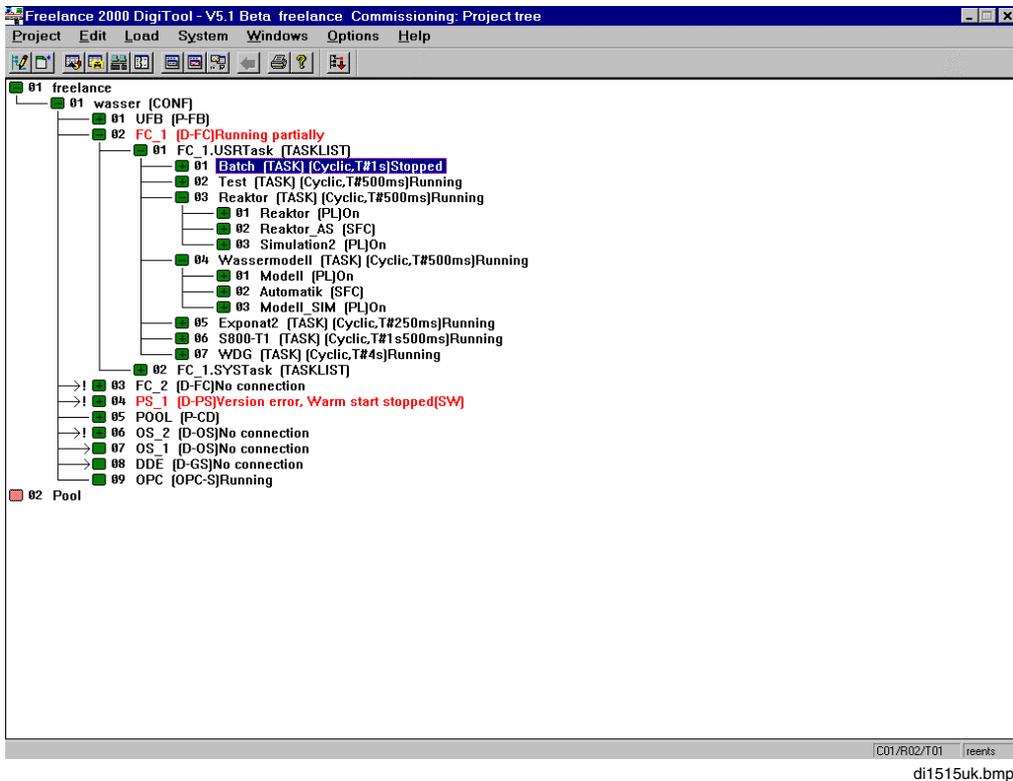
## F 3 Commissioning User Interface

### F 3.1 Project tree

In commissioning mode, as in configuration mode, the project tree forms the starting point for all further operations.

Unlike in configuration mode, state information such as 'running' or 'stopped' is kept for the individual project objects.

Project objects which need to be loaded are shown preceded by an arrow. If a compressed branch contains at least one object that needs to be loaded, then an exclamation point precedes the visible representative (topmost object) of that branch.



For further information see also **chapter Project Tree**.

## F 3.2 Differences between commissioning and configuration modes

### F 3.2.1 Configuration functions

Menu options displayed in bold can only be used during *configuration*, not during *commissioning*.

Project	<b>Save</b> <b>Documentation</b> <b>Check</b> <b>Check all</b> <b>Show error list</b> Header <b>Comment</b> Commissioning Exit	Options	Hardcopy Long state line Colors
		Help	Contents Overview Use Help
Edit	Undo Program <b>Insert above</b> <b>Insert below</b> <b>Insert next level</b> Expand Full expand Compress <b>Cut</b> <b>Copy</b> <b>Paste</b> <b>Delete</b> <b>Export block</b> <b>Import block</b> <b>Import block as redundant</b> <b>Access rights</b> (only on DigiLock) <b>User groups</b> (only on DigiLock)		
System	Variable list Tag list <b>Structured data types</b> Global message processing Local message processing Hardware structure <b>Display access</b> Show global variables Show exported variables Show all objects Show selected objects <b>Tele control signals</b> (only on Tele control library)		

### F 3.2.2 Commissioning functions

Menu options displayed in bold can only be used during commissioning, not during configuration.

Project	Header <b>Configuration</b> Exit	Options	Hardcopy <b>Set system time</b> Long state line Colors
Edit	Program Expand Full expand Compress	Help	Contents Overview Use Help
Load	<b>Whole Station</b> <b>Variables</b> <b>Message configuration</b> <b>Selected objects</b> <b>Changed objects</b> <b>Parameters</b> <b>Show version info</b> <b>Ignore version error</b> <b>Adjust version error</b> <b>Loading all DPV1 parameters (only via shortcut menu)</b>		
System	Variable list Tag list Global message processing Local message processing Hardware structure Show global variables Show exported variables Show all objects Show selected objects		
Windows	<b>Define variable windows</b> <b>Show value window</b> <b>Show trend window</b>		

### F 3.3 State displays in the project tree

In contrast to configuration, state information is maintained on project objects during commissioning. This state information appears after each project object in brackets (like the object types) and is updated as it changes. The state information reflects the state of this object in the process station or FieldController. If arrows should appear before the nodes, these project objects must first be loaded or reloaded into the station because of a configuration change. An exclamation point indicates that other objects at levels below the one so marked need to be updated because of changes in their configuration.

The color of the node in the display provides information about the effects of its configuration change on other objects. Higher-level information may also need to be updated in the station.

In commissioning, **nodes** are displayed in the same formats as in configuration.

See **chapter Project Tree, Display of project object states**

### F 3.4 Representation of the side effects caused by configuration changes

Switching to configuration and carrying out a configuration change can affect the higher-level task or resource (side effects).

The configuration change must therefore be loaded into the higher-level resource.

After the plausibility check, all modified objects are always flagged with an arrow symbol next to the affected node in the project tree. If the project tree is compressed, an exclamation point shows that one or more affected nodes exist below the marked object. See also **Changes with Side Effects on Other Project Objects, page F-33**.

- Programs modified with no side effects are represented by a green node.
- Objects changed and having **side effects on the task** are represented by a yellow node.
- Objects changed and having **side effects on the resource** are represented by a red node.
- **Added objects** are shown along with the side effects produced.
- **Deleted objects** are flagged with an arrow next to the affected program list and task.

 When side effects involve a task or even a resource object, all project objects underneath the affected object, together with the function blocks they use, are held up for the duration of the loading process. See also **Load changed objects, page F-28**.

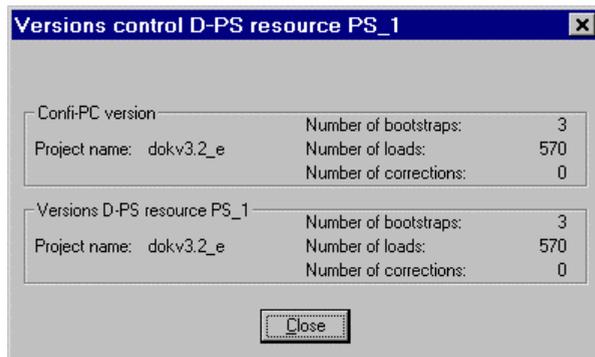
## F 3.5 Project version

### F 3.5.1 Version check

The following version control check is carried out by DigiTool in order to check which project is loaded in the selected station and whether or not this project corresponds to the project currently open in DigiTool. Information on the individual project versions can be displayed as follows.



→Load → Show versions info



di1539uk.bmp

The information can be deduced from the number of times the function blocks and HW components have been loaded (corrections) and undergone parameter changes.

The critical factor is whether the project name in the engineering station matches the one in the process station or FieldController.

The true project version number is stored in system variables. These system variables hold the project version. They are overwritten with current values on initialization or bootstrapping of the resource. The new values are then retained until the next initialization or bootstrapping (even in the case of a cold start).

*Project name*            Name assigned to the project when it was set up.

*Number of bootstraps*

**CMajorVerNo:**        Incremented by 1 on each bootstrapping.

*Number of loads*

**CMinoVerNo:**        Is set back to zero on initialization or bootstrapping. After each successful loading of an object into the station it is incremented by 1, but is not incremented for a cold start.

*Number of corrections*

**CPatchVerNo:**        Incremented by 1 after each successful correction of the block parameters.

### F 3.5.2 Adjust version error

If the DigiNet S connection is interrupted during a load operation, a version error may occasionally occur. In this case the number of load operations in the version data differs between the configuration PC and the station by exactly 1. This version error can be adjusted as follows:



*System* → *Adjust version error*



Adjust **version error** should only be used to reset version errors when there is no doubt that the objects in the station are identical with those specified in the configuration.

### F 3.5.3 Show global variables of the resource

In a D-PS or D-FC resource, a maximum of 32 Kbytes of memory can be allocated for variables. In the case of real values 4 bytes in size, this equates to 8,000 variables per resource.



*System* → *Global variables of the resource*



Select resource → *Project* → *Header* → *SHOW...*



Double-click on resource name → *SHOW...*

Comp.No:	Variable name:	State:	Type:	Offset:
421	Binab_In	CLEAN	BOOL	31915
420	Aow_In	CLEAN	REAL	31916
419	AI_TR_OUT	CLEAN	REAL	31920
418	DDE_LWK	CLEAN	BOOL	31924
417	DDE_LKW	CLEAN	BOOL	31925
416	MAN_AUTO	CLEAN	BOOL	31926
415	Lauf	CLEAN	BOOL	31927
414	TSCont_OUT	CLEAN	REAL	31928
413	RE_Active3	CLEAN	BOOL	31932
412	RE_LOAD3	CLEAN	BOOL	31933
411	RE_Active2	CLEAN	BOOL	31934
410	RE_LOAD2	CLEAN	BOOL	31935
Max. memory size:		32768	Byte	
Size of free memory:		31686	Byte	
Largest free memory block:		31676	Byte	
Free memory in type domain:		60518	Byte	

di1525uk.bmp

Comp. No.:	Sequence number of variable	
Variable name:	Variable label, 16 characters in length	
State:	CLEAN	Object is correct and loaded into station.
	DIRTY	Object version in engineering station does not match object version in the process station.
	CREATE	Object not yet loaded into station.
	DELETE	Object deleted from project database, but still present in station.
Type:	REAL, BOOL, UINT etc.	
Offset:	Memory address offset	

#### F 3.5.4 Show exported variables of the resource

A variable can be accessed for reading and writing. These operations can be performed within a resource. Other resources have read-only access. Furthermore, they can read a variable only if it is defined for **Export**, i.e. if when the variable was originally defined, **Export** permitted to other resources was indicated by a Yes entered in column X of the variable's entry in the list. This state may be changed subsequently.

 A maximum of 1400 bytes per resource can be specified for lateral communication (i.e. for export from one resource to another).

 Where variables are assigned to an I/O component, these or the I/O component itself cannot be written to from other resources.

See **Engineering Manual, IEC 61131-3 Programming, Variables List**

 *System* → *Show exported variables*

The window setup is identical to that of the list of all global variables of the resource.

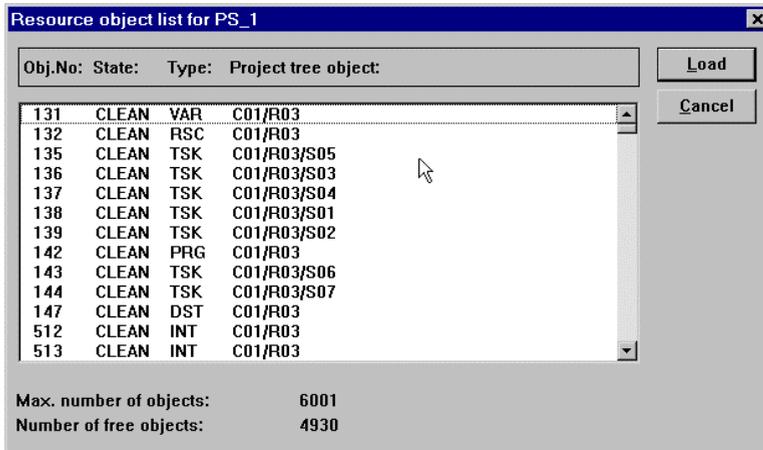
See **page F-17, Show global variables of the resource.**

### F 3.5.5 Show all objects

All objects configured for the project are displayed.



System → Show all objects



di1517uk.bmp

Obj. no	Object number	
State	CLEAN	Object is correct and loaded into station.
	DIRTY	Object version in engineering station does not match object version in the station
	CREATE	Object not yet loaded into station.
	DELETE	Object deleted from project database, but still present in station.
Type	VAR	Variable block
	TSK	Task
	CLS	Non-resident function-block class
	PA	Process image block
	PRG	IL, LD or FBD program
	SFC	SFC program
	FB	Function block
	DEL	Undefined object
	VRS	Version info
	ACC	Access rights
	CODE	Program code
	ICON	Selection icon in the overview display
	GLBL	Global object
RSC	Resource object	
DST	Daylight saving time table	
INT	Object for internal process station management	

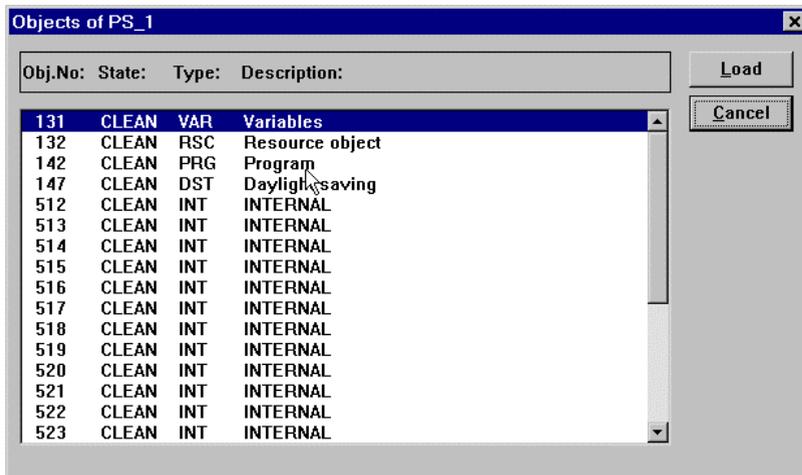
Project tree object	Path in the project tree
Max. number of objects	Number of objects possible in a project
Number of free objects	Number of additional objects possible
LOAD	Load all selected objects
CANCEL	Exit from the object list

 All Objects not being CLEAN or DEL will be cold started when loaded!

### F 3.5.6 Show selected objects

Only the objects below a project object selected in the project tree are displayed.

 *System* → Show selected objects



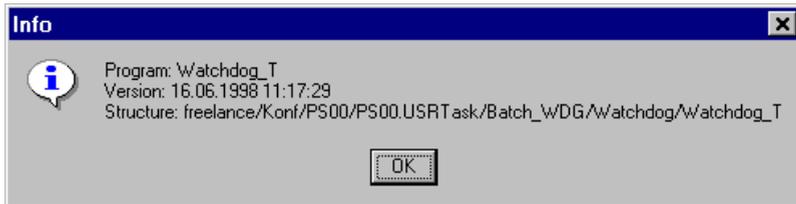
di1518uk.bmp

The setup of this window is basically identical to that of the list of all objects. The only difference is that instead of the project tree path, a brief description of the object is displayed. This description is generated in a fixed form by the system. If, for example, a function block which is included in a graphic image is involved, the tag name is displayed. See also **page F-19, Show all objects.**

### F 3.5.7 Program information



Call program → *Options* → *Version*



di1430uk.bmp

Program	Program name
Version	Date and time of last saved change
Structure	Path of program in project tree

The name of the called program, date of last change and current path in project tree can be looked up in a program's info window.

For a short description of project paths please refer to: **chapter Project Tree**

## F 4 Time Settings

### F 4.1 System time, local time and time zone

**SystemTime:** Current time kept by the station, used for internal transfer time stamps. System time is equal to **Greenwich Mean Time GMT (UTC)**.

**LocalTime:** In addition to **SystemTime**, **LocalTime** is defined for each station. The local time takes account of time zones as follows:

#### **LocalTime = SystemTime - TimeZone**

The default setting, a time zone offset of 0h, is Greenwich Mean Time (GMT). The relation of the local time to GMT is always set upon installation. In the station the local time is available in the system variable **name.DateTime** (name = resource).

**ActualTime** The local time with any daylight saving time shift applied is the **actual time**, i.e. the time which the user sees on his or her watch. This time is used when setting the Freelance 2000 clock time in commissioning and is generally the time used at the Freelance 2000 user interface.

**TimeZone** The time zone (TZ) is calculated from the difference between the Greenwich mean time (GMT) setting and the local time (TZ = GMT - Local Time). For Germany TZ is equal to -1 (at 13:00 GMT it is 14:00 in Germany, 13-14 = -1).

#### F 4.1.1 Set time zone

In Version 3.3 and later the time zone is set for the entire project in the project-tree configuration node.

#### F 4.1.2 Synchronize system time

All Freelance 2000 stations are equipped with a real-time clock. When the system is running, the clock times must correspond throughout the system, so that for example entries can be made in logs and trends in the correct time sequence. For this to occur, the system time must be synchronized at all stations. A system master (a process station) leads all the other stations physically connected to an Ethernet bus during the synchronization process. The master is generally the resource or station with the lowest station number. This station synchronizes the clocks of all other stations. Station number settings can be checked under *System* → *Hardware structure* → *Network*.

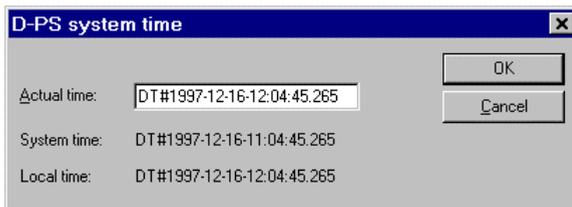
The leading station (master) synchronizes all other stations (slaves) once per minute. Discrepancies of less than one second are equalized by adjusting the system clock. In the case of greater deviations, the time is set by means of a time jump. An accuracy of approximately 3 ms can be achieved, but discrepancies of up to 5 ms are quite possible.

 System time synchronization only occurs if the clock time has been set at least once by the engineering station.

### F 4.1.3 Set system time

The system time can be set system-wide from the engineering station.

 *Options* → *Set system time*



di1537uk.bmp

The actual Freelance 2000 time is shown in the *Actual Time* field. Here the user can enter the correct time. If the current time is daylight saving time, this is indicated by an “S” following the time value.

The system time is calculated from the actual time set in commissioning and loaded in the time-leading station (master):

Local time = actual time – daylight saving time shift

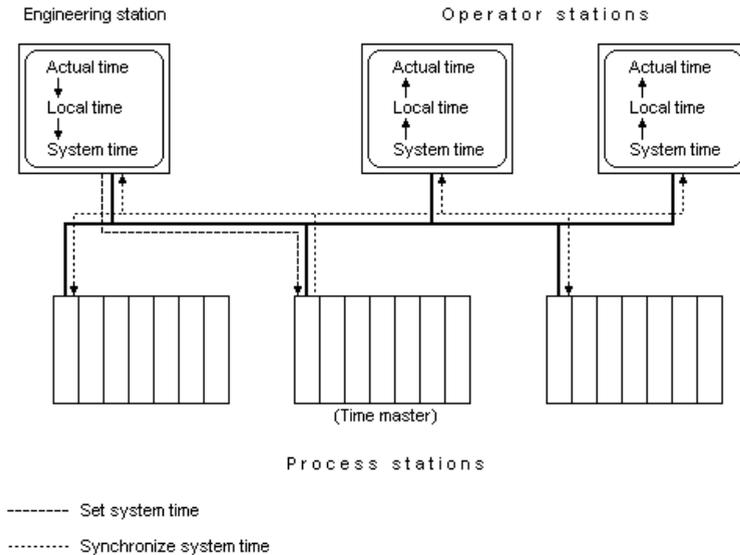
System Time = Local time + Time Zone

#### Example:

TimeZone (TZ)= -2, daylight saving time shift = 1h:

Actual time = 14:00S, LocalTime = 13:00, SystemTime = 11:00

This new system time is transferred to all other resources (Slaves) via **system time synchronization**. These resources then convert the new system time to the individual local time of each.



tg001uk.bmp



The system variable **Name.DateTime** (Name = resource) exists in each resource. The local time used for time-controlled events is obtained from this variable. If this variable is displayed in the value window, the actual variable content is shown. Even when using daylight saving time, the daylight saving time shift is never added here.

#### F 4.1.4 Writing DT variables

Activation of daylight saving time only effects the display of time points. When operating on a variable, the user must specify whether the edited time is a daylight saving time or not. A daylight saving time must be identified by an "S" following it. If this "S" is missing, the time value input is interpreted as local time. If an "S" is specified for a time when daylight time is not in effect, a message is sent to the user requesting a correction.

**Example:** Input `..16:00..` produces 16:00 at the station; an input of `..16:00..S` produces (daylight saving time in Germany) 15:00.

#### F 4.2 Daylight saving time

In Version 3.3 and later, daylight saving time is set for the entire project in the project-tree configuration node.

## F 5 Loading the Project

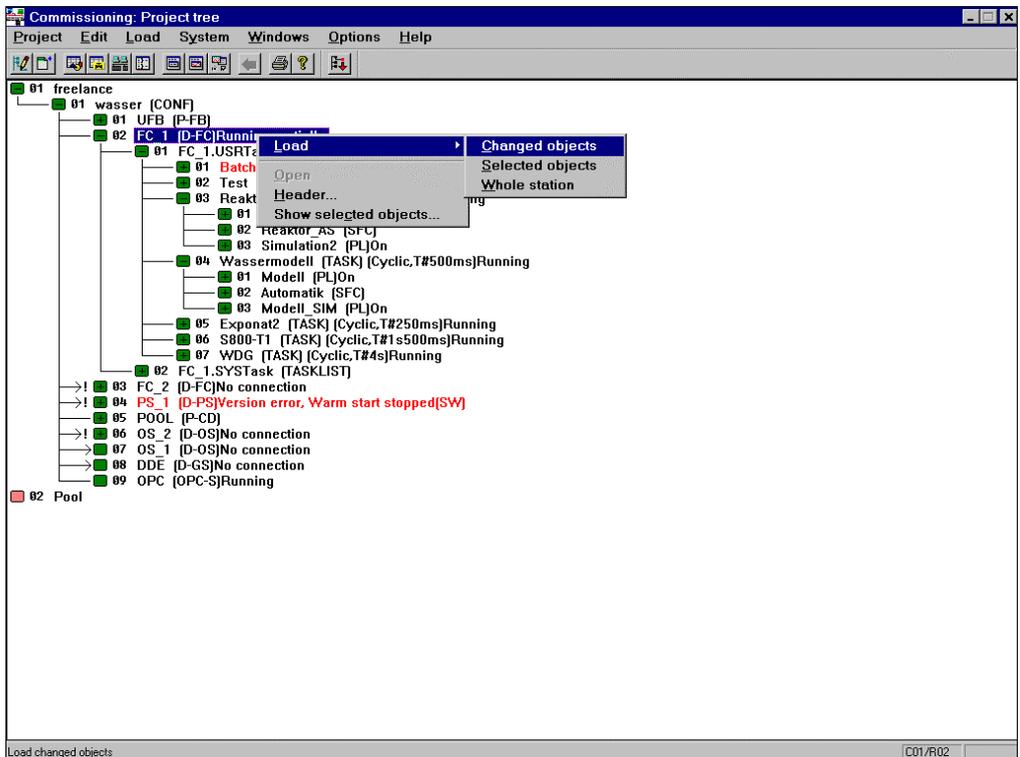
During loading, the user program, or parts of the user program, are loaded into the process station.

In this procedure, either the *Whole station* can be loaded, or parts of the user program.

Changes to *Variables* and *Message configuration* can be loaded individually into the station, as can changes to individual objects.

Via load → *Selected objects* all the selected project objects in the project tree are loaded; via load → *Changed objects* only changed objects or program blocks (including individual function blocks) are loaded.

 When the system is running, all changes are brought up to date through *Changed objects*. As a result, however, care must be taken in case of **side effects** on D-PS resources and tasks. The side effects are indicated in the project tree by displaying the affected nodes in different colors. See **chapter Project tree, Display of project object states**.



## F 5.1 Load whole station

All parts of the user program associated with the selected resource are loaded. This is achieved by first deleting the user program from the resource. The output modules and field devices assume their configured safety values. This may mean retaining the last value or assuming a fixed value, settable channel by channel.

If when loading the whole station it becomes clear that a different project is being loaded, the I/O modules are also initialized. The output modules output zero voltage/ current in this process.

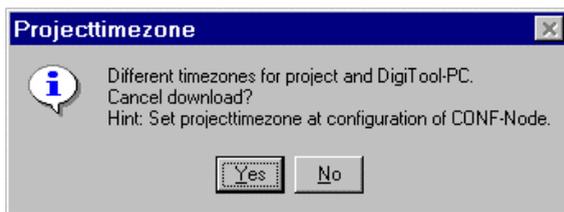
If no operating system has been loaded in the resource at the time of loading the whole station, the station's operating system is installed automatically with initial loading.

 When there are field devices connected (Profibus), only the user parameters are loaded. DPV-1 parameters must be loaded manually. See **Engineering Reference Manual, Communications and Fieldbuses, Profibus**.

 *Load → Whole station*

 Regardless of which project object within the resource is selected, on *Load → Whole station* the complete station (D-PS, D-FC, D-OS or GWY) is deleted and reloaded. The user must ensure that this resource belongs to the loaded project. The result of the system's version check is merely displayed without interrupting the load process.

If the time zone of the project does not match that of the DigiTool PC, a dialog box is displayed which draws attention to the discrepancy:



tg009us.bmp

Yes The download operation is canceled.

No The download operation is continued.

## F 5.2 Load variables

Freelance 2000 includes both predefined and user-defined variables. Predefined variables are set up automatically in the resource and created in the station resource at the time of bootstrapping.

User-defined variables are those variables defined by the user, either in the programs or added directly to the variable list.



Loading of variables entails initialization of the variables concerned. This process erases all previous information concerning the variable. Programs running in other objects which rely on information from the variables being loaded can thus react in ways which, under certain circumstances, may be undesirable. If, for example, the export attribute of a variable is changed, then this variable will be reloaded and its value will be reset to the initial value. It is therefore important to be very sure that the loading of changed objects does not have a negative influence on the process being controlled.



*Load → Variables*

All variables belonging to the selected resource, including exportable variables, are loaded into the station. The size of the global variable area in the resource is adjusted accordingly.



Load variables applies only to user-defined variables, not to this resource's system variables.

All variables are entered in the variable list. No variable may ever be allocated to more than one resource. For further information see **Engineering Manual, IEC 61131-3 Programming, Variables.**

## F 5.3 Load message configuration

The station-specific (local) message processing configuration is loaded into the selected station. This includes specifications of message and acknowledgment handling, audible warning (horn) control and relating to the message list and message line.



*→ Load → Message configuration*

See also **Engineering Manual, Operator Station, Messages and Hints**

## F 5.4 Load selected objects

Loading one or all of the selected project objects without their message configuration, variables or hardware structure segments.



Loading of objects entails initialization of the objects concerned. This process erases all previous information concerning the object. Programs running in other objects which rely on information from the objects being loaded can thus react in ways which, under certain circumstances, may be undesirable. If, for example, the export attribute of a variable is changed, then this variable will be reloaded and its value will be reset to the initial value. It is therefore important to be very sure that the loading of changed objects does not have a negative influence on the process being controlled.



When changed or selected objects of a resource are loaded, if the resource itself is not selected a check is made as to whether all function block classes (domain classes) of the resource have CLEAN status. If there is a domain class that does not have CLEAN status, then a message box appears recommending loading at the resource level: "Changed function block classes have to be downloaded. Please select resource node and load changed object first here". The download below the resource level is not carried out. When loading individual domains of a resource, the check is made independently of the object selected.



When there are field devices connected (Profibus), only the user parameters are loaded. DPV-1 parameters must be loaded manually. See **Engineering Reference Manual, Communications and Fieldbusses, Profibus**.



→ Load → Selected objects

## F 5.5 Load changed objects

The smallest loadable unit of a project is the object. There are visible project objects, such as the resource, task, program list and program, and there are hidden objects such as the process image, function block, variables and messages.

When Load *Changed objects* is executed, only the project's changed objects are loaded. Care must be taken in case of side effects on the higher-level task or resource.



Loading of objects entails initialization of the objects concerned. This process erases all previous information concerning the object. Programs running in other objects which rely on information from the objects being loaded can thus react in ways which, under certain circumstances, may be undesirable. If, for example, the export attribute of a variable is changed, then this variable will be reloaded and its value will be reset to the initial value. It is therefore important to be very sure that the loading of changed objects does not have a negative influence on the process being controlled.

-  When only object deletions have occurred, *Load → Changed objects* will delete the objects will on the station only.
-  When changed or selected objects of a resource are loaded, if the resource itself is not selected a check is made as to whether all function block classes (domain classes) of the resource have CLEAN status. If there is a domain class that does not have CLEAN status, then a message box appears recommending loading at the resource level: “Changed function block classes have to be downloaded. Please select resource node and load changed object first here”. The download below the resource level is not carried out. When loading individual domains of a resource, the check is made independently of the object selected.
-  All Objects not being CLEAN or DEL will be initialized when loaded!
-  If after a project change only the version information of an OPC or DDE gateways has changed, in version 6.1 and later the version information will be automatically loaded into this gateway when loading a process station.
-  When there are field devices connected (Profibus), only the user parameters are loaded. DPV-1 parameters must be loaded manually. See **Engineering Reference Manual, Communications and Fieldbusses, Profibus**.
-  *Load → Changed objects*

## F 5.6 Uploading current block parameters

The current working data of a block can be modified by manual entry from the commissioning process, or from DigiVis via a gateway or DDE server. The changed parameters will be lost if a cold start is performed or if a download is used for reconfiguration.

Individual values can be modified in the commissioning phase in such a way that they are not lost during a cold start by using the CORRECT function.

Larger amounts of working data can be saved using the **Upload parameters** function

**Procedure**

- Select a station or a section of the project tree



→ Load → Parameters...

- All the block parameters below the selected project tree node are read and displayed in a list.
- Each parameter is displayed with the following information: parameter name, data type, tag name, short text, project path, block class, configured value and current value.
- The list can be sorted on any one of these fields (by pressing the appropriate column button).

At the beginning of each line is a check box that is used for specifying whether or not the parameter is to be corrected.

Parameter	Type	Tag	Shorttext	Path	Class	Config. Val...	Actual Value
<input checked="" type="checkbox"/> Gw1	REAL	TIC1217		C01/R01/...	C_CU	1120.0	1107.0
<input checked="" type="checkbox"/> Gw2	REAL	TIC1217		C01/R01/...	C_CU	830.0	819.0
<input checked="" type="checkbox"/> Wi	REAL	TIC1217		C01/R01/...	C_CU	980.0	992.0
<input type="checkbox"/> Yi	REAL	TIC1217		C01/R01/...	C_CU	70.0	51.61492
<input type="checkbox"/> Py	INT	TIC1217		C01/R01/...	C_CU	0	1
<input checked="" type="checkbox"/> Gw1	REAL	TIC1305		C01/R01/...	C_CU	1120.0	1126.0
<input checked="" type="checkbox"/> Gw2	REAL	TIC1305		C01/R01/...	C_CU	830.0	829.0
<input checked="" type="checkbox"/> Wi	REAL	TIC1305		C01/R01/...	C_CU	980.0	983.0
<input type="checkbox"/> Yi	REAL	TIC1305		C01/R01/...	C_CU	70.0	57.03508
<input type="checkbox"/> Py	INT	TIC1305		C01/R01/...	C_CU	0	1
<input checked="" type="checkbox"/> Gw2	REAL	TIC1307		C01/R01/...	C_CU	830.0	842.0
<input checked="" type="checkbox"/> Wi	REAL	TIC1307		C01/R01/...	C_CU	980.0	956.0
<input type="checkbox"/> Yi	REAL	TIC1307		C01/R01/...	C_CU	70.0	78.0

tg004uk.bmp

The contents of the list display can also be modified.

**Parameters**

*all*

All parameters under the selected project tree node are displayed.

*only different*

Only those parameters are listed whose current value is different from that contained in the database (standard setting).

*not corrected* (Only worth selecting after a correction process has been carried out.)  
 Those parameters are displayed that would have been corrected but for the fact that the process was not completed successfully.  
 On completion of a Correct process this radio button is selected automatically.

### Search criteria

Another way of modifying the parameter list is by using search criteria:

- Parameter type,
- Tag name,
- Class name.

Each of these criteria may include the wildcards “\*” and ‘?’.  
 Class can also be selected from a list.

After the SEARCH button is pressed, those parameters are displayed which satisfy the search criteria entered. Search criteria only take effect when radio buttons **all** and **only different** are selected.

 Correction identification (check box) for a marked block can be completely activated or deactivated with the SPACEBAR.

CLOSE The dialog is closed.

UPLOAD ALL parameters are re-read from the working data and the list is regenerated using the search criteria set.

CORRECT The current values of the selected parameters in the list are copied to the project database.



All **selected parameters** are copied to the database regardless of any active search criteria!

EXPORT The parameters currently displayed - which depend on the active search criteria - are transferred to a file <name>.upl. A prompt to enter the file name follows.

PRINT The parameters currently displayed - which depend on the active search criteria - are printed out.

HELP A language-specific on-line help window is opened giving the description of the module parameter.

### F 5.7 Delete project from the process station

The station can be erased in various ways. For this the D-PS resource should be selected in commissioning via



Select resource → *Project* → *Header* or by double clicking on the resource

With INITIALIZE ALL, everything is deleted, even the operating system. INITIALIZE the resource deletes the complete user program from the station, and in contrast to cold start, also sets the output modules to zero voltage/current. Although all current data and variable values are deleted in COLD START, the user program is unaffected and the output module channels assume their configured safety values.

In *Load* → *Whole station* the station is likewise first erased.

See also **page F-26, Load whole station** and **page F-17, Adjust version error**.

### F 5.8 Ignore version error

If the version number of the project currently open does not match that of the software loaded on the station, a version error will result.

If loading is rejected, this version error can be bypassed by selecting *Load* → *Whole station*.



If after a project change only the version information of an OPC or DDE gateways has changed, in version 6.1 and later the version information will be automatically loaded into this gateway when loading a process station.

See also **page F-16, Version check** and **F-17, Adjust version error**.

## F 6 Changes with Side Effects on Other Project Objects

The possible side effects of a configuration change can be classified in three groups:

- **No side effects**

The objects involved in the configuration change can be loaded into the task or resource running them without stopping it.

- **Side effects on the task**

The objects involved in the configuration change can only be loaded after the task has been stopped. Example: a change in an IL or FBD program also causes a change in the process image of a task. The program can only be run again when both the program and the process image of the task have been updated.

- **Side effects on a resource**

The objects involved in the configuration change can only be loaded after the resource has been stopped. Example: a change in an IL or FBD program also causes a change the resource global variables. The program can only be run again when both the program and the global variables of the resource have been updated.

### F 6.1 Display of changes in the project tree

After the plausibility check, all modified program blocks are always flagged with an arrow symbol next to the affected node. If the project tree is compressed, an exclamation point shows that one or more affected nodes exist below the marked object.

- Programs modified with no side effects are represented by a green node.
- Objects changed and having **side effects on the task** are represented by a yellow node.
- Objects changed and having a **side effect on the resource** are represented by a red node.

 The indication of changes will be passed from the affected object in the project tree to the resource node. Only the most far-reaching side effect will be displayed.

See also **chapter Project tree, Display of project states**.

## F 6.2 Changes with no side effects

- Adding functions and function blocks only
- Adding functions and function blocks with links to variables in the those programs
- Adding a new task (or moving one out of the pool)
- Deleting functions
- Deleting a task (or moving it into the pool) ⇒ ...side effects on the operator station!
- Moving tasks from one resource to another
- Changing task parameters such as autostart, priority, execution
- Changing the program list parameter *on/off*
- Changing function block parameters
- Changing the processing sequence within a program



If after a project change only the version information of an OPC or DDE gateways has changed, in version 6.1 and later the version information will be automatically loaded into this gateway when loading a process station.

## F 6.3 Changes with side effects on the task

- Adding functions and function blocks with links to variables already known to the system
- Deleting function blocks
- Deleting programs or program lists (or moving them to the pool)
- Adding programs or program lists (or moving them out of the pool)
- Moving programs or program lists from one resource to another
- Modifying the processing sequence for project objects at a level lower than the task
- Modifying the process image, e.g. by entering a new variable in the variable list
- Changing a task's interval duration

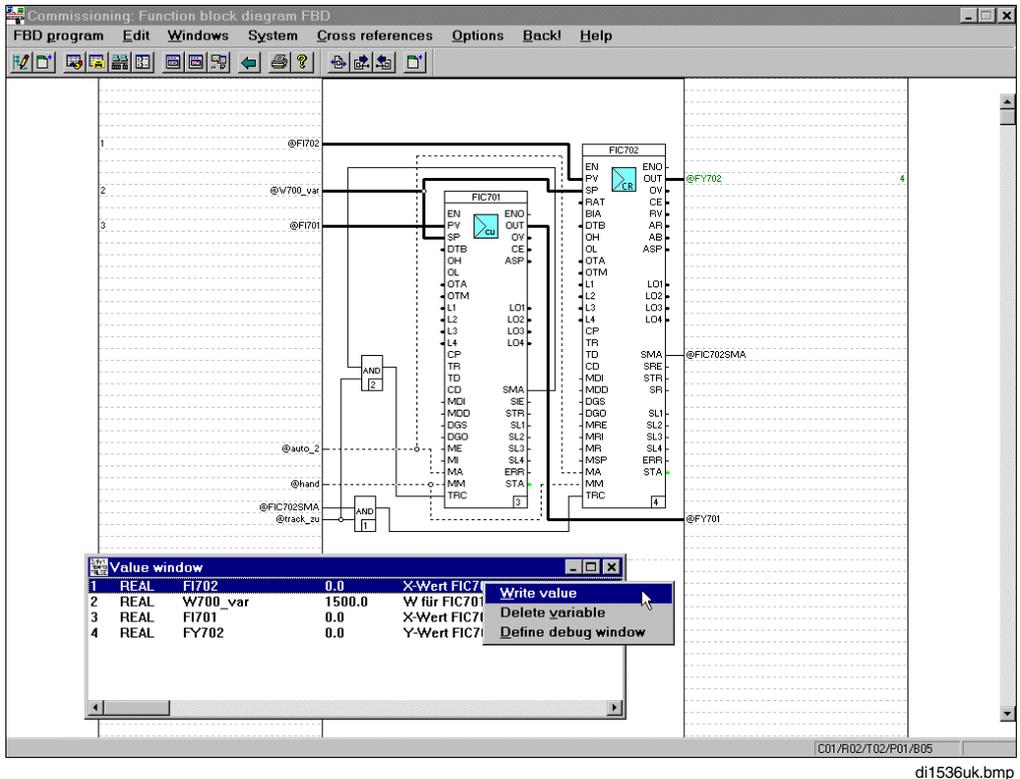
## F 6.4 Changes with side effects on the resource

- Adding functions or function blocks with links to variables not yet known to the system
- Adding links to variables which are not yet known to the system
- Loading boot EPROM (Update/Upgrade)
- Changing the data type of a variable
- Creating or changing the assignment of a variable to a resource
- Changing, deleting or assigning to a variable of I/O components
- Defining a variable as an *Export* variable (which leads to intercommunication between resources)
- Changing memory allocation for variables

## F 7 Displaying and Writing Values in a Window

During commissioning simulating and displaying signals, it is important to be able to carry out online checks on parameter changes during processing.

The commissioning window makes it possible to observe a variable. This enables the user to intervene in an online process from the engineering station.



*Define debug windows*

Settings established for debug and trend windows

*Show value window*

*Show trend window* Make it possible to display the process signals and variables requested in the form of values and trends

*Enter variable*

Variables are introduced into the value or trend window

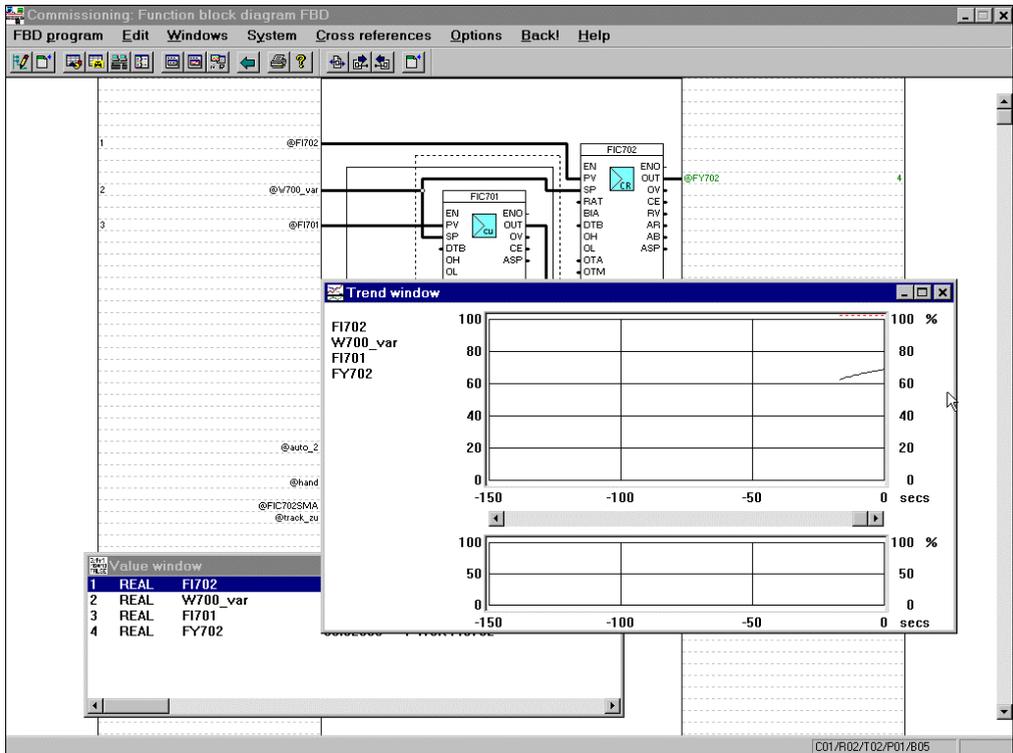
*Write value*

Enables the one-time write of a binary value

The value window can display numeric data - which can also optionally be displayed in a trend window - and strings.

The values for the value and trend windows are updated once per second.

Both windows can be displayed simultaneously and remain on screen when switching programs. This enables variables from other programs to be inspected in conjunction with the program being viewed.



In conjunction with the application from which it is started, the Window menu opens up a variety of possibilities.

	Project-tree	Variables list	Tag list	Function Block Diagram (FBD)	Instruction List (IL)
Define debug window	●	●	●	●	●
Show value window	●	●	●	●	●
Show trend window	●	●	●	●	●
Enter variable	-----	●	-----	●	●
Write value	-----	●	-----	●	●
Processing On/Off	-----	-----	●	-----	-----

- selection can be made from corresponding menu
- selection cannot be made from corresponding menu.

### F 7.1 Enter variable

The menu command *Enter variable* can be selected from within the instruction list, the variable list and the function block diagram (FBD).



Select variable (via mouse click) → *Window* → *enter variable* → **Define debug window**

The selected variable is added to the variable list in the commissioning window. The *Define debug* window appears on the screen and the variable can immediately be assigned a display format.

As soon as a variable is entered, it has a sequence number appended to it in the instruction list and in the FBD. This number reflects the order in which variables are entered.

In FBD it is possible to display signal-line values in a value window and/or a trend window. Signal-line values are deleted from the debug window when exiting the FBD program. Signal-line values have no names in the debug window, they can be identified only by sequence number.



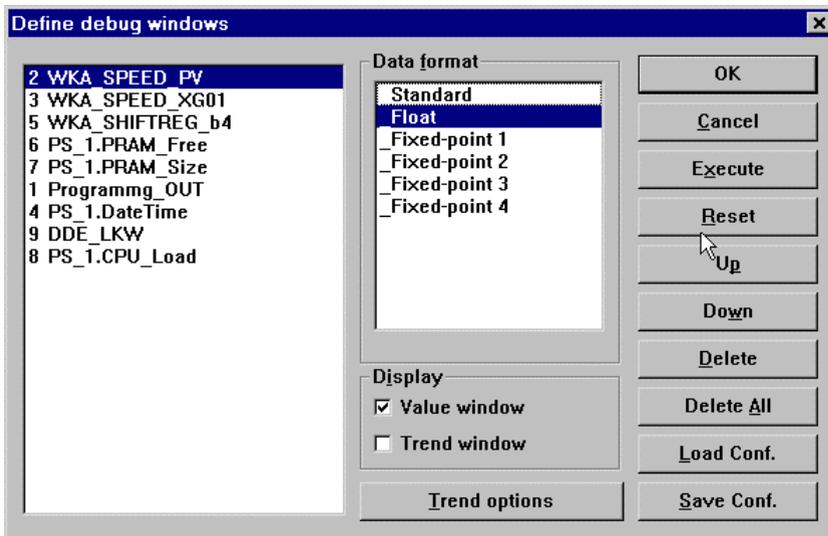
Select signal line via mouse click → *Window* → *Enter variable*  
→ Define debug window



In the FBD, a double mouse click on a variable name or the signal line is all that is required to enter it.

### F 7.1.1 Define debug windows

This part of the program is where a variable is allocated a display format. The variable to be processed is first selected in the list with a mouse click. Now the display format for the **value window** and **trend window** can be selected. Entries can be added to the variable list by calling the menu option *Enter variable* or directly by double-clicking on the variable. These entries may be saved and reloaded.



di1650uk.bmp

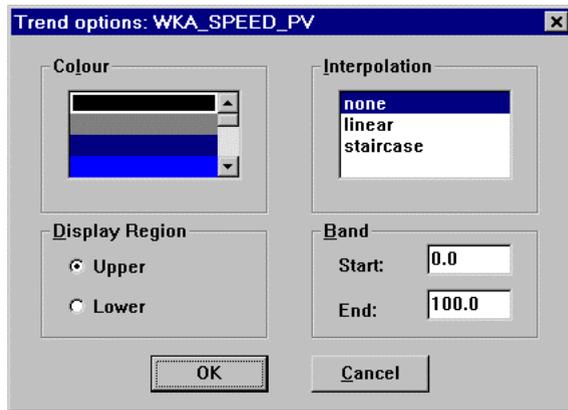


*Window* → *Define debug windows*

OK	Save entries and exit from <i>Define debug windows</i> .
CANCEL	Exit from <i>Define debug windows</i> without saving entries
EXECUTE	Activate entries in the value and trend windows without exiting <i>Define debug windows</i> .

RESET	Cancel the last entry.
Up	Shift the selected variable <b>up one position in the list</b> . The variable's sequence number remains unchanged.
DOWN	Shift the selected variable <b>down one position in the list</b> . The variable's sequence number remains unchanged.
DELETE	Delete the selected variable from the variable list in the value and trend windows.
DELETE ALL	Delete all variables from the trend and value windows.
LOAD CONF.	Load a stored configuration, selected from a list.
SAVE CONF	Save the current configuration, (followed by a prompt for a name under which to save it).
<b>Data format</b>	Display format for selected variables in the value window. After the required variables have been selected, all the valid data format options appear here on the screen. An appropriate display format for a variable can be chosen by clicking on the relevant format. In the value window, a variable can be displayed simultaneously in all the data formats offered. (Displaying a <b>variable of data type DT</b> in a format that differentiates daylight-saving time: if the value of the variable falls within daylight-saving time, then this value is increased by the daylight-saving time difference (1 hour in Germany), and the value is flagged by appending an 'S').
<b>Display</b>	Designate a variable for display in the value window and/or trend window by clicking on the appropriate box until a check mark appears..
Value window	<input checked="" type="checkbox"/> Variable will be displayed in the value window; data display formats available for the variable will be listed. <input type="checkbox"/> Variable will not be displayed in the value window.
Trend window	<input checked="" type="checkbox"/> Variable will be displayed in the trend window, the <i>Trend options</i> menu will be called up.   Variables of the <i>string</i> data type cannot be displayed in the trend window. <input type="checkbox"/> The variable will not be displayed in the trend window.
TREND OPTIONS	Calls up <i>Trend options</i> menu

## F 7.1.2 Trend options



di1651uk.bmp



Window → Define debug windows → Trend options

**Color** Choice of color for display of selected variables in the trend window.

**Interpolation** Three different interpolation methods are available:  
 None  
 Linear  
 Staircase

**Display region** Choose one of the two variable windows, in which the selected variable should appear. The Y-axis of both windows extends from 0-100%. The upper window is five times as high as the lower window. Each X-axis covers 150 values, (i.e. seconds, since recording in the process station is fixed at one reading per second). Both trend windows are displayed simultaneously on the screen. The *Lower* window can display a maximum of 4 different signals, while the *Upper* one can display up to 18.



A maximum of 20 variables can be selected for display. If all 20 values are entered in the variable window and *Show trend window* is executed, the engineering station will be overloaded. A routine is then executed which declares these variables invalid.

**Band** This is where limits are entered for the display of selected variables in the trend window. Data must be entered in an appropriate format for the variable.

## F 7.2 Show value window

The screenshot shows the Freelance 2000 software interface. The main window displays a ladder logic network. A network with a timer T1 is shown, with a coil N1704. The network includes logic elements such as AND, OR, and SEL. The Value window dialog box is open, displaying a list of variables with their data types and current values.

No.	Data type	Variable name	Value	Comment
1	BOOL	A11B2N	FALSE	
2	BOOL	A11B3N	FALSE	
3	BOOL	A11B4N	FALSE	
4	BOOL	A11_LOW_LIMIT	FALSE	
5	BOOL	A11_NO_CDMM	FALSE	
6	BOOL	A11_NO_CDMM_A...	FALSE	
7	BOOL	A11_OUT_OF_RA...	FALSE	
8	BOOL	A11_SUMMEN_ST	FALSE	
9	BOOL	A12B1N	FALSE	
10	BOOL	A12B2N	FALSE	
11	BOOL	A12B3N	FALSE	
12	BOOL	A12B4N	FALSE	
13	REAL	F1701	0.0	X:Wert FIC701
14	REAL	F1702	0.0	X:Wert FIC702
15	REAL	F1704	747.1394	X:Wert FIC704



→ Window → Show value window

The current values of variables can be displayed in the value window. Each variable is displayed in a format depending on its data type. Variables of the BOOL data type are displayed with the logical state “True” or “False”. Variables of the data type REAL, INT, TIME and WORD are shown with their value, and variables of the STRING data type are shown with their text content.

The value window consists of the following five columns, from left to right on the screen:

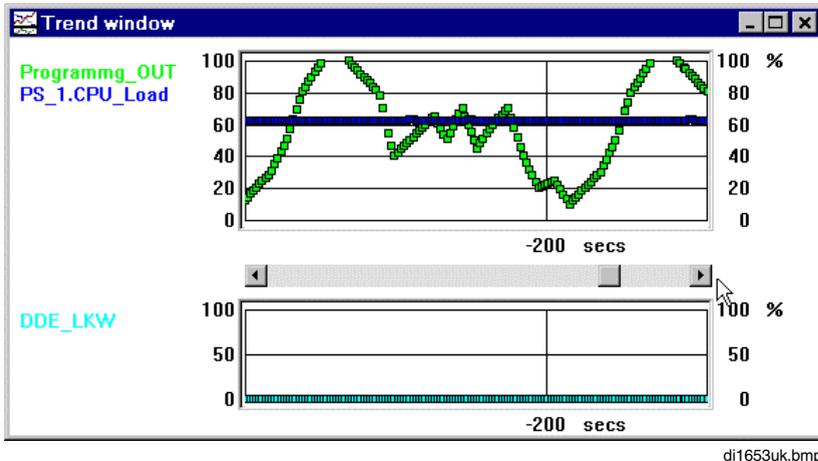
**Input (sequence) number**, from *Define debug*  
**Data type**,  
**Variable name**,  
**Value or State**,  
**Comment**



The setting of UseDaylightSavingTime is ignored for the display of DT variables in the value window. In commissioning mode, the user always sees the actual variable contents. (UseDaylightSavingTime is only taken into account in dialogs for setting the clock under DigiTool).

### F 7.3 Show trend window

 → Window → Show trend window



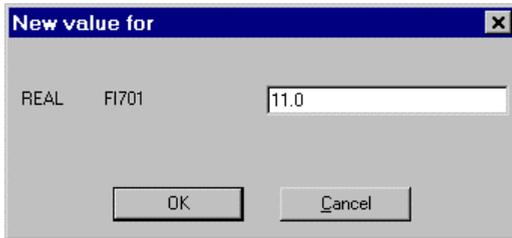
In the trend window, process signals covering a time span of 1,000 seconds can be displayed. At any one time, a time span of 150 seconds is visible. The trend window has no “memory”: upon leaving the trend window, the values recorded are lost.

In the lower border of the window is a push button used for scrolling back the display. By activating this button it is possible to see earlier signal sequences.

## F 7.4 Writing a value



*Window* → *Write value*



di1655uk.bmp

*Write value* can be used to assign a new value to a variable. When making such an entry, care must be taken to ensure that the entry is in the correct data format (as defined for the variable).



When *Write value* is called up, the window is displayed in the middle of the screen. If there is a value or trend window already displayed, the *Write value* window will be located behind that window. The value or trend window must be repositioned or closed before the *Write Value* window can be used.



The variable is only overwritten for one cycle. If, in the next program cycle the variable is rewritten, the value specified here, in commissioning, is overwritten. To force a value, the variable must be forced at the I/O module.



## G Documentation

Name: PROJECT	
<input type="text" value="Documentation scope project tree"/>	<input checked="" type="checkbox"/> Project tree >>
<input type="radio"/> All selected objects on the same level	<input checked="" type="checkbox"/> Programs >>
<input checked="" type="radio"/> All successive objects	<input checked="" type="checkbox"/> Variables list >>
<input checked="" type="checkbox"/> Coveragepage	<input checked="" type="checkbox"/> Variables cross reference >>
<input checked="" type="checkbox"/> Index	<input checked="" type="checkbox"/> Tags list >>
<input checked="" type="checkbox"/> Documentation settings	<input checked="" type="checkbox"/> Tags cross reference >>
	<input checked="" type="checkbox"/> Hardware structure >>
	<input checked="" type="checkbox"/> Message processing >>
	<input type="checkbox"/> Plausibility check errors >>
	<input type="checkbox"/> Data types >>
	<input type="checkbox"/> Display access >>
Comment :	<input type="text" value="Documentation FL2000DOCU.PRO"/>
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	



## Contents

<b>G 1</b>	<b>General Description - Documentation</b> .....	<b>G-5</b>
<b>G 2</b>	<b>Compilation of the Drawing Header / Footer</b> .....	<b>G-7</b>
G 2.1	General information.....	G-7
G 2.2	Project-wide drawing header / footer .....	G-7
G 2.2.1	Editing the drawing header.....	G-8
G 2.2.2	Editing the header title.....	G-9
G 2.2.3	Editing the drawing footer.....	G-10
G 2.2.4	Editing the footer title.....	G-11
G 2.3	Program-specific drawing header / footer .....	G-12
G 2.4	Automatic allocation of object parameters .....	G-12
G 2.5	Bitmaps in the drawing footer.....	G-13
<b>G 3</b>	<b>Define the Documentation Job</b> .....	<b>G-14</b>
G 3.1	Creating a new print job .....	G-14
G 3.2	Detail of the print job .....	G-14
G 3.2.1	Documentation scope.....	G-15
G 3.2.2	Cover page.....	G-15
G 3.2.3	Index.....	G-16
G 3.2.4	Documentation settings.....	G-17
G 3.2.5	Project tree .....	G-18
G 3.2.6	Program documentation .....	G-18
G 3.2.7	Variables list .....	G-20
G 3.2.8	Variables crossreference list .....	G-22
G 3.2.9	Tags list .....	G-24
G 3.2.10	Tags crossreference .....	G-26
G 3.2.11	Hardware structure .....	G-28
G 3.2.12	Message processing .....	G-32
G 3.2.13	Plausibility check errors.....	G-33
G 3.2.14	Structured variables .....	G-33
G 3.2.15	Display access .....	G-34
<b>G 4</b>	<b>Documentation Scope</b> .....	<b>G-35</b>
G 4.1	Hard disk requirement.....	G-35
G 4.2	Viewing selected documentation scope .....	G-35
G 4.3	Sorting the output .....	G-36
G 4.4	Selecting the project tree objects .....	G-37
G 4.5	Selecting the Hardwarescope .....	G-38
G 4.6	Selecting notify print errors.....	G-39
<b>G 5</b>	<b>Outputting Documentation</b> .....	<b>G-40</b>
G 5.1	Print .....	G-40
G 5.2	Print file.....	G-41
G 5.3	Printer setup .....	G-42

<b>G 6</b>	<b>Documentation Manager.....</b>	<b>G-43</b>
G 6.1	Editing the documentation manager.....	G-43
G 6.2	Changing the column width.....	G-43
G 6.3	Menu Structure Documentation.....	G-44
G 6.4	Documentation.....	G-45
G 6.4.1	Insert new documentation job.....	G-45
G 6.4.2	Copy new documentation job.....	G-46
G 6.4.3	Comment.....	G-46
G 6.4.4	Export contents.....	G-47
G 6.4.5	Exit.....	G-47
<b>G 7</b>	<b>Comment field editing.....</b>	<b>G-48</b>
G 7.1	Edit Comment Field.....	G-48
G 7.2	Export comment field.....	G-48
G 7.3	Import comment field.....	G-49
<b>G 8</b>	<b>Tables.....</b>	<b>G-50</b>
G 8.1	Document types.....	G-50
G 8.2	Description of the fields or Contents.....	G-53
G 8.3	Field names in the drawing footer / header.....	G-54
G 8.4	Variables for drawing footer/ header inscriptions.....	G-56
G 8.5	Fonts, national languages and bitmaps in the drawing footer / header.....	G-57
G 8.5.1	Using fonts in the drawing footer.....	G-58
G 8.5.2	Using fonts in the drawing header.....	G-58
G 8.6	Presetting the field contents and titles.....	G-59

## G 1 General Description - Documentation

All configured objects can be documented from DigiTool. Likewise, provision has been made to ensure easy location of individual documents as well as precise incorporation into an overall plant documentation.

ABB		Freelance	
Name:	f12000docu	Start:	/
Comment:	Demo Project Freelance 2000	End:	/
			COV
<b>Project Documentation</b>			
<i>Freelance 2000</i>			
Project file name:	f12000docu.PRO		
Comment:	Demo Project Freelance 2000		
Orderer:	Product Management		
Date:	06/19/2000		
Project number:	123456		
Order number:	123456789012		
Project manager:	PL6		
Generation date:	06/19/2000 15:14:20		
Date		Customer	
Resp.	PL6	Product Management	
Check		ABB	
Origin	Doc.	COVERPAGE	
Doc. No.	123456	Customer No.	123456789012
Prog. No.	123456	Doc. No.	2001
Doc. T.		Doc. T.	

ti009us.eps

A common layout has been used for all documents. It is possible to incorporate the company logo into the document header or footer. Document handling is facilitated by a flexible sorting on outputting documents, as well as by a complete table of contents for all project parts output.

An unequivocal designation of the document type (Doc. Type) using the document identification key facilitates location of documents. For example, &EFT20 for electrical engineering, function chart, FBD program.

An automatically inferred documentation identification (Doc ID) refers to the functional environment (=) via the program name and to the location environment (+) via the cabinet identifier. Specification or also output is possible via the customer drawing number.

A direct cross-reference can be found in the program sheet, while the cross-reference list constitutes a supplement to the program sheet for larger scopes, so that all references can be easily located. On displaying variables, the allocated resource can always be recognized. If an input or output channel is allocated directly to a variable, the former is displayed also.

Whenever possible, a tabulated presentation is chosen in order to reduce the quantity of paper used. Otherwise, presentations are made in masks, making it easier to read them.

In tabulated form:

- Object attributes of dynamic graphic displays
- Step and transition parameters of the sequential function chart
- Criteria window of the sequential function chart

In mask form:

- Function block parameters
- SFC parameters
- Module parameters
- Logs and trends
- Network adjustment

Documentation output is controlled by the print job, i.e. a job must always be selected for printing out. The contents of the print job are defined in such a job. Each print job is valid for all projects on the engineering station.

## G 2 Compilation of the Drawing Header / Footer

### G 2.1 General information

In order to be able to enter data into the drawing header / footer, the header and footer mask has been divided into several horizontal levels, so as to enhance the clarity of layout.

### G 2.2 Project-wide drawing header / footer

A drawing header/footer system-wide valid for the project is defined on the project level. All settings made in the header or footer are saved in the project.



Project manager → *Project* → *Header*

Configuration: Project Header

Project name:

Project manager:

Project no:

Project orderer:

Project order no:

Project password:

Project size: 134360K    Version: 06/16/2000 13:39:15

Project comment:

ti010us.bmp

The texts entered into the upper section of the mask are allocated to the drawing footer or header. See also **chapter Project Management, General information on the project.**

### G 2.2.1 Editing the drawing header



→ Project manager → *Project* → *Header* → *Edit drawing header*

t011.us.bmp

OK	Exit header entries, entries are saved.
CANCEL	Exit header entries, entries are rejected.
RESET	All entries are accepted by means of the defaults from the file FRAMES.INI.
MERGE	All variables are accepted with the defaults from the file FRAMES.INI, i.e. all fixed texts are preserved.
CLEAR	All field contents are deleted.
EXPORT	The current entries of the header are written into the file FRAMES.INI and are hence the new defaults for RESET and MERGE

By using CLEAR AND then EXPORT you will lose all entries!

SELECT	From the list field of the FRAMES.INI file a section can be selected. Its contents or default settings are then activated, when RESET or MERGE is pressed.
<i>Preview</i>	A preview showing the contents as they are printed is superimposed. The appearance of inserted bitmaps can thus be evaluated.
<i>Titles</i>	The titles defined by the user are superimposed, giving a brief description of the field or of its intended contents. The field designation (e.g. F52) is entered into the fields in which the user has not entered a header title designation.

### G 2.2.2 Editing the header title



Project manager → *Project* → *Header* → *Edit header title*

			...
...			
	name		...
	comment		
...	start		OBID
	end		

ti012us.bmp

Title designations, which can later be superimposed on the header inscription mask, can be entered into this mask.

## G 2.2.3 Editing the drawing footer



Project manager → Project → Header → Edit drawing footer

t013us.bmp

- OK Exit footer entries, entries are saved.
- CANCEL Exit footer entries, entries are rejected.
- RESET All entries are accepted by means of the defaults from the file FRAMES.INI.
- MERGE All variables are accepted with the defaults from the file FRAMES.INI, i.e.. all fixed texts are preserved.
- CLEAR All field contents are deleted.
- Export The current entries of the footer are written into the file FRAMES.INI and are hence the new defaults for Reset and Merge

By using Clear and then Export you will lose all entries!

- SELECT A section from the file FRAMES.INI can be selected from the list field. Its content or pre-settings can then be activated when RESET or MERGE are used.

*Preview* A preview showing the contents as they are printed is superimposed. The appearance of inserted bitmaps can thus be evaluated.

*Titles* The titles defined by the user are superimposed, giving a brief description of the field or of its intended contents. The field designation (e.g. F24) is entered into the fields in which the user has not entered a footer designation.

### G 2.2.4 Editing the footer title



Project manager → *Project* → *Header* → *Edit footer title*

Sta1	Revision1	Date1	Norm1		Date0
Sta2	Revision2	Date2	Norm2		Resp
Sta3	Revision3	Date3	Norm3		Check
					Norm0

...						
	Origin		Cref			Creb

...	Title1	
...	Title2	...

			Function
Custom.D.No			Localisation
...	Localisation	DocId	
	Proj. Nr.		DocKz Pg

ti014us.bmp

Title designations, which can later be superimposed on the footer inscription mask, are entered into this mask.

### G 2.3 Program-specific drawing header / footer

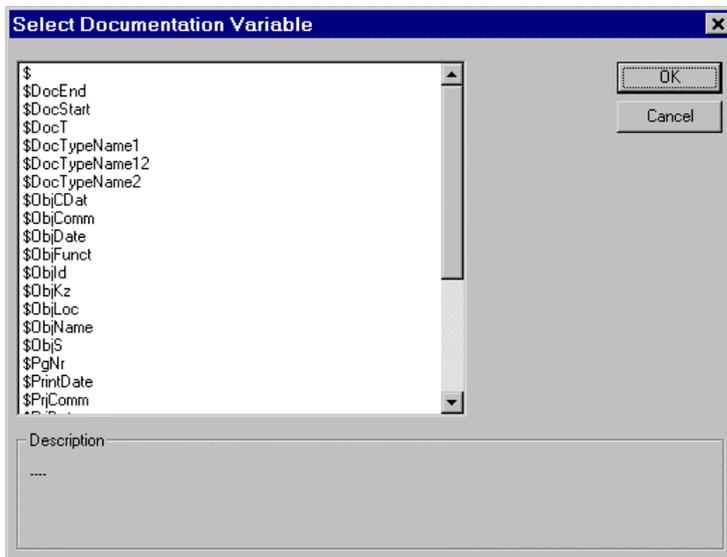
The project-specific frame inscription is valid initially for all sheets printed out from DigiTool. Each program features a menu item *Program, Header*. Both the drawing header and drawing footer can be defined under this menu item. However, each modification effected on this level is valid only for this one program (object).

An entry deviating from the default from the file FRAMES.INI is recognized by the fact the yellow field colour changes to green.

Hence, the fields **Date** and **Name** are program-specific in the following example.

### G 2.4 Automatic allocation of object parameters

To change entries, the cursor must be placed on a field and the required entry made. In the case of the fields for the header and footer inscription, a list of the frame inscription variables available in DigiTool (Documentation variables) can be called additionally with key **F2**. On selecting a variable, a corresponding explanatory text appears in the lower section. For a list of the variables see [page G-56.Variables for drawing footer/ header inscriptions](#)



ti015us.bmp

## G 2.5 Bitmaps in the drawing footer

It is possible to implement bitmaps in the drawing footer, header and in the cover page. The bitmaps are used from the directory c:\freelance\bitmaps.

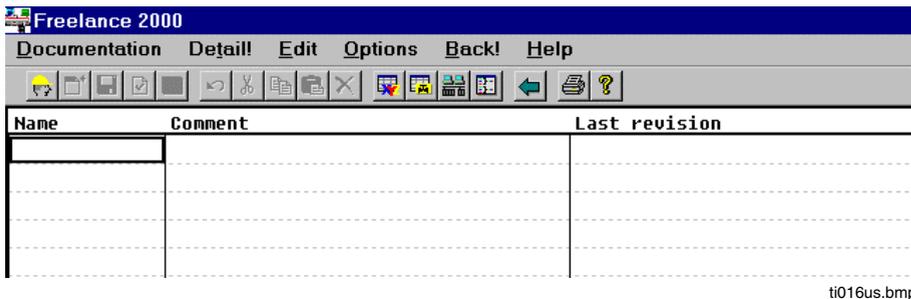
See also **page G-57, Fonts, national languages and bitmaps in the drawing footer / header.**

 Any other bitmaps can be implemented with #bitmap name, if they are saved in the c:\freelance\bitmaps directory.

## G 3 Define the Documentation Job

### G 3.1 Creating a new print job

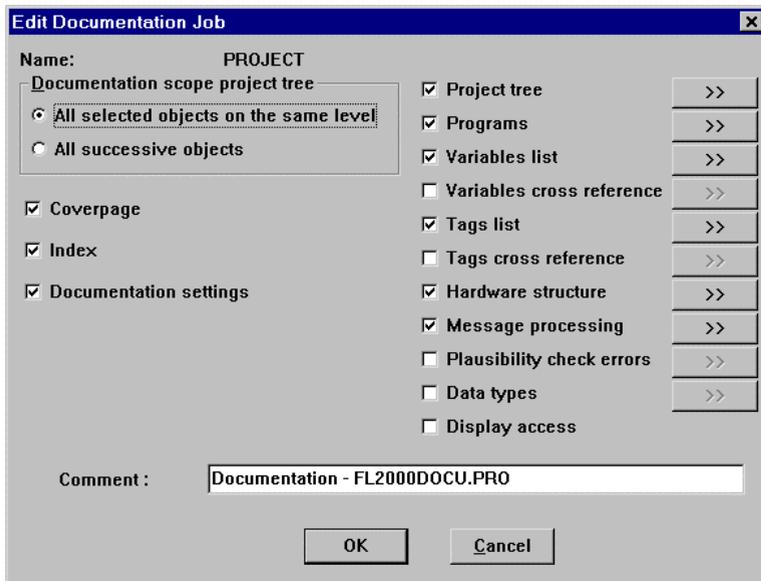
The easiest way to create a new print job is to select the field NAME with a double-click of the mouse and enter an unequivocal name. Then an explanatory text can be entered into the field COMMENT: In the field LAST REVISION, the date and time are entered automatically by the system. Now the menu item *Detail* is selectable and the documentation scope required under this job is selected.



### G 3.2 Detail of the print job



→ Project → Documentation → Documentation → Detail





G 3.2.3 Index

If this field is selected, a table of contents is output before the documentation job, featuring a list of all documents printed out, acc. to the specified order. See page G-36, **Sorting the output.**

DocId	DocNs	OBID	Title1	Title2	Function	Custom.D.No	Date0	Check	Pg.n
EA110		COV	COVERPAGE			123456789012			1
EA112		MAC	SETTINGS OF THE	DOCUMENTATION					2
EA113		MAN	PROJECT TREE	GRAPHICS					3
EA110		TAG	TAG LIST						24
EF110		MSG	MESSAGE						1
EF112		DSP	DISPLAY ACCESS						16
EF110		ERR	CHECK LIST	ERRORS					12
EF110		VAR	VARIABLE LIST						104
EF115		SVAR	STRUCT.VAR						1
EA110		IND	INDEX			123456789012			1

ABB	Freelance	Name: f120000docu	Start: /	IND
		Comment: Demo Project Freelance 2000	End: /	
Customer	Product Management	ABB	INDEX	Custom D. No. 123456789012
Doc. 7.	Proj. No. 123456	Doc. 7.	0001	

ti018us.eps

**G 3.2.4 Documentation settings**

If this field is selected, a sheet is output featuring the settings selected for the print job.

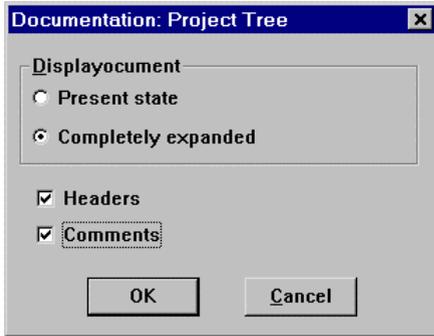
<b>ABB</b>											
Name: f120006docu		Start: C01/R02/T01/P01/B05								MAC	
Comment: Demo Project Freelance 2000		End: C01/R02/T01/P01/B05									
<b>Formular</b>		<b>Contents</b>		<b>S</b>		<b>Sort order:</b>		<b>Wildcard</b>			
General		Coverage Index All levels		X X X							
Project tree		Full expanded Comments Headers		X							
Programs		Contents CR Parametrisation data Comments						*			
Variable list		List CR List Name Comment Data type Station name Export Group type Process image Initial value OPC Address		X X X X X X X X X X X		Name				*	
Copyright 2004 ABB										Freelance 2000/02/04/04 10:11:17/14	
Date		Customer		<b>ABB</b>		Title		Custom D. No.		=	
Check		Product Management				SETTINGS OF THE		1244478012		P	
Check						DOCUMENTATION		= f120006docu		#	
Doc. Revision		Name		Origin		Doc. No.		Prog. No. 124456		# 52412 Doc. T. 0001	

ti019us.eps

### G 3.2.5 Project tree



→ Project → Documentation → Detail → Project tree → >>



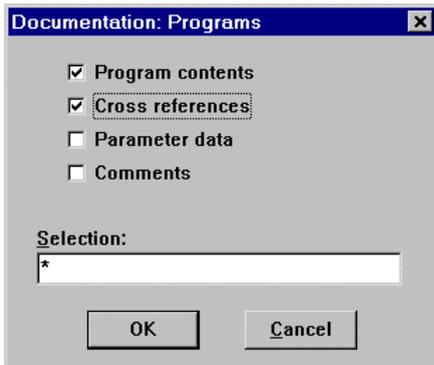
ti020us.bmp

Documentation of the project tree can be effected in the present state (e.g. parts blanked out) or as a whole. Furthermore, if required, the comments or header information can also be output for the project tree elements to be documented.

### G 3.2.6 Program documentation



→ Project → Documentation → Detail → Select Programs → >>



ti021us.bmp

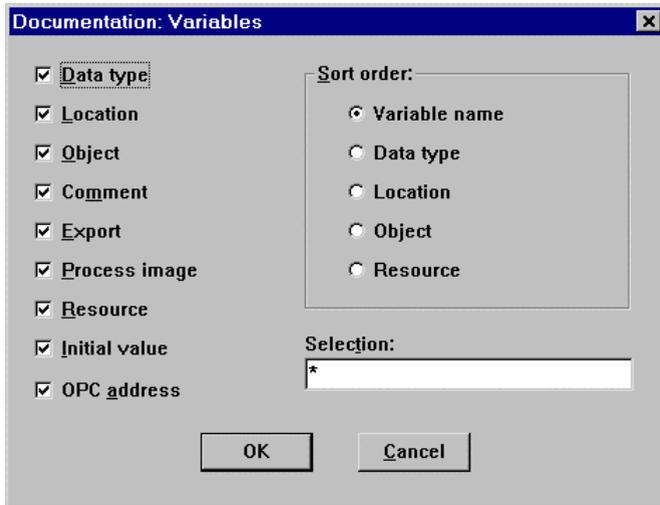
The programs selected in the project tree are documented with their contents. A wide variety of sheet types can be selected in a documentation job:



### G 3.2.7 Variables list



→ Project → Documentation → Detail → Variables list → >>



ti023us.bmp

Specify which parts of the variables lists are to be printed.

Select the required fields:

All the selected variables are printed as a list, featuring supplementary information such as comment, type, resource etc.

It is also possible to define how the list is **Sorted**.

It can be sorted acc. to *Variable name, Data type, Slot, Module type or Resource*

The list size can be limited by a selection with wildcards.

ABB		Freelance	
Name:	f12000docu	Start:	C01/R02/T01/P01/B05
Comment:	Demo Project Freelance 2000	End:	C01/R02/T01/P01/B05
		VAR	
Name	Comment or component name	Type	Res. X Module Location
FI701	X-Wert FIC701	REAL	PS01 N
FI702	X-Wert FIC702	REAL	PS01 N
FIC702SMA	Status Hand/Auto	BOOL	PS01 N
FY701	Y-Wert FIC701	REAL	PS01 N
FY702	Y-Wert FIC702	REAL	PS01 N
W700_var	W für FIC701, FIC702	REAL	PS01 N
auto_2	Pulssignal für BA:Automatik	BOOL	PS01 N
hand	Pulssignal für BA:Hand	BOOL	PS01 N
track_zu	Track: Zufußregler	BOOL	PS01 N

D-PS=Resource	X=Expert	P=Process image
Date	Customer	Title
Prog. PS.1	Product Management	ABB
Check		VARIABLE LIST
Rev. Name	Rev. Name	Origin
Prog. No. 121450	Doc. No. 10001	Doc. T.

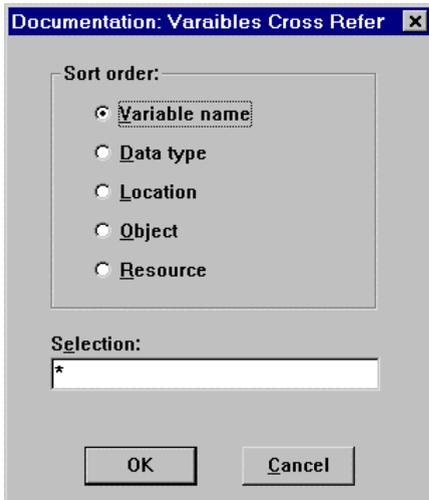
ti024us.eps

Name	Variable name
Comment or component name	Comment of the variable or component name of the structured variable
Type	Data type as REAL, BOOL or WORD
D-PS	Allocation of resource
X	Enable variable for reading other resources
Module	Module types (e.g. DD01)
Slot	Slot designations of the module, e.g. PS_1_0_2
	PS_1            Resource name
	0                Unit (rack)
	2                Slot
P	The process variable is processed via the process image or directly
Initial value	The default value set at cold start of the process station
See also <b>chapter Hardware structure</b> and <b>Engineering Manual, IEC 61131-3 Programming, Variables.</b>	

### G 3.2.8 Variables crossreference list



→ Project → Documentation → Detail → Variables crossreference → >>



ti025us.bmp

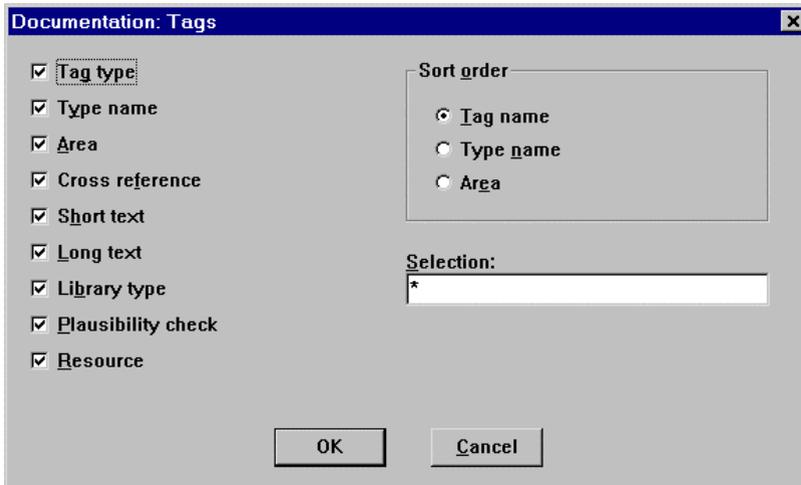
A crossreference list of the variables used in programs and displays can be output. The same selection window is displayed as for the variables list. See also **page G-20, Variables list**.



## G 3.2.9 Tags list



→ Project → Documentation → Detail → Tags list → >>



ti027us.bmp

Specify which parts of the tag list are to be printed.  
Select the required fields.

All the selected modules are printed as a list, featuring supplementary information such as comment, type name, resource etc.

It is also possible to define how the list is **sorted**. It can be sorted acc. to *Tag name*, *Type name*, *Area*.

The list size can be limited by a selection with wildcards.

ABB										Freelance	
Name: f12000docu				Start: C01/R02/T01/P01/B05		TAG					
Comment: Demo Project Freelance 2000				End: C01/R02/T01/P01/B05							
Name	T	A	R	Short text	Long text	Type name	L	P	Res.		
FIC701	S	A	+	Stoff A	Einsatzstoff A	C_CU	S	@	PS01		
FIC702	S	A	+	Stoff B	Verh.-Regler Stoff B 20% von A	C_CR	S	@	PS01		
<small>           Ack=Ack / A=H P=Check not coded? not CR / -&gt;CR exists, no edit req. L=Library type(S=Standard/ E=Extra (SFC)/ U=User function block) P=Check( @ =FB is checked/ # =FB is not checked) D-PS=Resource         </small>											
Date		Customer		Title		Custom D. No.					
12/04/2000		Product Management		ABB		f12000docu					
Revision		Name		Origin		Prog.No.		Doc. T			
		Name		Cw./		121450		4 1001.0			
		Name		Cw./				0001			

ti028us.eps

Name	Tag name, max.12/16 characters
A	Plant area (A-O)
R	Processing state + Block being processed (processing). - Block not being processed (processing). ? Block undefined (processing) Sequence chains and I/O blocks are always shown with?"
Short/long text	max. 12/30 characters
Type name	Brief designations of the function block type
L	S = Standard tag library U = User defined functionblock library E = Extra Library (SFC)
P	# Block not checked for plausibility @ Block checked for plausibility
D-PS	Allocated resource

### G 3.2.10 Tags crossreference



→ Project → Documentation → Detail → Tags crossreference → >>



ti029us.bmp

The dialog box is the same as for the tags list. See also **page G 3.2.9, Tags list**.

							
	Name: f12000docu		Start: C01/R02/T01/P01/B05		CR T		
Comment: Demo Project Freelance 2000		End: C01/R02/T01/P01/B05					
Tag name	Short text	Long text	Type name	Res. R	DocId	DocKz	Type
FIC701	Stoff A	Einsatzstoff A	C_CU	PS01 +	EFT20 EFL50 EFL73 EFL20 EFL50 EFL50 EFL50 EFL50	ZUFLUSS D_REAKTOR_1 D_RT04 D_GRB1 BMP_Kessel BMP_Zement BMP_Brauerei BMP_Petro	FBD FGR TR_D GRU FGR FGR FGR FGR
FIC702	Stoff B	Verh.-Regler Stoff B 20% von A C_CR		PS01 +	EFT20 EFL50 EFL73 EFL20 EFL50 EFL50 EFL50	ZUFLUSS D_REAKTOR_1 D_RT04 D_GRB1 BMP_Kessel BMP_Zement BMP_Brauerei BMP_Petro	FBD FGR TR_D GRU FGR FGR FGR FGR
<small>R:Cross ref. code? =no CR / + =CR exists / - =CR exists, no addr req / D:PS-Resource</small>							
				Title: CR TAGS TAG LIST		Custom D. No.: f12000docu Prog.Nr.: 121450	
Rev	Revision	Name	Rev	Origin	Chg./	Chg.D.	Doc. F.

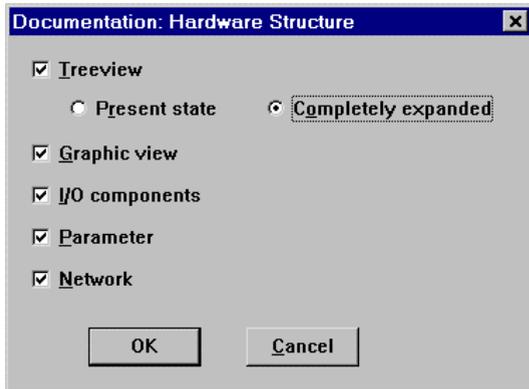
ti030us.eps

- Tag name                      Name of the function block, max. 12/16 characters
- Short/long text              max. 12/30 characters
- Type name                      Brief designations of the function block type
- D-PS                              Allocated resource
- R                                      Cross reference code, ? = no CR, + = CR exists
- Dokld                              Document type
- DokKz                              Document identification
- Typ                                      Program or display type (IL, LD, FBD, FGR)

### G 3.2.11 Hardware structure



→ *Project* → *Documentation* → *Detail* → *Select Hardware structure* → >>



ti031us.bmp

Specify which parts of the hardware structure are to be printed. Select the required fields.

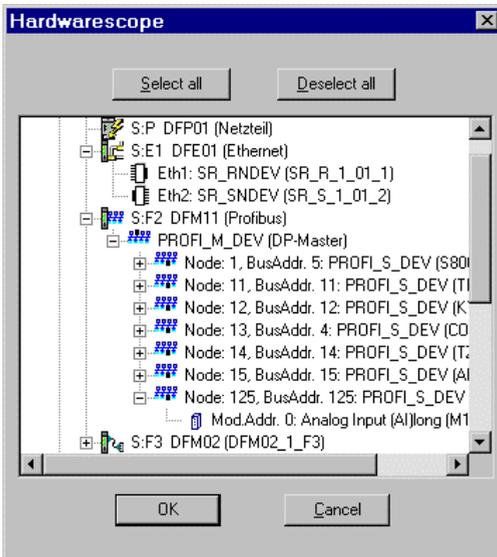


For selective documentation of the menu item **I/O-components and parameter** → *Project* → *Documentation* → *Options* → *Hardware scope*. See **page G-38, Selecting the Hardwarescope**



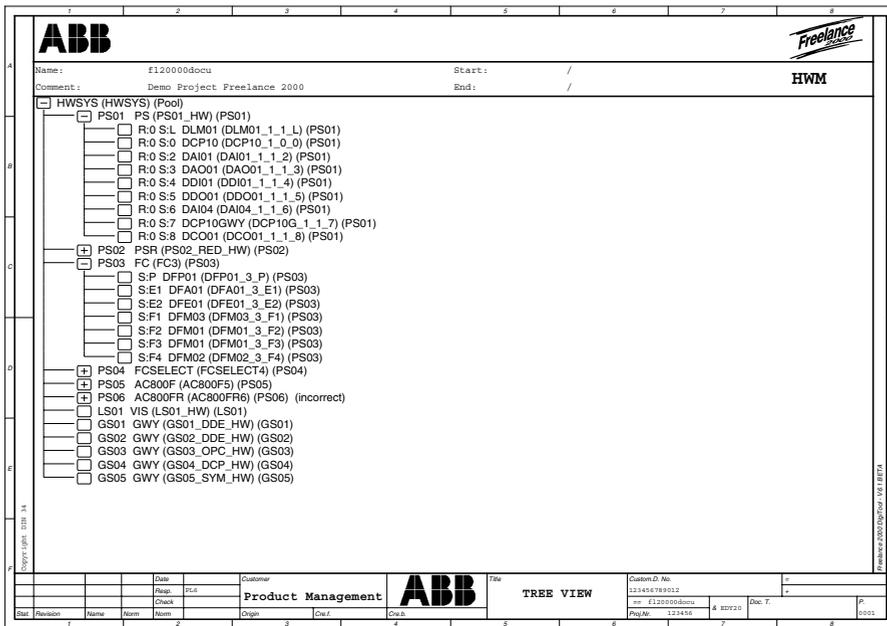
The menu items, I/O components and Parameters, should only be selected if an I/O component selection has first been made, which is useful for the project as described above. Otherwise, an expression for the I/O components and the parameters is created for each hardware component entered in the project, including all configured Profibus slaves and their modules!

If **modular Profibus slaves** are configured in the project, it is useful for a clearly viewable and compact display of the I/O components and parameters (user and DPV1 parameters), to mark only the modular Profibus slave without the associated modules, when setting up a print job for the hardware structure in the selection window under *Project* → *Documentation* → *Options* → *Hardware Section*. Here, the hardware tree must be opened under the slave which is to be documented. Otherwise, all the modules beneath it are also marked. Of course, single modules for the hardware documentation can also be selected.



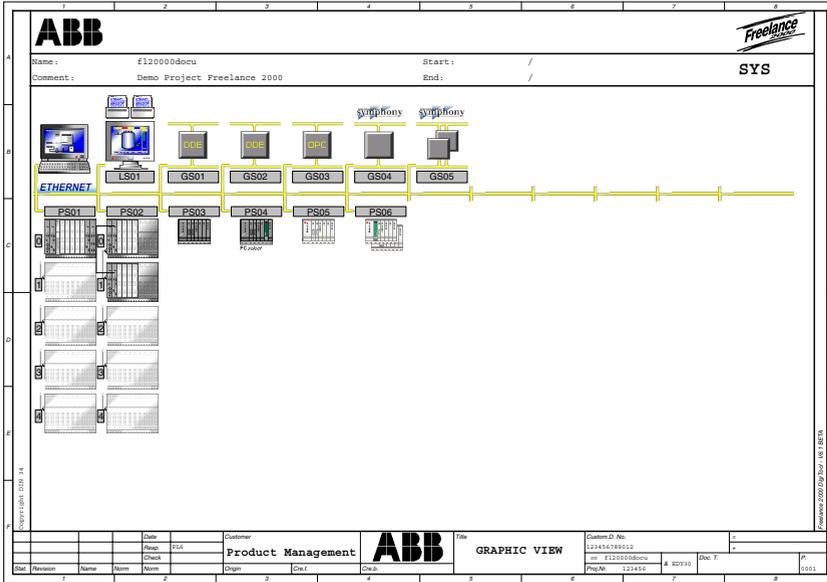
ti064us.bmp

Tree view

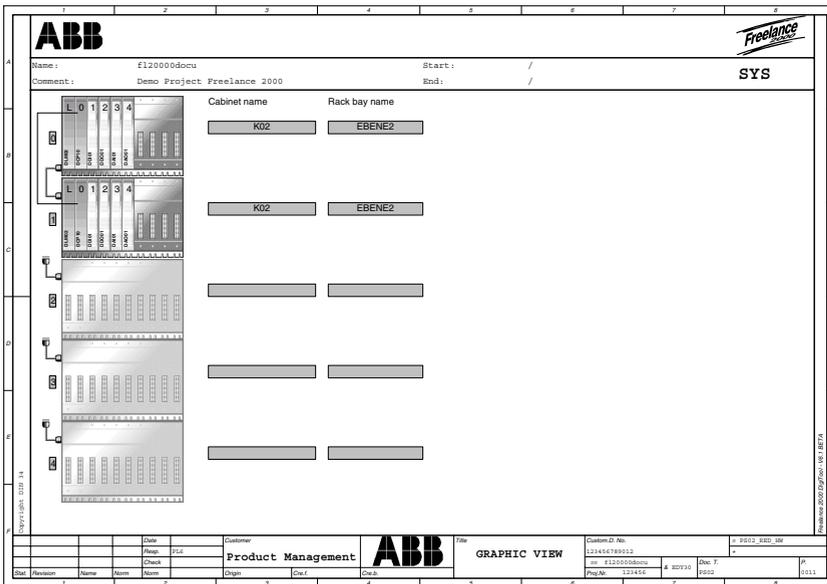


ti032us.eps

Graphic view



ti033us.eps



ti034us.eps

I/O Components

Input Component		Data type	Byte	Bit	Length	Variable	Comment
CH0	UINT	0	0	16	ANALOG01		
CH1	UINT	1	0	16	ANALOG02		
CH2	UINT	2	0	16	ANALOG03		
CH3	UINT	3	0	16	ANALOG04		
CH4	UINT	4	0	16	ANALOG05		
CH5	UINT	5	0	16	ANALOG06		
CH6	UINT	6	0	16	ANALOG07		
CH7	UINT	7	0	16	ANALOG08		
CH8	UINT	8	0	16	ANALOG09		
CH9	UINT	9	0	16	ANALOG10		
CH10	UINT	10	0	16	ANALOG11		
CH11	UINT	11	0	16	ANALOG12		
CH12	UINT	12	0	16	ANALOG13		
CH13	UINT	13	0	16	ANALOG14		
CH14	UINT	14	0	16	ANALOG15		
CH15	UINT	15	0	16	ANALOG16		

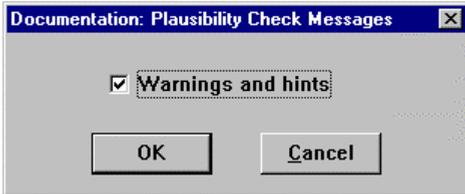
Diagnostics Component		Data type	Byte	Bit	Length	Variable	Comment
WDR	BOOL	0	0	8			
TMR	BOOL	1	0	8			
ES11	BOOL	2	0	8			
ES12	BOOL	3	0	8			
ES13	BOOL	4	0	8			
ES14	BOOL	5	0	8			
ES15	BOOL	6	0	8			
ES16	BOOL	7	0	8			
ES17	BOOL	8	0	8			
ES18	BOOL	9	0	8			
ES19	BOOL	10	0	8			
ES20	BOOL	11	0	8			
ES21	BOOL	12	0	8			
ES22	BOOL	13	0	8			
ES23	BOOL	14	0	8			
ES24	BOOL	15	0	8			
ES25	BOOL	16	0	8			
ES26	BOOL	17	0	8			
ES27	BOOL	18	0	8			
ES28	BOOL	19	0	8			
ES29	BOOL	20	0	8			
ES30	BOOL	21	0	8			
ES31	BOOL	22	0	8			
ES32	BOOL	23	0	8			
ES33	BOOL	24	0	8			
ES34	BOOL	25	0	8			
ES35	BOOL	26	0	8			
ES36	BOOL	27	0	8			
ES37	BOOL	28	0	8			
ES38	BOOL	29	0	8			
ES39	BOOL	30	0	8			
ES40	BOOL	31	0	8			
ES41	BOOL	32	0	8			
ES42	BOOL	33	0	8			
ES43	BOOL	34	0	8			
ES44	BOOL	35	0	8			
ES45	BOOL	36	0	8			
ES46	BOOL	37	0	8			
ES47	BOOL	38	0	8			
ES48	BOOL	39	0	8			
ES49	BOOL	40	0	8			
ES50	BOOL	41	0	8			
ES51	BOOL	42	0	8			
ES52	BOOL	43	0	8			
ES53	BOOL	44	0	8			
ES54	BOOL	45	0	8			
ES55	BOOL	46	0	8			
ES56	BOOL	47	0	8			
ES57	BOOL	48	0	8			
ES58	BOOL	49	0	8			
ES59	BOOL	50	0	8			
ES60	BOOL	51	0	8			
ES61	BOOL	52	0	8			
ES62	BOOL	53	0	8			
ES63	BOOL	54	0	8			
ES64	BOOL	55	0	8			
ES65	BOOL	56	0	8			
ES66	BOOL	57	0	8			
ES67	BOOL	58	0	8			
ES68	BOOL	59	0	8			
ES69	BOOL	60	0	8			
ES70	BOOL	61	0	8			
ES71	BOOL	62	0	8			
ES72	BOOL	63	0	8			
ES73	BOOL	64	0	8			
ES74	BOOL	65	0	8			
ES75	BOOL	66	0	8			
ES76	BOOL	67	0	8			
ES77	BOOL	68	0	8			
ES78	BOOL	69	0	8			
ES79	BOOL	70	0	8			
ES80	BOOL	71	0	8			
ES81	BOOL	72	0	8			
ES82	BOOL	73	0	8			
ES83	BOOL	74	0	8			
ES84	BOOL	75	0	8			
ES85	BOOL	76	0	8			
ES86	BOOL	77	0	8			
ES87	BOOL	78	0	8			
ES88	BOOL	79	0	8			
ES89	BOOL	80	0	8			
ES90	BOOL	81	0	8			
ES91	BOOL	82	0	8			
ES92	BOOL	83	0	8			
ES93	BOOL	84	0	8			
ES94	BOOL	85	0	8			
ES95	BOOL	86	0	8			
ES96	BOOL	87	0	8			
ES97	BOOL	88	0	8			
ES98	BOOL	89	0	8			
ES99	BOOL	90	0	8			
ES100	BOOL	91	0	8			
ES101	BOOL	92	0	8			
ES102	BOOL	93	0	8			
ES103	BOOL	94	0	8			
ES104	BOOL	95	0	8			
ES105	BOOL	96	0	8			
ES106	BOOL	97	0	8			
ES107	BOOL	98	0	8			
ES108	BOOL	99	0	8			
ES109	BOOL	100	0	8			
ES110	BOOL	101	0	8			
ES111	BOOL	102	0	8			
ES112	BOOL	103	0	8			
ES113	BOOL	104	0	8			
ES114	BOOL	105	0	8			
ES115	BOOL	106	0	8			
ES116	BOOL	107	0	8			
ES117	BOOL	108	0	8			
ES118	BOOL	109	0	8			
ES119	BOOL	110	0	8			
ES120	BOOL	111	0	8			
ES121	BOOL	112	0	8			
ES122	BOOL	113	0	8			
ES123	BOOL	114	0	8			
ES124	BOOL	115	0	8			
ES125	BOOL	116	0	8			
ES126	BOOL	117	0	8			
ES127	BOOL	118	0	8			
ES128	BOOL	119	0	8			
ES129	BOOL	120	0	8			
ES130	BOOL	121	0	8			
ES131	BOOL	122	0	8			
ES132	BOOL	123	0	8			
ES133	BOOL	124	0	8			
ES134	BOOL	125	0	8			
ES135	BOOL	126	0	8			
ES136	BOOL	127	0	8			
ES137	BOOL	128	0	8			
ES138	BOOL	129	0	8			
ES139	BOOL	130	0	8			
ES140	BOOL	131	0	8			
ES141	BOOL	132	0	8			
ES142	BOOL	133	0	8			
ES143	BOOL	134	0	8			
ES144	BOOL	135	0	8			
ES145	BOOL	136	0	8			
ES146	BOOL	137	0	8			
ES147	BOOL	138	0	8			
ES148	BOOL	139	0	8			
ES149	BOOL	140	0	8			
ES150	BOOL	141	0	8			
ES151	BOOL	142	0	8			
ES152	BOOL	143	0	8			
ES153	BOOL	144	0	8			
ES154	BOOL	145	0	8			
ES155	BOOL	146	0	8			
ES156	BOOL	147	0	8			
ES157	BOOL	148	0	8			
ES158	BOOL	149	0	8			
ES159	BOOL	150	0	8			
ES160	BOOL	151	0	8			
ES161	BOOL	152	0	8			
ES162	BOOL	153	0	8			
ES163	BOOL	154	0	8			
ES164	BOOL	155	0	8			
ES165	BOOL	156	0	8			
ES166	BOOL	157	0	8			
ES167	BOOL	158	0	8			
ES168	BOOL	159	0	8			
ES169	BOOL	160	0	8			
ES170	BOOL	161	0	8			
ES171	BOOL	162	0	8			
ES172	BOOL	163	0	8			
ES173	BOOL	164	0	8			
ES174	BOOL	165	0	8			
ES175	BOOL	166	0	8			
ES176	BOOL	167	0	8			
ES177	BOOL	168	0	8			
ES178	BOOL	169	0	8			
ES179	BOOL	170	0	8			
ES180	BOOL	171	0	8			
ES181	BOOL	172	0	8			
ES182	BOOL	173	0	8			
ES183	BOOL	174	0	8			
ES184	BOOL	175	0	8			
ES185	BOOL	176	0	8			
ES186	BOOL	177	0	8			
ES187	BOOL	178	0	8			
ES188	BOOL	179	0	8			
ES189	BOOL	180	0	8			
ES190	BOOL	181	0	8			
ES191	BOOL	182	0	8			
ES192	BOOL	183	0	8			
ES193	BOOL	184	0	8			
ES194	BOOL	185	0	8			
ES195	BOOL	186	0	8			
ES196	BOOL	187	0	8			
ES197	BOOL	188	0	8			
ES198	BOOL	189	0	8			
ES199	BOOL	190	0	8			
ES200	BOOL	191	0	8			
ES201	BOOL	192	0	8			
ES202	BOOL	193	0	8			
ES203	BOOL	194	0	8			
ES204	BOOL	195	0	8			
ES205	BOOL	196	0	8			
ES206	BOOL	197	0	8			
ES207	BOOL</						



### G 3.2.13 Plausibility check errors



→ *Project* → *Documentation* → *Detail* → *Plausibility check errors* → >>



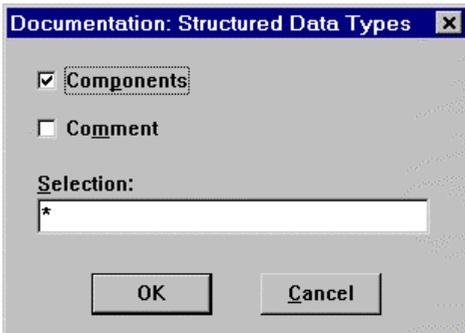
ti039us.bmp

The plausibility check errors can be output. If the field *Warnings* is selected, messages are output additionally.

### G 3.2.14 Structured variables



→ *Project* → *Documentation* → *Detail* → *Select Structured Variables* → >>



ti040us.bmp

By selecting, the components and comments of the user-specific structured variables can also be outputed.

G 3.2.15 Display access



→ Project → Documentation → Detail → Display access → >>

The display allocations are documented as tables.

ABB		Freelance	
Name: f12000docu		Start: /	
Comment: Demo Project Freelance 2000		End: /	
Tag name		Display name	Default
L1120_G1	Trend_10	N	
	AK11	N	
L3020_G1	Trend_11	N	
	R30	N	
L3120_G1	Trend_30	N	
	R31	N	
L6001_G1	Trend_31	N	
	T60	N	
L7001_G1	T70	N	
L8001_G1	T80	N	
L9001_G1	T90	N	
LI3005	REAKTOR_3_G1	N	
	REAKTOR_3	N	
	REAKTOR_3_KB	N	
	REAKTOR_3_AS	N	
LI4005	REAKTOR_4_G1	N	
	REAKTOR_4	N	
	REAKTOR_4_TR	N	
	REAKTOR_4_SF	N	
LI700	D_GRB3	N	
	D_REA_TANK	N	
LI7005	REAKTOR_7_G1	N	
	REAKTOR_7	N	
	REAKTOR_7_TR	N	
	REAKTOR_7_AS	N	
LI720	D_R704	N	
LI720_1	D_R704	N	
LI750	D_FUELLEN_R1	N	
	D_Rez_Reakt	N	
LI752	D_Rez_Reakt	N	
LI753	D_Rez_Reakt	N	
LIC704	D_GRB1	N	
	BMP_Abwasser	N	
	D_R704	N	
NI704	D_GRB1	N	
	D_REAKTOR_1	N	
NP2000A_M1	B20	N	

Customer	Product Management	ABB	Title	DISPLAY ACCESS
Customer No.	123456789012		Customer No.	f12000docu
Doc. T.	123456	123456	Doc. T.	0008

ti041us.eps

- Tag name** Tag name of the allocated tag
- Display name** Display names of the displays which are allocated to the tag
- Default** YES if the display is defined as the default display

## G 4 Documentation Scope

### G 4.1 Hard disk requirement



If outputting several sheets (e.g. an entire project), please ensure that there is enough space available on the temporary disk. 3 sheets are always combined for one print job for the Windows print manager. This ensures that already at the time of creating the print job for DigiTool, the first data can be transferred to the printer. Generation of the print data and output to the printer are hence effected with a slight delay and the first temporary data can be deleted again.

### G 4.2 Viewing selected documentation scope



→ *Project* → *Documentation* → *Documentation* → *Preview*

A table of contents of the sheets that have been selected in the previously selected documentation job is displayed, e.g.

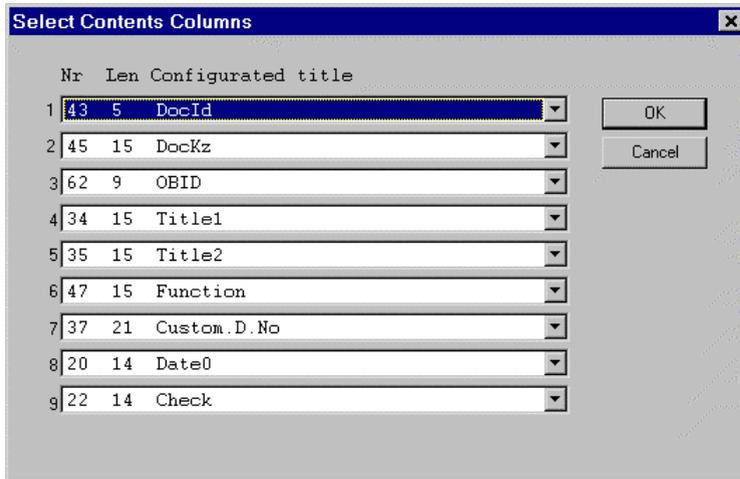
DocID	Doc ID	ObjID	Title1	Title2	Function	Custom D.No.	Date0	Check
EAA10		COV	COVERPAGE			123456789012	30.05.00	
EAB12		MAC	SETTINGS	DOCUMENTATION		123456789012	30.05.00	
EDY20		HWM	TREE VIEW	GRAPHICS		123456789012	30.05.00	
EEC10		TAG	TAG LIST			123456789012	30.05.00	
EFP15		CR_V	CROSS-REFERENCE	VARIABLE LIST		123456789012	30.05.00	
EFT20	ZUFLUSS	FBD	FBD PROGRAM	GRAPHICS	ZUFLUSS	123456789012	30.05.00	correct

### G 4.3 Sorting the output

Start in Window "Configuration: Documentation"



→ Project → Documentation → Options → Sort fields



ti043us.bmp

Sorting of the output can be defined. Documentation output can be sorted acc. to a maximum of 5 criteria (64, 20, 20, 12, 3 characters). The field contents of the drawing footer can be selected as sorting criteria. See also **page G-54, Field names in the drawing footer / header**. The standard setting of DigiTool is shown in the example.

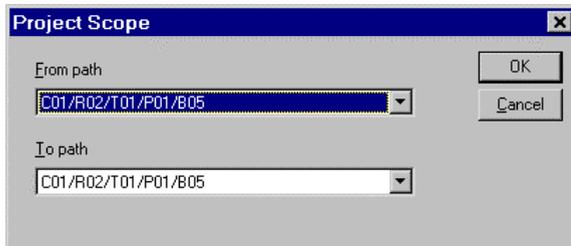
Each sorting criterion set here automatically becomes a column in the table of contents when printing out. Sorting to date isn't possible!

#### G 4.4 Selecting the project tree objects

Start in Window "Configuration: Documentation"



→ *Project* → *Documentation* → *Options* → *Project scope*



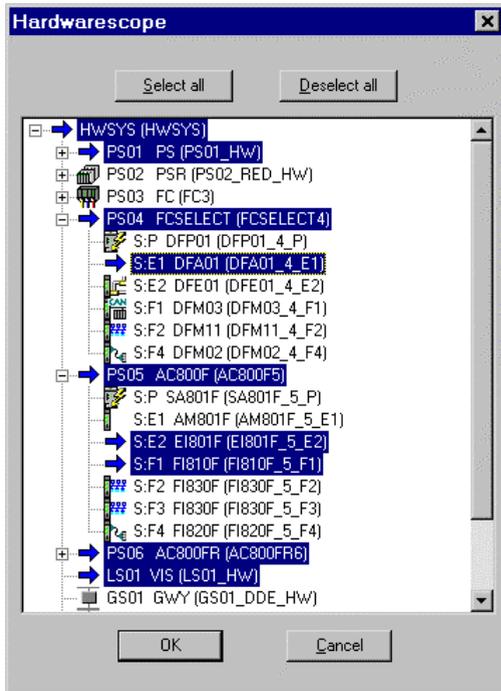
ti043us.bmp

In this mask a selection of the project tree objects **From - To** (only in the order of the project tree objects) can be made for the printout, if this has not already been selected itself for printout in the project tree.

## G 4.5 Selecting the Hardwarescope



→ *Project* → *Documentation* → *Options* → *Hardwarescope*



ti045us.bmp

In this mask, a selection can be made for the documentation of desired I/O components.



This selection is only active as long as you are in the "Documentation" dialogue. After leaving this dialogue, the completely configured hardware is selected by default! This point is only important in connection with the selection under:



→ *Project* → *Documentation* → *Details* → *Hardware structure*  
→ *I/O components* and → *Parameter*

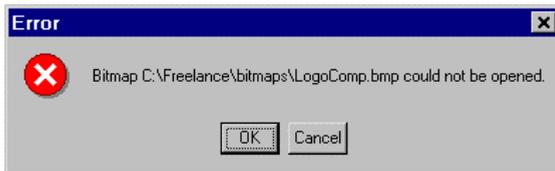
## G 4.6 Selecting notify print errors



→ *Project* → *Documentation* → *Options* → *Notify print errors*

At this point, you determine whether errors appearing in connection with the printing of the project documentation, such as printing errors, missing objects (e.g. bitmaps) and invalid or undefined field functions in drawing header/footer, are displayed on the screen.

Example of a possible error message:



ti058us.bmp

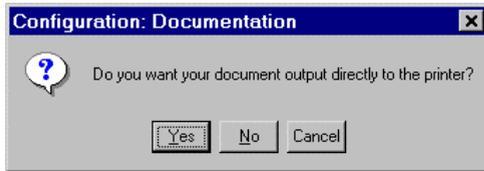
## G 5 Outputting Documentation

### G 5.1 Print

Start in Window "Configuration: Documentation"



→ Project → Documentation → Documentation → Print



ti047us.bmp

Start printing to a file or directly to the printer.

- |        |  |
|--------|--|
| YES    | The output is effected to the standard printer entered in Windows. Depending on the printer option, the print quality can still be adapted and the number of required copies entered. Printing can be aborted with Cancel. Using the Setup button, another printer type other than the standard printer can be selected if necessary, <b>see page G-42, Printer setup</b> . A message box is displayed as a further indication that the printer is set to DIN A4 landscape format. |
| NO     | It is possible to print to file. If this option is selected, a window opens and the path, file name of the target file can be edited here. Default setting for the file name: name of the print job with the extension ".DPD"; Default setting for the path: "\\Freelance\PROJ", <b>see page G-41, Print file</b> .  |
| CANCEL | Cancel printing  |



The maximum amount of hard disk space available must be kept in mind.

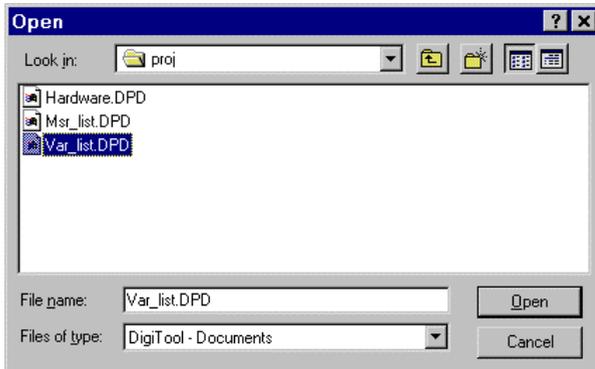
See **page G-35, Hard disk requirement**. The preview function should be used to get a survey of the print scope. See **page G-35, Viewing selected documentation scope**.

## G 5.2 Print file

Start in Window "Configuration: Documentation"

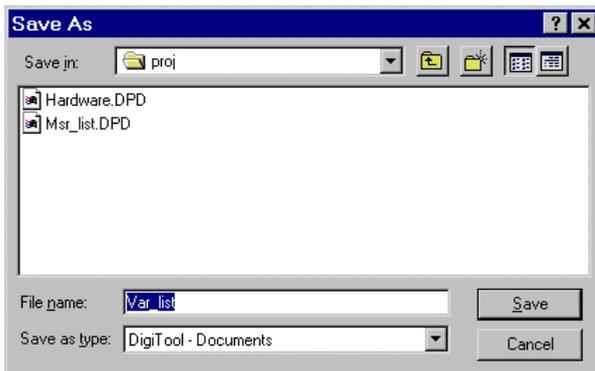


→ Project → Documentation → Documentation → Print file



ti048us.bmp

Print out a previously generated print file: The file to be printed is selected in a special Windows selection box. The files will be saved with the extension. DPD.



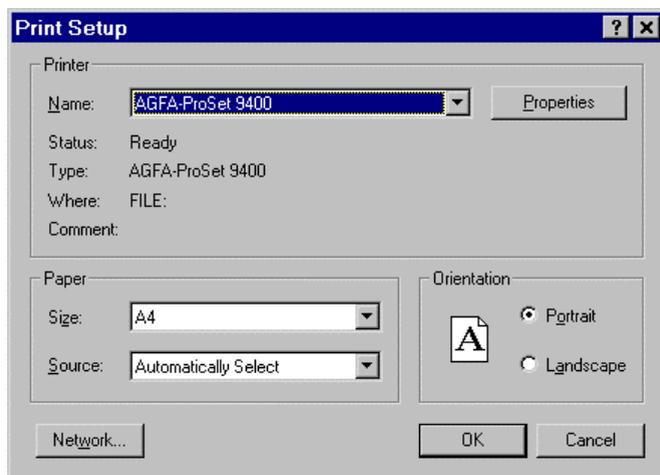
ti059us.bmp

### G 5.3 Printer setup

Start in Window "Configuration: Documentation"



→ Project → Documentation → Documentation → Printer setup



ti049us.bmp

Select the printer. If a printer other than the standard printer is to be used, **Special Printer** is selected and the associated button is clicked with the mouse pointer. Now the required printer can be selected. Under **Options** fine adjustments can be made for the selected printer, see Windows documentation for this purpose.



The selected paper format (DINA4, letter) is supported. Default: landscape format. Adjustments for the page layout can be changed under DigiTool Configure, page layout.

## G 6 Documentation Manager

### G 6.1 Editing the documentation manager

The documentation manager menu can be accessed in one of the following ways:



Select project tree → *Project* → *Document*  
Marked objects in the project tree are selected for documentation.



Out of Programs (IL, LD, FBD, SFC) or displays (FGR)  
→ *Program* → *Documentation*  
The current program or display is selected for documentation.

Name	Comment	Last revision
FL20000DCU	Documentaion Project	06/16/2000 13:12:29
HARDWARE	Hardware structure	06/16/2000 13:26:26
PROGRAMMS	Programms	06/16/2000 13:28:05
TAGS_LIST	Tages list	06/16/2000 13:27:30
UARI_LIST	Variables list	06/16/2000 13:28:40

ti050us.bmp

On entering the documentation manager, the print jobs available are shown beneath the menu line. The blank table lines represent the free print jobs.

The documentation jobs are structured in the following manner:

Name	job name, max. 12 characters, must begin with a letter, no special characters, no blanks.
Comment	max. 34 characters free text (special characters as well).
Last revision	the system automatically enters the date and time of the last change. Adaptation of the column width.

### G 6.2 Changing the column width

If the mouse pointer is placed on a column separation line, the mouse pointer changes to a horizontal arrow. With the left mouse button pressed, the column width can now be changed. This change is saved on quitting the document and is thus available for subsequent calls.

### G 6.3 Menu Structure Documentation

**Documentation**     *Insert new documentation job*  
*Print*  
*Print file*  
*Printer setup*  
*Comment*  
*Preview*  
*Export contents*  
*Exit*

**Detail!**

**Edit**                 *Field*  
*Delete Field*  
*Cut*  
*Copy*  
*Paste*  
*Delete*

**Options**             *Hardcopy*  
*Sort fields*  
*Project scope*  
*Hardware scope*  
*Notify print errors*

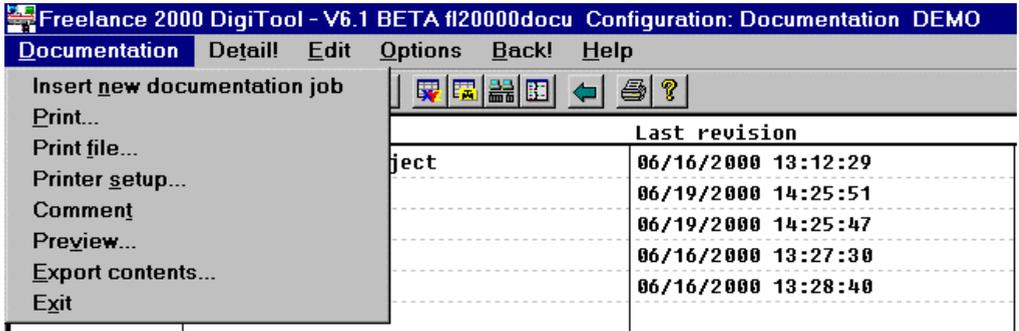
**Back!**

**Help**                 *Contents*  
*Overview*  
*Use help*  
*Info*

## G 6.4 Documentation



→ *Project* → *Documentation* → *Documentation*



ti051us.bmp

### G 6.4.1 Insert new documentation job



Place cursor on a blank line

→ *Documentation* → *Insert new documentation job*

→ Enter a documentation name into the field Name

### G 6.4.2 Copy new documentation job



Place cursor on an existing line  
→ *Documentation* → *Insert new documentation job*

The contents of the already selected documentation job are accepted for the new one. In the following mask the new name must be entered or be taken from the old name.

Insert Documentation Job

The names of documentation jobs must be unique.

Old: TAGS\_LIST

New: HARDWARE

OK Next Cancel

ti052us.bmp

A new documentation job can be created also by copying an old documentation job and saving it under a new name. All definitions concerning the documentation contents are also copied and can be modified later.

### G 6.4.3 Comment



*Project* → *Documentation* → *Documentation* → *Comment*

A long comment can be made here on the selected print job, describing the contents. For description see **chapter Project Management, Editing a project comment.**

### G 6.4.4 Export contents



*Project → Documentation → Documentation → Export contents*

The table of contents of the selected job is saved in a CSV format with file extension \*.dco. The file name and directory can be selected in a Windows menu, with the project directory being presented as a default and the first 8 characters of the documentation job name as file name. This file can be opened in table form with, for example, Excel, and worked on further.



ti063us.bmp

### G 6.4.5 Exit



*Project → Documentation → Documentation → Exit*

Exit the documentation menu, return to the project tree.

## G 7 Comment field editing

### G 7.1 Edit Comment Field



*Project → Documentation → Comment → Edit → Text input*

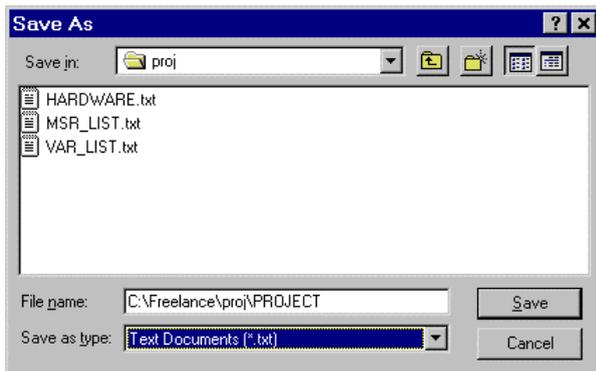
The text contents of the comment field can be changed.

### G 7.2 Export comment field



*→ Project → Documentation → Comment → Edit → Export*

The text content of the comment field is saved as a text file \*.txt, and serves as a basis for new print job comments. This text file is stored in unicode format.



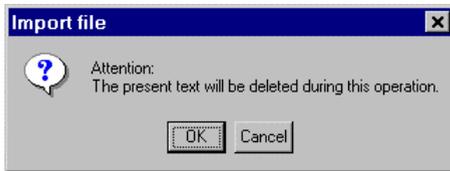
ti060us.bmp

### G 7.3 Import comment field

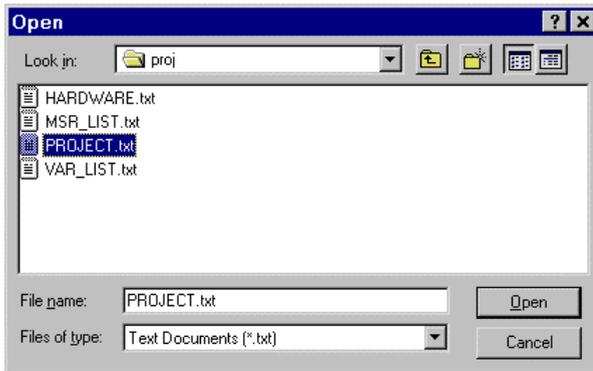


*→ Project → Documentation → Comment → Edit → Import*

The text content of an originally exported comment field (saved as a text file \*.txt) is inserted into the selected print job. Existing entries are completely overwritten. At this point, any text desired can be inserted as long as it is changed into unicode format.



ti061us.bmp



ti062us.bmp

## G 8 Tables

### G 8.1 Document types

Term		Type	Name
<b>Documentation-Description Documents</b>			
Title-/cover pages	EAA10	COV	Title page
Directories	EAB11	IND	Table of Contents
	EAB12	MAC	Setting the documentation job
<b>General Technical Information Documents</b>			
Gen.techn. doc.	EDY10	MAN	Project tree
	EDY12	OBJ	Head confi./resource/task/program list
	EDY19	OBJ	Comment
	EDY20	HWM	Tree view
	EDY22	NET	Network
	EDY30	SYS	Graphic view
	EDY33	MOD	Module parameters
<b>Technical Requirements and Dimension/Design Documents</b>			
Tag lists, block schema	EEC10	TAG	Tag list
	EEC11	CR_T	Cross-reference tag list
	EEC20	FB_FBD	User def. function blocks FBD
	EEC40	FB_IL	User def. function blocks IL
<b>Function-Description Documents</b>			
Signal descriptions	EFP10	VAR	Variables list
	EFP15	CR_V	Cross-reference variable list
	EFP16	SVAR	Structured variables
freely available	EFQ10	I/O	I/O components
freely available	EFL10	OV	Overview display
	EFL15	OV	Overview display cross references
	EFL19	OV	Overview display comment
	EFL20	GRU	Group display
	EFL25	GRU	Group display cross-reference
	EFL29	GRU	Group display comment
	EFL50	FGR	Graphic display (graphic)
	EFL53	FGR	Graphic display (parameters)
	EFL55	FGR	Cross-references graphic display
	EFL59	FGR	Comment graphic display
	EFL60	SFC_D	Sequential function chart

Term		Type	Name
	EFL63	SFC_D	Sequential function chart parameters
	EFL65	SFC_D	Sequential function chart cross-references
	EFL69	SFC_D	Sequential function chart comment
	EFL70	TR_D	Trend display
	EFL73	TR_D	Trend display chart parameters
	EFL75	TR_D	Trend display cross-references
	EFL79	TR_D	Trend display comment
	EFL80	MSG	Global message processing
	EFL81	MSG	Local message processing
	EFL82	DSP	Display allocation
	EFL90	PG	Programmer display
	EFL93	PG	Programmer display parameters
	EFL95	PG	Programmer display cross-references
	EFL99	PG	Programmer display comment
freely available	EFM10	ERR	Plausibility check error
freely available output	EFR10	SSL1	Signal sequence log automatic
	EFR13	SSL1	Signal sequence log parameter auto.
	EFR15	SSL1	Signal sequence log cross-ref. auto.
	EFR19	SSL1	Signal sequence log comment auto.
	EFR20	SSLN	Signal sequence log, manual
	EFR23	SSLN	Signal sequence log parameter manual
	EFR25	SSLN	Signal sequence log cross-ref. manual
	EFR29	SSLN	Signal sequence log comment manual
	EFR30	DCLD	Disturbance course log
	EFR33	DCLD	Disturbance course log parameter
	EFR35	DCLD	Disturbance course log cross-references
	EFR39	DCLD	Disturbance course log comment
	EFR40	OPL	Operational log
	EFR43	OPL	Operational log parameter
	EFR45	OPL	Operational log cross-references
	EFR49	OPL	Operational log comment

Term		Type	Name
SW-specific Documents	EFT20	FBD	FBD program (graphic)
	EFT23	FBD	FBD program (parameters)
	EFT25	FBD	FBD program (cross-references)
	EFT29	FBD	FBD program (comment)
	EFT40	IL	IL program (graphic)
	EFT43	IL	IL program (parameters)
	EFT45	IL	IL program (cross-references)
	EFT49	IL	IL program (comment)
	EFT50	SFC	SFC program (graphic)
	EFT53	SFC	SFC program (parameters)
	EFT55	SFC	SFC program (cross-references)
	EFT59	SFC	SFC program (comment)
	EFT60	LD	LD program (graphic)
	EFT63	LD	LD program (parameters)
	EFT65	LD	LD program (cross-references)
	EFT69	LD	LD program (comment)

## G 8.2 Description of the fields or Contents

All fields used in the documentation must be unequivocal, i.e. each field is given a serial number F1-F62. This is used e.g. for describing the field titles, if no other entry has been made. For a definition of the fonts used, see also **page G-57, Fonts, national languages and bitmaps in the drawing footer / header.**

### Drawing Footer:

F0	F4	F8	F12	F16	F20	...
F1	F5	F9	F13	F17	F21	
F2	F6	F10	F14	F18	F22	
F3	F7	F11	F15	F19	F23	

...	F24				F30		...
	F25						
	F26	F27	F28	F29	F31	F32	

...	F33			...
	F34			
	F35			

...	F36				F <sub>46</sub>	F47	
	F37				F <sub>48</sub>	F49	
	F38	F39		F <sub>42</sub>	F43	F44	
	F40		F41		F45		F50
					F45		F51

ti053us.bmp

### Drawing Header:

F52					...
-----	--	--	--	--	-----

...					F53
-----	--	--	--	--	-----

F54	F56			...
F55	F57			

...	F58		F60		F62
	F59		F61		

ti054us.bmp

### G 8.3 Field names in the drawing footer / header

The following table shows the correlation between:

- Field designation (**FIELD**),
- Default variable (**CONTENTS**)
- Default title (**TITLE**)
- Description or contents of the field acc. To DIN (**COMMENT**)

FIELD	CONTENTS	TITLE	COMMENT
F0		Sta1	State 1 of printout
F1		Sta2	State 2 of printout
F2		Sta3	State 3 of printout
F3	Sta		Fixed text in frame
F4	\$ObjDate	Revision1	Revision date1
F5		Revision2	Revision date2
F6		Revision3	Revision date3
F7	Revision		Fixed text in frame
F8		Date1	Date1 of revision
F9		Date2	Date1 of revision
F10		Date3	Date1 of revision
F11	Name		Fixed text in frame
F12		Norm1	
F13		Norm2	
F14		Norm3	
F15	Norm		Fixed text in frame
F16	Date		Fixed text in frame
F17	Resp		Fixed text in frame
F18	Check		Fixed text in frame
F19	Norm0		Fixed text in frame
F20	\$ObjCDat	Date0	Compilation date
F21	\$PrjMan	Resp	
F22	\$ObjS	Check	Checked by
F23		Norm0	
F24	Customer		Fixed text in frame
F25	#Logocust.bmp		Customer logo (bitmap) or text
F26	Origin		Fixed text in frame
F27		Origin	Original of
F28	Cre.f.		Fixed text in frame
F29		Cref	
F30	%LogoComp		Company logo (bitmap)
F31	Cre.b.		Fixed text in frame
F32		Creb	
F33	Title		Fixed text in frame
F34	\$DocTypeName1	Title1	Drawing name1
F35	\$DocTypeName2	Title2	Drawing name 2
F36	Custom.D.No.		Fixed text in frame

FIELD	CONTENTS	TITLE	COMMENT
F37	\$PrjOrdNr	Custom.D.No.	Customer drawing number
F38	==		Fixed text in frame
F39	\$PrjName	Localization	Installation site
F40	Proj.No.		Fixed text in frame
F41	\$PrjNr	Proj.No.	Project number
F42	&		Fixed text in frame
F43	\$DocT		Key to documentation type
F44	Doc.T.		Fixed text in frame
F45	\$ObjKz		Object designation
F46	=		Fixed text in frame
F47	\$ObjFunct		Function name of object
F48	+		Fixed text in frame
F49	\$ObjLoc		Location of object
F50	P.		Fixed text in frame
F51	\$PgNr		Sheet number
F52	%LogoLeft		Logo (bitmap) in upper left of header
F53	%LogoRight		Logo (bitmap) in upper right of header
F54	Name:		Fixed text in frame
F55	Comment::		Fixed text in frame
F56	\$ObjName		Name of object (Path in the project tree)
F57	\$ObjComm		Comment of object
F58	Start:		Fixed text in frame
F59	End:		Fixed text in frame
F60	\$DocStart		Start of selected print scope
F61	\$DocEnd		End of selected print scope
F62	\$ObjId		Type of editor

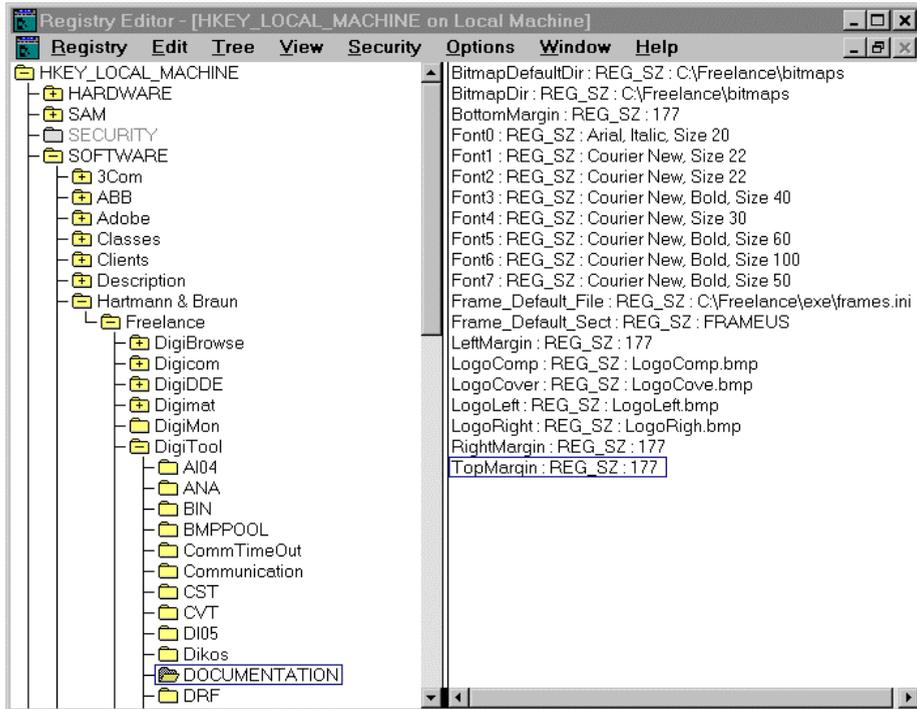
## G 8.4 Variables for drawing footer/ header inscriptions

\$	Avoids overriding through the project name
\$DocEnd	End of the selection range in the project tree (depends on the documentation job)
\$DocStart	Start of the selection range in the project tree (depends on the documentation job)
\$DocT	DIN number of the current documentation type (Hardcoded, language dependent).
\$DocTypeName1	First part of the name of the current documentation type. (Hardcoded, language dependent).
\$DocTypeName12	Name of the current documentation type. (Hardcoded, language dependent).
\$DocTypeName2	Second part of the name of the current documentation type. (Hardcoded, language dependent).
\$ObjCDat	Creation date of the current documentation object. (If available)
\$ObjComm	Short comment to the current documentation object. (If available)
\$ObjDate	Date of the current documentation object. (If available)
\$ObjFunct	Name of the current project tree object. (If available)
\$ObjId	Short Id to the current documentation object. (hardcoded, language dependent, typically 3 capital letters)
\$ObjKz	Documentation identifier of the current object.
\$ObjLoc	Localization of the current object.
\$ObjName	Name of the current documentation object. (For example name of the project tree object)
\$ObjS	State of the current documentation object. (CORRECT, INCORRECT)
\$PgNr	Current page No. of the print job.
\$PrintDate	Date of the print issue.
\$PrjComm	Comment of the current project. (Editable in the project header)
\$PrjDate	Date of the current project. (As shown in the project header)
\$PrjMan	Manager of the current project. (Editable in the project header)
\$PrjName	Name of the current project.
\$PrjNr	Number of the current project. (Editable in the project header)
\$PrjOrd	Orderer of the current project. (Editable in the project header)
\$PrjOrdNr	Order number of the current project. (Editable in the project header)
%LogoComp	Default bitmap for your company.
%LogoCover	Default bitmap for your cover page.
%LogoCust	Bitmap for customer project
%LogoLeft	Default bitmap for the upper left corner.
%LogoRight	Default bitmap for the upper right corner.

Allocation of displays (only BMP files are permitted) is effected in the file DIGITool.INI. See **page G-57, Fonts, national languages and bitmaps in the drawing footer / header.**

## G 8.5 Fonts, national languages and bitmaps in the drawing footer / header

The fonts or the bitmaps used are defined in the file RedEdit.32



ti055us.bmp

BitmapDefaultDir=c:\freelance\bitmaps

Directory containing the bitmaps

BitmapDir=c:\freelance\bitmaps

Temporary directory

Font0=Arial, Italic, Size 20

in example A

Font1= Courier New, Size 22

in example B

Font2= Courier New, Size 22

in example C

Font3= Courier New Bold, Size 40

in example D

Font4= Courier New, Size 30

in example E

Font5=Courier New Bold, Size 60

in example F

Font6=Courier New Bold, Size 100

Inscription of cover page

Font7=Courier New Bold, Size 50

Inscription of cover page

Frame\_Default\_File:REG\_SZ:

path and file with the default field entries

e:\freelance\exe\frames.ini

Frame\_Default\_Sect.: REG\_SZ:

languages section in the file Frames.ini

LogoComp=mycompny.bmp

Name of bitmap for company logo

LogoCover=mycover.bmp

Name of bitmap for cover logo

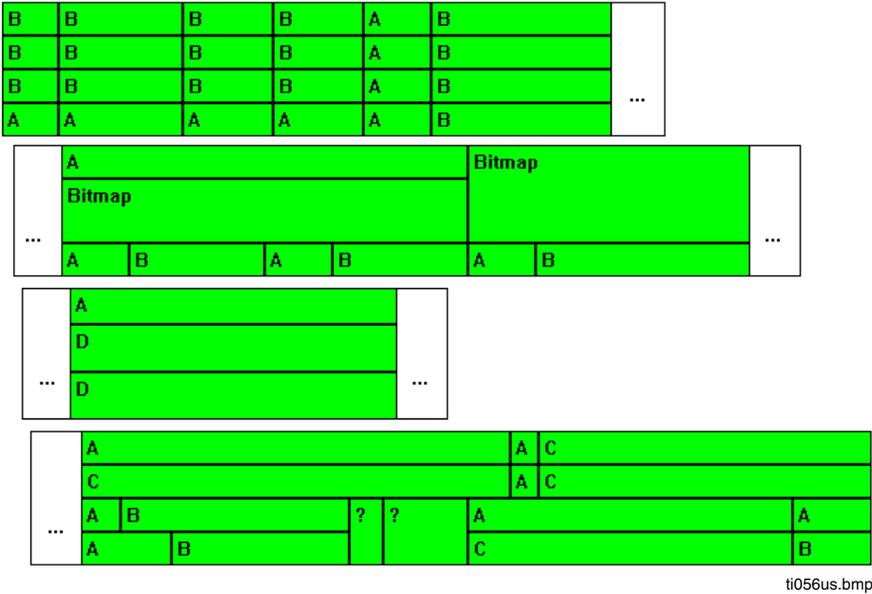
LogoLeft=mylogo1.bmp

Name of bitmaps for the upper left corner

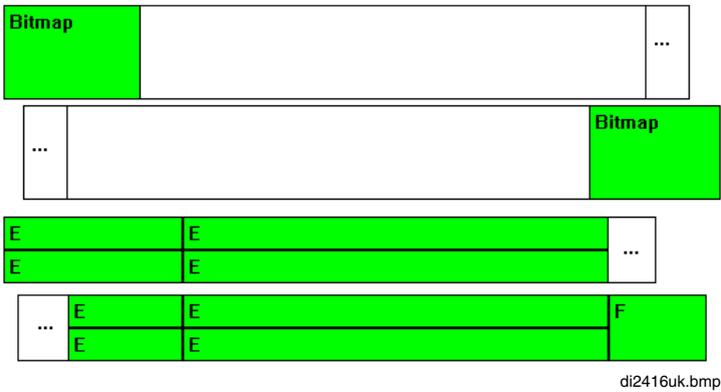
LogoRight=mylogo2.bmp

Name of bitmap for the upper right corner

G 8.5.1 Using fonts in the drawing footer



G 8.5.2 Using fonts in the drawing header



## G 8.6 Presetting the field contents and titles

The presets for the field contents or titles are located in the file FRAMES.INI. Hence in the English version the section FRAMESUS is used as default. But another section can also be selected.

See also **page G-57, Fonts, national languages and bitmaps in the drawing footer / header.**

Generally the file is located in the directory (C:\Freelance) and can be edited with any ASCII editor, e.g. EDIT of DOS or the EDITOR of WINDOWS.

The editor used must not append **any** control characters to the file!

The lines with the code FIELD\_CONT\_Pxx (xx=0..62) define the required field contents for the printout. The lines with the code FIELD\_TITLE\_Pxx (xx=0..62) specify a title for the fields. The title text can be superimposed at any time while entering the field contents to be printed, in order to show the user which entry is to be made in the fields.



# **Glossary**



Access	Parameters from function blocks and preconfigured displays that either authorize or prevent certain sizes, values or actions for the user.
Acknowledge	Operator action whereby one or more messages is/are acknowledged as seen and understood. Every action performed by the logged-in operator can be logged in the signal sequence log.
Acknowledge button	Button in the message line that enables the most recently received messages in the message line (VA, visual acknowledge) or message list (A, acknowledge) to be acknowledged without having to call up the message list.
Acknowledge level	The acknowledge level determines how events or messages on an operator station have to be acknowledged: 1) incoming and outgoing; 2) incoming only; 3) no acknowledgment.
Acknowledgment strategy	The concept of acknowledgment is part of the global event processing and is therefore valid for all operator stations. The way in which events or messages on an operator station must be acknowledged is specified here. Must both the incoming and the outgoing message be acknowledged or only the incoming message? Is no acknowledgment called for at all ?
Activate	Function forming part of the hardware structure through which a component is introduced to the system. See also de-activate.
Alarm	An alarm is an event at a point of measurement or in the system with an associated message at the operator station.
Allocate	The trends and logs run on an operator station. This means that they are displayed and archived on the operator station. However, the data they require are captured at the process station. For this purpose an acquisition module is configured in the D-PS resource.
Alphanumeric display	A dynamic symbol from the graphic editor for displaying analog values on the operator station.
Analog variable	Generic term for all non-digital variables with numeric formats such as real, integer, double integer or word.
Archive	An archive is the file in which the data from a trend display or log are saved. Depending on how data is logged, in this way large volumes of data can be generated under certain circumstances.
Area attributes	Area attributes include the color and fill pattern of the graphic objects like rectangle, polygon, ellipse and arc. These area attributes can be changed when making a display dynamic.
Authentication key	The authentication is a combination of numbers that, comes with the hard key, forms the copy protection for a DigiTool and/or DigiVis license. The authentication key is to be found on a separate disk among the installation disks.

## Glossary

Authorization disk	The authorization disk contains the rights conferred by your software license in an encoded format. You will need this authorization disk at installation time in order to be able to utilize all the features of DigiTool and/or DigiVis that you ordered.
Auto/Automatic	Operating mode in which processing is carried out automatically through the program's control settings. The operator is not allowed to do any input in automatic mode. For operator input you must change to manual operating mode.
AutoCAD	CAD program from Autodesk, produced in versions running on a variety of computers.
Autoexec.bat	File name (autoexecuted batch file) of a batch file executed when a computer is booted up with the DOS operating system and/or MS Windows.
Autostart	Autostart is a configuration parameter of the project object task. If an autostart is activated, the task will start up automatically once it has been loaded into the process station. This causes all subsidiary program lists configured as ON to be run automatically.
Background	The background is the static part of a free graphic display.
Background color	A separate background color can be selected for the draw area for each graphic display and for the graphic pool.
Band	The band is the value range of a variable that is displayed in a trend display. The limits of this range - the band start and band end - are configurable.
Band end	The band end defines the value for the end of the scale on the Y-axis in the single trend display.
Band start	The band start defines the value for the start of the scale on the Y-axis in the single trend display.
Battery, rechargeable	The accumulator is a virtual storage location for storing the interim result of an operation in the instruction list.
Block	A combination of several separate objects to form a single object with the same processing possibilities.
Block export	Combined objects (-> block) can be written to file. The data will be saved as Unicode file. The file extension specifies the type of data or the editor used to export the file. With „Import block“ data saved this way can be imported into another part of the project or into a different project.
Block import	The content of a file saved with ‚Block export‘ can be imported into the current project.

Block name	In contrast with functions, all function blocks are displayed with a block name (max. 12 characters). The block name is the same as the tag name in the tag list.
Block type	Identical to the tag type. The block type forms a short description of the block in the library.
Blocked	SFC operating mode in which a transition blocks the advancement of the program run.
Bootstrap	Loading the operating system into the RAM of an "empty" process station. The operating system can also be loaded together with the application program (menu item: Load whole station). In this case, a bootstrap is not required. → Initial all
Button	A dynamic symbol from the graphic editor. Buttons can be used to call up displays or tags, enter default values for variables or display states.
Button field	A button field comprises one or more buttons. When one of the buttons is activated, the action must be confirmed.
Button type	Form of the button in the graphic display as a rectangle, in 3-D, as an ellipse or as a circle.
Carry out	Operator action performed on an SFC program whereby all active transitions whose transition criteria have been satisfied activate the steps next in sequence.
Caption	Display of the engineering station operating mode and the currently active editor.
Central unit	A process station consists of one central unit and up to two I/O units. The central unit contains the CPU module, which does the processing.
Channel	Input or output of an I/O module. The channels of an I/O module are numbered consecutively.
Channel number	Number of a channel in an I/O module.
Class	A group of data sets or in Freelance 2000 a group of function blocks sharing the same characteristics.
Classifying	The entry of values (parameters) in a fixed configuration on the engineering station or operator station.
Coding switch	Switch at CPU module and Ethernet module used to set the Process Station or FieldController IP address. Also switch to set the rack ID on the link module.
Cold junction compensation	Method for taking into account the terminal temperature when using thermocouples for temperature measurement.

## Glossary

Cold start	State of the D-PS resource and thus also of the process station. The selected D-PS resource is restarted. The current parameter data, i.e. the data saved in the parameters, are lost. All the properly corrected parameters, on the other hand, are maintained.
Cold start executed	The D-PS resource, and along with it the assigned process station, was cold-started. The CPU module is set to RUN and is advanced to state running. All values have been set to their initial values.
Cold start stopped	The D-PS resource, and along with it the assigned process station, was cold-started. The CPU module is set to STOP and is not advanced to state running until its setting is switched to RUN. Only then is the cold-start performed and all values initialized
Cold start task	The cold start task is performed once when the D-PS resource is initialized or when a RESET command is sent to the CPU module. The cold start task is executed when the CPU module changes from cold start or cold start stopped to state running.
Colors	The colors that cannot be changed through the Windows interface can be set individually in DigiTool in the project tree and the program editors.
Combine	Function of the graphic editor enabling several objects to be configured jointly.
Comment	Fuller explanations to increase understanding can be added to the project and all project objects in the project tree, e.g. notes on how variables should be used, on a program's functionality or in general on the process field.
Commissioning	Engineering station operating mode in which there is an on-line connection to a Freelance 2000 process station and user programs can be loaded and operated. See also configuration data.
Common display pool	The common display pool (D-OSP) is a project tree object enabling all the control stations configured in the project to access the shared displays.
Communication module	The communication module accommodates five serial interface for connecting subsystems. The interfaces can be used as RS232C, RS422 or RS485, as required. Additionally, the communication module has a diagnostic interface.
Configuration	The level in the project tree and in an object for configuring a user program in accordance with IEC 1131.3. Below the configuration level lie the resources, tasks, program lists and programs.
Configure	Engineering station operating mode in which user programs can be configured or modified. Configuration can be carried out in the absence of an on-line link to a Freelance 2000 process station. See also commissioning.
Continuous controller	Continuous controllers that output a signal to the final control element continuously.

Control room horn	A control room horn can be configured along with the field horn. This is done by entering a wave file (requiring a sound board) in the local event processing. When an event occurs with the configured priority entered, this wave file is executed.
Corner	A linear attribute in the graphic editor. Corner rounding allows the corners of polygons, rectangles or continuous lines to be displayed with a rounded form.
Correct (1)	Correcting (patching) is used to change the parameters for the selected tag during commissioning. In contrast with writing, the changed parameters are saved in the project.
Correct (2)	The state Correct is only assigned if no errors have been detected in the project tree or program when running plausibility checks on the selected project objects. A program object cannot be loaded into a station unless it has passed a plausibility check.
CPU	<b>C</b> entral <b>P</b> rocessing <b>U</b> nit. Microprocessor providing the main processing functions. An Intel 80960 CPU is used for the Freelance 2000 CPU module.
CPU module	Data processing module of the Freelance 2000 central unit. Must be plugged in slot 0 (second from left). Features: 80960 RISC processor, RAM of up to 8 Mbytes, Ethernet connector.
Criteria window	Used for displaying the transition criteria and commands of an SFC program. Criteria windows can be configured both for steps (actions) and for transitions (conditions).
Cross-references	Cross-references show which programs or displays the selected variable or tag is used in.
Crosshairs	In the graphic editor one of the display formats of the mouse cursor is termed crosshairs; here there is a horizontal and a vertical line each extending across the entire draw area. The point of intersection of the two lines corresponds to the cursor position.
CSV format	A standard ASCII format (comma-separated values) used as an export format for Freelance 2000 projects or parts of a project.
Cut	In contrast with deleted objects, the selected objects can be and actually are re-inserted into the program through Insert.
D-GS resource	The communications facilities needed for the gateway are implemented automatically in the project object D-GS resource.
D-OS resource	The displays and logs that are to be displayed on a Freelance 2000 operator station are configured in the project object D-OS resource.
D-PS resource	All the parts of a user program that are to run in a process station are configured in the project object D-PS resource.

## Glossary

Data bank recovery	If the PC crashes in the course of a Freelance 2000 session, the latest position in the configuration can be recovered.
Data format	The data format specifies how a data type is displayed in the operation interface, i.e. how many places before and after the decimal point (e.g. fff.ff for 340.05).
Data type	Data types are assigned to variables either directly in the program or through a variable declaration in the variable list. Along with the basic data types such as REAL or BOOL, user-defined data types can also be set up (structured variables).
Daylight-saving-time	Freelance 2000's clock can be adjusted automatically to daylight-saving-time. A function block is provided for this purpose; this function block converts a variable of type Date and Time (DT) to daylight-saving-time.
Deactivate	Deactivating is the opposite of activating: station entries are removed from the hardware configuration.
Default display	The default display is a display with a special significance for the display selection dialog in the context of display allocation. This allows a specific display to be called up for each tag when F11-key is pressed.
Default text	Text messages set up during configuration that are incorporated into the log, the alarm list or the hints list when particular events occur; these text messages may also be displayed in the dialog line.
Default value	Configurable, fixed default value in REAL format which, when the process signal is overstepped or understepped, is transferred through an analog input converter.
Default-Task	The default task is executed as quickly as possible. This PLC mode has the lowest priority in comparison with the cyclical task. After one run-through the default task will start again automatically.
Delete	When program objects are deleted they are removed from the program permanently and the program is set to state Incorrect. When objects are deleted in the project tree they are removed permanently from the project.
Demo mode	If you start either the DigiTool or DigiVis software without the hard key, it will run as a demo version. In the demo mode the software will run for 100 days without restrictions. After 100 days the software cannot be used anymore.
Design points	When a graphic object is selected, it is displayed with design points. Such an object can be resized by clicking on one of these points and dragging it while keeping the left mouse button pressed.
Detail	Function within hardware structure in which the process station details are configured. This includes the modules fitted and the I/O channel allocations.

Dialog area	The dialog area is used for displaying either the operation dialog for operating a tag or the display selection dialog for rapid selection of related displays.
Dialog box	The dialog box is used to display messages or hints to facilitate both configuration and operation tasks.
DigiBrowse	DigiBrowse is an add-on package for the PC to view trend and log archives.
DigiDDE	The DDE server is a PC which can read any data from process stations or write any data to process stations. For this reason, the <b>DigiDDE</b> add-on package must be installed on the station (PC) which will be used to process Freelance 2000 data.
DigiLock	DigiLock is an add-on package for the DigiTool or DigiVis to permit the configuration of user groups. Various access rights can be awarded to user groups. Each user specifies an individual password which the user can only change himself/herself.
DigiTool	Software for the engineering station; runs under MS Windows. → Engineering station.
DigiVis	Software for the operator station; runs under MS Windows. → Operator station.
Directory	During installation, directories are set up in advance for the storage of Freelance 2000-specific data.
Discrete	Dynamic graphic objects are output on defined x/y coordinates in conjunction with a maximum of 3 binary variables.
Display	The sum of the static and dynamic graphic objects and graphic symbols with the associated background color. The display name is allocated in the project tree of DigiTool.
Display access	Through display access DigiTool can be used to allocate displays to tags, transitions and steps; these displays can then be called up through the display selection dialog in DigiVis.
Display format	The display format determines the format in which a variable's value is displayed in the operation interface.
Display object	Display objects can be simple or complex, static or dynamic symbols that can be created using the graphic editor and have parameters defined. A display object contains information for displaying the graphic symbol on the operator station.
Display selection	An operator action that alters the content of the display area of the operation interface.

## Glossary

Display selection dialog	Display of the operation line which enables displays to be called up rapidly. These displays are held in a fixed order which can only be changed using DigiTool. Clicking the right mouse button in the dialog area toggles between the operation dialog and the display selection dialog.
Disturbance course log	The disturbance course log is one of the state logs. It is used for logging chronological sequences of selected analog and binary variables.
Disturbance course log acquisition	Function block that records in the process station the values of up to 6 variables from a start point onwards and passes these values to the operator station.
Documentation	A variety of different forms of documentation (Project tree, Hardware structure, Program lists, Parameters etc.) are provided in DigiTool for documenting the user program. The contents, format and scope of the documentation are entirely under the user's control.
Documentation header	For each documentation job a comment can be generated using the documentation header editor; the purpose of this is to provide an explanation of the documentation content of a job.
Documentation job	A documentation job is a defining list specifying the user program areas that are to be documented.
Documentation management	When entering the documentation management system, below the menu line there appears a list of available print jobs. Blank lines in the list represent unallocated print jobs.
Documentation scope	All project objects positioned on the currently selected level of the project tree are documented.
Draw area	The draw area is the user's actual working area for creating and editing displays.
Drawing footer / header	Freelance 2000 documentation is printed out with a drawing footer and drawing header. This makes it possible for the drawing footer to be defined for the project and for each project object. At the same time system variables and bitmaps can be assigned to the individual fields
Duplicate	A selected graphic object or symbol is duplicated on the drawing surface without being saved in the clipboard as with copying. The duplicate copy is displayed on top of the original but in a slightly shifted position; it is also ready-selected.
Dynamic	When a display is made dynamic the static display, created using the graphic editor, has variables and tags from the user program introduced into it. This is achieved by accessing a shared database, thus helping to avoid errors in data entry.
Edit	DigiTool menu through which fields, objects or programs are edited. It includes functions like calling programs or displays, general parameters, inserting, copying or deleting fields or objects.

Enable	For the SFC program or function blocks to run, an Enabled state must be detected. The enable command should be issued automatically by the program or in the parameter mask.
Engineering station	PC or laptop with MS Windows and DigiTool software. Used by the Freelance 2000 system operator for configuration, commissioning, and documentation.
EPROM update	If you have been using an earlier version of Freelance 2000 and have updated to version V2.1, you must also update the EPROM in the CPU module.
Error	Errors are reported in Freelance 2000 if a program is found that cannot satisfy the plausibility check due to syntactical errors. Errors can also occur when executing a task (e.g. division by zero); such errors are reported from the process station.
Error handling	Each runtime error detected during execution of a task command results in an entry regarding the cause and location of the error in system variables (NoMathErrCorrection) specially predefined for error handling. If automatic error handling is active and a recoverable error is detected, the affected task remains in the state running. In this case the error is ascertained only by evaluation of the system variables.
Error list	All plausibility errors are added to an error list. This error list contains all the errors for the currently selected level in the project tree.
Error message	An error is reported if the system has encountered an error in the user program. In this case a system error message is generated with a message priority of 0.
Error task	The error task is the task with the highest priority (100); its function is the handling of errors in sections of the user program that form part of the common D-PS resource.
Ethernet	Serial bus system to DIN/ISO 8802, Part 3 (IEEE 802.3). Used for the DigiNet S system bus.
Ethernet board	PC plug-in board for connecting engineering station and operator station to Ethernet system bus. Built in the PC central unit. Provided with an AUI connector (SUB-MIN-D jack) or a BNC jack for connecting 10BASE2 network cables.
Event	An event triggers a message or a control action.
Event function block	The name of the event function block that starts a logging process.
Event logs	Event logs are used for logging events such as messages, faults, switching actions and/or operator actions. Logging is performed automatically when the appropriate event occurs.

## Glossary

Event processing	Event processing is subdivided into a local and a global section. The priorities allocated locally, i.e. for each D-OS resource, are those that are to be reported at the operator station. Details are also recorded for the message list and hint list as well as for horn control. The acknowledgment strategy is configured in global event processing, determining which messages with a particular priority should be acknowledged and how this should be done.
Event-controlled	A logging process or the transfer of a file is initiated by an event.
Exit	The menu choice Exit is used for leaving the current editor.
Expand	The project tree can be displayed either in expanded or compressed format by selecting the junctions. This has the effect of either opening up or closing down individual levels thus enabling the individual program parts to be displayed in a manageable way.
Export	Any parts of a project or program can be exported for use in other projects or programs. The export format used is CSV.
Exported variable	A variable declared with the export attribute can be read in other D-PS resources without any further configuration if it is declared with the export attribute.
Extension	All Freelance 2000 files are saved in the directory in which DigiTool or DigiVis was installed. This is generally a directory named Freelance. The different types of files can be distinguished from one another by their extensions, e.g. .pro, .csv, .arc etc.
External parameters	External parameters are parameters or values for a block that are passed over from the user program by connecting a signal flow line or variable.
Faceplate	A faceplate is used for operation and observation of a tag. This shows the current status from the related process. It can be incorporated into any other display.
Failure (LED)	<b>Link module:</b> Failure LED indicates the Power Fail signal of the power supply. Lights up red in case of undervoltage at the power input. <b>CPU module and communication module:</b> Failure LED indicates internal hardware and software errors. Lights up red permanently or flashes red, depending on error.
Fault message	Messages of priority 1 (red), 2 (yellow) and 3 (orange) for showing faults such as violations of limit values in the operation interface.
FBD program	An FBD program is a program created using the Function Block Diagram.
Field	In lists such as the tag lists, individual fields can be edited. Any change to particular fields such as Name or Type will result, after confirmation, in automatic system-wide changes to preserve consistency.

FieldController	Serves for process control. Next to the functions of a normal central unit it collects and processes data from up to four fieldbus lines. The FieldController runs programs that are configured under the D-FC resource.
Field horn	Along with the control room horn, the field horn is one of the components of event processing. A horn block configured in the D-PS resource converts a signal from the field (process) into an event, with the effect that a message is then sent to DigiVis.
Fill level	A dynamic symbol from the graphic editor. The fill level symbol is used to represent analog values with a polygon being filled either (optionally) from the bottom or top, from the left or right.
Fill pattern	Area attribute of a graphic object or symbol which allows the hatching style of the area to be modified.
Flash	Flashing refers to the cyclical pulsing of display objects.
Font type	Text attribute in the graphic editor.
Force	SFC operating mode whereby a transition forces an advancement when the program activates the transition but the transition criterion is not yet satisfied.
Forcing	In order to allow program response and system behavior to be tested during commissioning, certain variables can be "forced". To retrospectively build a particular system state into the program to be tested ***** Variables holding fixed values and transferred to the user program, using these values in place of the real values.
Foreground	Brings a selected graphic object to the fore. In this way the object is overlaid over any other graphic objects occupying the same position. *Overlap
Format	In Freelance 2000 a distinction is made between the logging format and the data format. Logging formats can be assigned to incoming and outgoing messages of priority levels 1 to 5 as well as to incoming operator actions. Data formats are used in graphic displays and operating logs, and they control the representation of different data types in the DigiVis displays.
Frame	The block frame delineates the block's selection area in the Function Block Diagram. Its color shows whether the block is selected or whether the parameters set for it have been flagged as incorrect. The colors used here can be changed.
Free Colors	Colors in graphic editor used to display color courses.
FTP	When trend archives or log files are transferred from an operator station to another subscriber on Ethernet, the transfer protocol used is FTP (File Transfer Protocol). The transfer may be manual, cyclical or event-controlled.

## Glossary

Function	When a function is executed it returns a single data item. Functions do not hold any state information. Each time a function is called with the same arguments (input parameters) it therefore always returns the same result (output parameter).
Function block	More complex unit for performing open- or closed-loop control tasks. When a function block set up in Freelance 2000 is used in a program it is assigned a tag name, through which a preconfigured operation is then made available to the point of measurement in DigiVis.
Function block diagram	A graphically-oriented user program for control tasks in which symbols representing functions and function blocks are arranged in a draw area; its whose terminal is linked with variables from other programs via signal flow lines.
Function key	The function keys (F1 - F12) form a standard part of the keyboard, and some of them have standard functions assigned to them.
Global variables	Global variables are variables recognized throughout the system. In order to release a variable for system-wide use, the attribute Export=Yes is all that needs to be set.
Graphic display	Free graphic displays can be created with the -> Graphic editor to visualize and operate the process at the operator station. Several static and dynamic -> graphic objects are available.
Graphic editor	An editor as part of DigiTool to create graphic displays.
Graphic object	Elements of the -> Graphic editor to create free graphic displays to visualize and operate the process at the operator station. Static objects are e.g. line, rectangle, ellipse; dynamic objects are bar graph, alphanumeric display, trend window,...
Graphic pool	Buffer used to exchange parts of graphic displays within a project.
Graphic region	The graphic region is the region in the editor in which the program itself is configured.
Graphic symbol	A special graphic object, used to make static objects dynamic, for example to change the color of a line.
Graphics animation tool	Standard component of the graphics editor enabling the display objects to be animated. The DigiVis operating dialog is configured by the process of making it dynamic.
Grid	In order to allow more accurate positioning in the graphics editor a background grid can be included in the display. The Grid command can be used to switch on and off the row and column grid lines in the display area. When grid is on, a tick appears in front of the menu choice.

Group display	A group display is a combination of several faceplates. It provides the user with the opportunity to display related tags in the same display. After a displayed faceplate has been selected, the corresponding tag can then be operated.
Hard disk requirement	The amount of hard disk space needed for a trend archive or a log file is calculated, during the configuration. If it is established at that point that the free storage capacity on the fixed disk is not sufficient, an appropriate message is issued.
Hard disk space	20 MB of free hard disk space are required for installing the DigiTool software. More disk space again will be required to work with the software. The actual amount of space required is determined by the size of your user program.
Hard key	Module which protects software from unauthorized use. The Freelance 2000 software only works if the hard key it is connected to the parallel interface of the PC central unit. You can also connect the printer cable to this interface. The hard key will not affect the printer functions.
Hardware structure	This is the area of DigiTool in which user programs are assigned to the process stations and operator stations, and addresses are allocated on the DigiNet S system bus.
Header	Parameter window for a project or project tree object in which universal details like name, short comment, and particular detail like cycle time for a task are entered. The drawing header and footer for documenting the parts of the user program are also configured in the header. During commissioning the D-PS resource, task and program lists should be started and stopped through the header.
Help	The Freelance 2000 help system enables operators to quickly call up relevant information onto the screen while they are working with the software.
Hint	The option is provided to configure a hint for the operator for every message of type Fault Message or Switch Message. These hints should be configured to inform the operator of the cause of the message, options for remedying the process anomaly and, where necessary, further operating hints. All hints are saved in the hint list.
Hint filter	This is used to specify the priority levels for which hints should be displayed in the hint list.
Hint list	The hint list contains a list of all hints that have arisen and not yet been acknowledged.
Hint message	Events with message priority 5 produce only hints but no messages. Hint messages can only be found in the hint list.
Horn	A field horn or control room horn can be controlled by an event. A horn function is also provided that allows you to attach any control signals you wish to a horn.

## Glossary

Host name	A name identifying the computer. Each name must be unique within the Freelance 2000 network. No other PC in the Freelance 2000 network may have the same name.
Hotline	If you should experience any unexpected problems during installation that are not covered here, our customer service department will gladly come to your assistance. Simply call us on: 069/799-4600.
I/O channel	The variables which are to be linked with the process via the I/O modules are entered in the I/O channel allocation. The I/O module slot and channel resulting from the allocation is shown in the variable list.
I/O module	Digital input module, digital output module, analog input module, analog output module.
I/O unit	Component of the process station. An I/O unit accommodates one link module and a maximum of nine I/O modules. The I/O modules are controlled by the CPU module of the central unit.
Icon	Windows terminology. Represents a program or a link that is started with a double-click.
Idle	The task is loaded on the process station but has not yet been started.
IL program	An IL program is a program generated in Instruction List.
Import	In order to re-use parts of other programs or projects, such previously-exported sections can be imported. When importing on the project level, these units are deposited in the project pool. The format used for import is the CSV format.
Incorrect	<p>The focus is displayed at the left-hand end of the state line. Once the focus has been selected, objects in the window can then be selected using the arrow keys.</p> <p>A plausibility check can be used to ascertain whether the selected project objects are correct. Each newly created project object in the project tree that has yet to pass a plausibility check or has not yet been correctly configured is assigned the state Incorrect. Every entry made in a program, display or log similarly leads initially to state Incorrect.</p>
Initial step	Every SFC (Sequential Function Chart) program starts with an initial step. A reset command always returns to the initial step.
Initial value	The value loaded when a process station is loaded, and therefore also when parts of a project are loaded.
Initialize	Deleting the section of the user program residing in the process station and re-starting the D-PS resource. The operating system is not affected.

Initialize all	Menu item for erasing the operating system of a process station from an engineering station. Switching off the power supply and removing the buffer battery has the same effect. After this, the process station RAM does not contain any data.
Input bar	Contains the input variables of an FBD program. See also output bar.
Input variables	Input variables are always held in an FBD program's input bar. They form the link to the system input modules (DAI, DDI) and - as flag variables - to function block outputs in other programs.
Insert	One or more program objects that have been previously copied or cut are inserted at any position in the program.
Instruction list	User program for control tasks similar to an assembler programming language. Its functional scope comprises operators that make a logical link between the contents of the accumulator and the operand. All the Freelance 2000 function blocks can be used alongside Instruction List.
Interface	The term interface refers to the point where two systems meet. The different features of the two systems are made compatible with one another through the interface (e.g. Modbus interface).
Internal parameters	Internal parameters for function blocks should only be entered within a parameter display. Details such as function name and limit values should be included here. They cannot be connected externally, e.g. via a terminal.
Internet address	→ IP address
Interpolation	The trend display and the representation therein between two captured values can appear in three varying forms: with no interpolation (whereby only the data point is shown), as a line connecting two points, or as a staircase.
Interval time	Interval time defines the time between processing of functions and function blocks from subsidiary programs.
Interval time	Interval time is entered in TIME format (e.g. T#4s500ms (=4.5s) d = days, h = hours, m = minutes, s = seconds and ms = milliseconds).
Invisible	Attribute of an area in the graphic editor. Such an area has no color, but is invisible, i.e. transparent.
IP address	Internet Protocol address of an Ethernet node, in accordance with the TCP/IP protocol. Adjustable through the coding switch at the CPU module back. The last digit of a standard address is adjusted by setting the switch to a position between 1 and 9 or A and F corresponding to the station number. Position 0 represents an individual IP address entered via the diagnostic interface.
Jump	A jump is brought about through a conditional jump instruction. Jumps are possible as operators in the Instruction List or as elements of the Sequential Function Chart.

## Glossary

Junction	Graphical representation of a project object in the project tree, indicating the object's state.
Keyboard	Input device of operator and engineering station. Standard AT keyboard layout (MF2). Keyboards to IP 65 available on request.
Label	Jump markers from L001 to L999 (labels) can be entered in the Instruction List (IL); these serve as target addresses for jump operators. These labels may be entered in any order.
Laptop	Portable personal computer. Preferably used for engineering station.
LatCRcv task	All global variables can automatically be read in other D-PS resources if they are enabled for this through the attribute Export = Yes in the resource to be sent. The cross-communication necessary for reading or receiving is performed automatically by the LatCRcv task (lateral communication receive task).
Lateral communication	Name for the cross-communication between the D-PS resources and thus between the process stations. The communication is handled automatically through the lateral tasks. See also export.
Level	The project's structure is provided by the levels in the project tree. Project objects of the same type are all held on the same level.
Library	Comprises on the one hand the block library supplied in Freelance 2000 by ABB, and on the other hand the macro library for graphic symbols.
License options	The license options indicate the functional scope authorized for the currently-installed DigiVis version. See also authorization diskette and hard key.
Limit value	Limit values form the basis for generating events. Limit values are thus declared in the operation interface and/or in a log, e.g. when a particular value is exceeded. Particular events are specified depending on the type of limit value. The message is generated with, amongst other attributes, a specific priority and a message text.
Line attribute	The attributes of a line are its color, weight, line type, line start and end and the corner style. The border color of graphic symbols can also be changed in the context of making them dynamic.
Line number	Line numbers are assigned automatically in Instruction Lists, running sequentially from 1 to 1000. If empty lines or command lines are inserted, the command lines further down the list are automatically incremented by the number of lines inserted.
Line start / end	Attribute of graphic object line. It is displayed either with or without an arrow (optional).
Line style	Attribute of the graphic object line. The line or border color can be displayed either as dashes, semicolons or a continuous line.

Line type	Attribute of graphic objects. The line or border color can be displayed either as dashes, semicolons or a continuous line.
Link module	<p>Module for supply voltage input and for connection of external batteries. Essential part of central units and I/O units. Must be plugged in slot A (first slot on the left hand side).</p> <p>Two different models of the link module are available. The standard link module DLM 01 provides a simple power supply. Link module DLM 02 with a redundant power supply meets higher requirements in terms of power supply availability.</p>
Load	Transfer of programs or sections of programs in commissioning that have passed the plausibility check without revealing any errors. In addition the objects or levels in the project tree are selected and transferred automatically to the assigned resource by the load procedure.
Log type	Three different types of logs are provided: the signal sequence log (SSL), operation log (OPL) and disturbance course log (DCL) .
Logs	Logs are used for documenting events, states and sequences from the process. The data they capture can be saved on the hard disk in the operator station, printed out on the printer or displayed on the monitor.
Long text	A text entry up to 30 characters in length for providing a brief description of a function block or preconfigured display.
Loop	A loop is a repetition of one section of a program that is self-contained and can be performed more than once. Every loop must have a correct entry point and exit point; if this is not the case the result is an endless loop, which is not allowed. There must also be a condition to determine whether or not the loop is entered. This is implemented by means of a conditional jump.
Macro	A graphic symbol composed of one or more graphic objects or symbols that share a common name. This unit can be used over and over again and can be retrieved from the macro pool and saved in a macro library.
Macro libraries	Project-independent collection of macros, created using the graphic editor and copied from the macro pool into the library.
Macro pool	The collection of all the macros in the project. The macros in the pool can be copied into a macro library from where they can be used in other projects.
Mandatory parameters	Mandatory parameters are essential details of function blocks such as tag name, scale start and scale end, and the link with an input or output variable such as process value, controller correcting variable or other function-specific parameters.
Manual	Operating mode in which the tag can be operated from within DigiVis.

## Glossary

Mark	All the lines from the IL program belonging to a function block are here given a color as long as the mandatory parameters are not all in use. When no further mandatory parameters are free, these fields are grayed out.
Media Colors	Contain Colors that flash of their own accord.
Message	Certain process states and/or events can be configured as messages. When that event occurs a message is then sent to the operator station. Such messages are configured through the function blocks.
Message box	The message boxes in the message line contain the first five still-to-be-acknowledged messages with priority 0 to 3.
Message Colors	Colors for message display.
Message filter	Definition of which message priorities are displayed in the message list.
Message line	The message line forms part of the operation interface and is included above every display on the operator station. The message allows the operator to log the most recently received messages without having to switch to the message list.
Message list	<p>In the message list the messages sent from the process stations to the operator station are displayed and managed. These messages may relate either to the Freelance 2000 system or to the automated process.</p> <p>In the message list a message can be selected and/or acknowledged, and the display assigned to a message can also be called up through the display selection dialog. In this way one can quickly obtain detailed information relating to the selected message</p>
Message order	Definition of the position of the newest message in the message list. Thus, when set to 'Display newest message at top', any newly-arrived message will appear at the top of the message list.
Message overflow box	A box in the message line used to indicate that the message line contains more than 5 messages.
Message page	A page from the message list that can be displayed on the operator station.
Message priority	6 different priority levels are used, with priority 0 being reserved for system errors, 1 to 3 for fault messages, 4 for switch messages and 5 for hint messages.
Message text	Additional output text of a message in the message box and in the message list.
Message types	In Freelance 2000 messages are subdivided into the following message types based on their significance for the process: system errors, fault messages, switch messages, hints and hint messages.
Monitor resolution	Number of pixels that can be displayed on the monitor, e.g. 640 x 480 or 1024 x 768.

Monitoring time	The length of time after which the transitions/advancing criteria that follow a step but have not been satisfied will be reported.
Mounting	Configuration element of the hardware structure whereby a module is assigned to a slot in the process station.
Mouse	Hand-operated device used for moving the cursor around on the screen. The left-hand mouse button is used for selecting objects from the graphic display. The right-hand mouse button enables the display selection dialog in DigiVis to be operated.
MS-DOS	Operating system used for operator station and engineering station.
MS-Windows	Graphical user interface. Requires MS-DOS.
Multicolored	Area attributes setting through which the colors of the static graphic symbol are adopted when the symbol is made dynamic.
Name	In the graphic editor a symbol name can be assigned to a graphic object. Names are textual labels for objects within the project. A name should be unique within the project, and every object must be given a name.
Natural sequence	Sort criterion that lists the variables or tags in chronological order of configuration.
Network	Part of the hardware structure allowing station numbers and IP addresses to be entered and configured on the DigiNet S (Ethernet / TCP/IP) system bus.
Network board	Plug-in Ethernet card for a PC through which connection to the DigiNet S system bus is effected.
Normal	Operating mode in the Sequential Function Chart in which a transition or step is processed normally. This means that the transition is neither blocked nor forced and steps are not switched permanently on or off.
Object list	A list of all the selected project objects.
Offset mode	This operating mode is used to determine which offset value is being used in the time scheduler.
Offset value	Value added to the time scheduler's set point.
On line	Commissioning is an operating mode of DigiTool, which works only with an on line connection to the Freelance 2000 process and operator stations.
Operand	In the Instruction List the accumulator combines the contents of the accumulator with the operand in accordance with the operator command and saves the result in the accumulator. The operand here may be a constant or a variable.
Operate and observe (man machine interface)	A general term describing the process-control approach in an industrial process.

## Glossary

Operating system	The operating system (firmware) is the software in the CPU module that enables communication with the process station and enables the user program to be run. Further functions of the operating system are program management, test routines and capture of I/O signals.
Operation dialog	Operation of a display or point of measurement is enabled through the dialog area. Clicking the right mouse button in the operation line toggles between the operation dialog and the display selection dialog.
Operation log	Cyclical, manual or event-related logging of up to 200 variables within a configurable text.
Operator	The person who holds responsibility at any given point in time for controlling process events through the control system. The operator is a standard function in the Instruction List IL such as Load LD or Save ST. The operator combines the accumulator with the operand. The result of this operation is saved back in the accumulator again.
Operator action	An operator action is an action on the part of the operator through which a process state or process value is changed from the operator station.
Operator hint box	Display field in the message line showing whether or not there are any available hints relating to a message or event. Through the operator hint box or message list one reaches a hint list showing any unacknowledged hints.
Operator interface	The operation interface is the sum of all the display objects and operating objects at the operator station.
Operator intervention	Intervention in the process by the operator. An operator intervention can only be carried out if the access parameter has been set to 'Yes' for this task. Furthermore, the operating rights for the operator currently logged in must be allocated.
Operator station	PC with MS Windows and DigiVis software. Used for operation and monitoring, alarms, trends, archives, and reports.
Optional parameters	Optional parameters are function block attributes such as short text, long text, dimension, conductivity and limit values - attributes that are not strictly essential. These function block parameters are set to default values and need not necessarily be entered.
Options	Operating mode in SFC programs whereby the progression through the following steps is controlled through actions on the part of the operator.
Overlap	Function of the graphic editor for moving graphic objects either to the foreground or the background when they are overlapping one another.
Overview display	Preconfigured display for quick selection of displays and/or logs.
Page layout	Specification of the output format of a log.

Parameter	Parameters are configurable attributes effecting the processing and display of function blocks, displays and logs.
Parameter mask	Dialog window through which parameters can be entered and modified.
Parenthesis depth	Parenthesis depth of logical operators in the Instruction List from level 1...8.
Permanent	Scheduler operating mode whereby all sections of the set curve are run through systematically.
Permanent off	SFC operating mode whereby a step is switched off permanently. Activating it has no effect.
Permanent on	SFC operating mode whereby a step is switched on permanently. The step is not necessarily active but will nonetheless be kept in the processing sequence.
Plausibility check	A project tree function whereby only those project objects that have not yet satisfied a plausibility check are checked. The state Correct is a requirement for the subsequent commissioning of the project. Incorrect project objects are displayed in the color selected for this purpose.
Plausibility check errors	Any errors detected by the plausibility check on the selected project objects that render commissioning impossible are displayed together in the error list. Along with the plausibility check errors, warnings are also issued in the case of less serious problems.
PLC mode	Task operating mode through which a D-PS resource task is processed as quickly as possible. This gives rise to variations in processing time according to system loading. In order that the cyclical tasks are processed correctly, the SPS mode task is assigned the lowest possible priority (50) within a D-PS resource.
Points	Design points for the graphic objects line and polygon can be repositioned, deleted or added.
Pool	An area of the project tree containing incorrect project objects or project objects that are no longer required in the sequence but which one might want to reintroduce into the process under certain circumstances.
Position	The position of graphic symbols can be shifted either continuously or in discrete steps within the display area according to an analog variable.
Preconfigured display	Along with the free graphic displays created using the graphic editor there is also a series of preconfigured displays; to display these in DigiVis you only need to configure a tag and a few parameters. The preconfigured displays in Freelance 2000 are the overview display, group displays, trend displays, SFC displays, time scheduler display and system display. The faceplates for each tag and the logs can also be displayed without any need for further configuration.

## Glossary

Print job	The output of documentation is always controlled by a so-called print job, i.e. there must always be a job selected before printing. The content of the print task is defined in this job.
Printer	Printers can be connected to the operator and to the engineering station. Different printer types can be used ; inkjet printer (b/w or color), depending on the requirements.
Printer setup	Printer selection, setting of options, see also Windows Print Manager.
Priority	The multitasking of several tasks with the same cycle time is controlled through priority (not to be confused with message priority). The lowest priority that can be assigned to a user task is 51 and the highest is 99.
Priority levels	Messages received from the process stations may have differing levels of importance in terms of effect on the smooth operation of the process. The Freelance 2000 system provides six different priority levels for messages.
Process image	A temporary storage location for the task where the task variables are read before the task is run and where those variables are written back again after the task has been run through a single time. These variables are flagged with the @ character.
Process station D-PS	Serves for process control. Consists of a central unit and a maximum of four I/O units. The process station contains the CPU unit, running all programs that are configured under the D-PS resource. *D-PS resource
Processing number	In the function block diagram, FBD, the key numbers displayed in the bottom-right hand corner of the functions and function blocks indicate the processing sequence within this FBD program.
Processing sequence	The processing sequence shows the sequence in time in which the project objects (e.g. program list, program or function block) contained in a task are processed.
Program	The structural unit of a project within which the open- and closed-loop control tasks can be configured. The Function Block Diagram (FBD) and Instruction List (IL) are provided for defining programs.
Program elements	Program objects are the smallest units of a program that can be displayed, namely functions and function blocks, signal flow lines and, sometimes, sections of such lines along with variables from the input and output bars.
Program list	Structural element of the project controlling the allocation of the different parts of the user program to the tasks. The cycle time is set in the task, while the sequence of processing individual programs is controlled in the program list.
Project documentation	All the components of a project can be documented directly. This is achieved by assembling the required documentation types in a documentation job.

Project name	A single project name is assigned to all the various parts of the user program. This project name will be used later for documentation purposes and also forms the file name for configuration.
Project object	The smallest unit of a project that can be loaded on the process station is a project object. Some of these project objects are visible, such as: configuration, resource, task, program list, program, function and function block. There are also invisible objects such as process image, variables and messages.
Project tree	An organizational tool used for structuring a project in the form of a tree.
Project version	The project version is made up of three parts as follows: Part 1 (major) states the number of bootstraps; Part 2 (minor) is incremented by 1 every time program sections are loaded; Part 3 (patch) states how often parameters have been corrected. All three parts can be read through system variables.  A version control system allows a comparison to be made between the engineering station and the process station.
Rack	Framework for the process station modules; used for the central unit and the I/O unit(s).
Rack ID	Identification code of the rack. Settable with the coding switch at the link module back. → Coding switch.
Radio button field	A dynamic symbol from the graphic editor allowing up to 25 values to be preset by selecting a radio button.
Reference point	Coordinates of a static/dynamic display object in a graphic display with particular related functions such as mirroring.
Reference window	The reference window is a rectangular screen region containing information on the graphic display and is the same size as a faceplate in DigiVis.
Reflect	Function of the graphic editor whereby one or more graphic objects can be displayed as either a horizontal or vertical mirror image.
Refresh cycle	The time in which I/O signals are exchanged between the I/O module and the CPU module. The refresh cycle is a mandatory parameter for the module.
Repeat time	The time at which an SFC program is restarted.
Report printer	The report printer is the printer connected to the operating station. A maximum of 2 printers can be connected to one operator station.
Reserve bar	The section in an FBD program through which variables are exchanged between the programs and displays. The signal flow in an FBD program always runs from left to right, from the input bar to the output bar.

## Glossary

Reset	When the reset button is activated, the SFC program is returned to its initial step. A reset can only be performed in manual operating mode. Counters can be reset to their Basic value.
Reset switch	Push-button switch at the CPU module. Resets the module, i.e. initiates a cold start when pressed for more than 5 seconds. Mounted recessed. Can only be pressed with a "tool" (e.g. ball-point pen, paper clip, etc.).
Resource	→ D-GS resource, D-OS resource, D-PS resource
Restart time	The restart time is the point in time at which the SFC is to be started. Unlike repeat time, restart time represents a single point in time for starting the SFC.
Rotate	Graphic editor function whereby one or more graphic objects are turned through 90 degrees counter-clockwise about the point central to the design points.
Run task	The run task of a D-PS resource is executed once when the D-PS resource is started.
Run/Stop switch	Stops functional sequence processing when set to the <b>Stop</b> position. Communication with the operator or engineering station via Ethernet is still possible. Processing stop is indicated by the Run/Stop LED. The process station can also be stopped from the engineering station, independent of the Run/Stop switch. If, however, the switch is in Stop position, the process station cannot be started from the engineering station for safety reasons.
Running	State of the task in the D-PS resource. The operating system is loaded and the RUN/STOP switch on the CPU module is set to RUN.
Save	Saving is the transaction whereby changes are saved in the project database or project file.
Save as	The Save As facility enables changes to be saved under a different project name.
Screen	An alternative term for the operator station monitor, which can be used for operating tasks.
Selection area	A dynamic symbol from the graphic editor. The selection area can be used to configure display calls. By selecting a selection area in DigiVis, the tag entered is called up or the current display on the operator's screen is replaced.
Selection	Operating technique from the operation interface. Selection is used to activate displays or to select tags for operation.
Selection list	Selection lists provide a way of accessing the project database in order to insert variables, tags or text into a program, display or function block.

Sequence selection	An object of the Sequential Function Chart for formulating alternative program paths. A sequence selection divergence represents the start of an optional branch in the sequence.
Sequence selection convergence	An object of the Sequential Function Chart for formulating alternative program paths. A sequence selection convergence represents the end of an optional branch in the sequence.
Sequential function chart	The Sequential Function Chart (SFC) is used for programming sequence control functions. It can be used to break down complex tasks into manageable units and to describe the flow of control between these units. See also SFC program.
Sequential function chart display	Standard display in DigiVis that allows a program sequence set up using the Sequential Function Chart to be operated.
SFC program	An SFC program is a project object within a task which, unlike the Instruction List, processes the allocated programs with the aid of a structure (Grafcet).
Short comment	A text entry in a project object's header which may be up to 159 characters in length. Unlike the fuller comment for a project object, only short comments can be entered here.
Short term archive	When a trend display is selected, the short-term archive for the individual signals is displayed first. The short-term archive holds 200 values, i.e. max. 6 x 200 values per trend display. For older values the system will then automatically return to access the archive file.
Short text	A text entry up to 12 characters in length as a brief description of a function block or preconfigured display.
Side effect	A configuration change that has an effect on the processing of the higher-level task or D-PS resource. Any tasks or D-PS resources flagged in this way must be re-loaded in order that the configuration change can come into effect.
Signal colors	Special signal colors are made available in the graphic editor's color table to enable event states to be displayed consistently in the operation interface.
Signal flow	The signal flow in an FBD program runs from left to right, from the input bar to the output bar.
Signal flow line	Signal flow lines (wiring) are the graphic representation of the signal flow in an FBD program. A signal flow line shows the relationship between a variable and a function block or between different function blocks.
Signal sequence log	Configurable logging of system errors, fault messages, switch messages, operator hints and operator actions.
Signal sequence log 1	Logging of system errors, fault messages, switch messages, operator hints and operator actions with continuous printing.

## Glossary

Signal sequence log N	Logging of system errors, fault messages, switch messages, operator hints and operator actions, recording these in a log file.
Simultaneous sequence convergence	An element of the SFC for formulating parallel program sequences. A simultaneous sequence convergence represents the end of parallel sequence branches.
Simultaneous sequence divergence	An element of the SFC for formulating parallel program sequences. A simultaneous sequence divergence represents the start of parallel sequence branches.
Slot	The rack has ten slots where the modules are plugged in. → Slot number
Slot number	Identifies the slot in the rack. The slots are numbered consecutively from the left to the right: A, 0, 1, 2 ... 8.
Software error	An error has occurred in the software that is preventing normal processing.
Sort	Lists can be re-ordered through sorting.
Sort criterion	The sorting order is defined by the sort criterion. Sorting can be performed in configuration order (the natural order) or by individual columns.
Sound board	Plug-in card for a PC that plays back acoustic files (wave files).
Start characteristics	Definition of the response of the system software at system startup.
Start time	The start time is the time at which the SFC program is to be/was started for the first time.
State log	The function of state logs is to record process states cyclically. This includes the cyclical logging of the state of a tag or the logging of sequences in the process.
Static colors	General colors from the graphic editor's color palette.
Station number	Adjusted with the aid of the coding switch at the CPU module back. In switch positions 1 ... 9 and A ... F the station number is the last digit of a standard IP address. → IP address The station number should be written into the appropriate field on the rack type label.
Station reserve	The station reserve is the detailed view of a process station. In the station reserve the I/O modules are fitted in the hardware structure.
State line	The state line shows the name of the program being processed and its current plausibility state. It is only allocated the state Correct if the program satisfied the plausibility check.

Step	An element of the SFC controlling actions in conjunction with a transition. In Freelance 2000 a string of FBD and/or IL programs continues to be processed until the next transition is satisfied.
Step list	List of all the steps in an SFC program.
Step name	Free-form label (8 characters) displayed and documented in the graphic object of a step (SFC program).
Stop Task	The stop task is executed once when the D-PS resource state changes from Running to Stopped. The stop task is not activated until all other tasks have been stopped. A restart cannot be performed until the stop task has finished executing. If the stop task does not contain any programs, it will not start.
Stopped	Task state in the D-PS resource. The operating system is loaded and the CPU module's RUN/STOP switch is set to STOP.
Structured variable	Structured variables are variables that can be configured with user-definable data types. The data type consists of a combination of labeled components of differing data type.
Structured data type	The structured data types are user-definable data blocks comprising a number of basic data types such as BOOL or REAL.
Subnet mask	Masking in the main network for a possible lower-level network.
Switch message	Priority 4 message (yellow) for reporting switching events such as Pump On; not to be confused with a fault message.
Syntax error	Syntax comprises a set of formal rules needed by a programming language in order to function correctly. Syntax errors occur when these rules are disobeyed, and they lead to a message through a dialog box or to a plausibility error.
System bus	All stations are linked through the DigiNet S system bus. This system bus is based on the Ethernet standard with the TCP/IP transfer protocol.
System log	Archive file in which all actions on the engineering station are logged.
System message	System errors have the top priority of 0. Messages with this priority cannot be either configured or changed by the user. System errors generate messages about error states in the system itself.
System task	General term describing all the preconfigured tasks in a D-PS resource.
System time	The system time is synchronized for all stations. The system time can be set throughout the entire system from the engineering station.
Tag	A tag is a preconfigured function block. Every function block must be assigned a tag name. All the tags recognized by the project are listed in the tag list.

## Glossary

Tag list	This list holds all the function blocks configured in the system which have already been assigned a tag name.
Tag type	Brief description of the function block in the function library. See also function type.
Task	The <b>tasks</b> control the processing of the user program parts in a process station. A distinction is made between cyclical tasks and those which are processed as quickly as possible.
TCP/IP	Transmission protocol for Ethernet system bus. In accordance with layers 2 to 4 of the ISO/OSI seven-layer model.
Terminal	<p>Display terminal with serial RS232C interface. Usable for diagnosis as an alternative to a PC with terminal emulation. Control sequences in accordance with ANSI standard, therefore often called "ANSI terminal".</p> <p>The terminals are the inputs and outputs for the blocks in the Function Block Diagram. A distinction is made between mandatory and optional terminals. Mandatory terminals require supply via a signal flow line in order that the module can operate correctly, while optional terminals do not. This difference is represented visually by making the optional terminals shorter.</p> <p>Through the parameter definition of constants certain optional terminals become completely redundant. As with the signal flow lines, the color and line width of the terminals contain information about the data type required or set.</p>
Terminal designation	Alongside every function block terminal a code indicates the function of this connection, e.g. EN for enable.
Text attributes	Describes the format in which text appears. Font size, font, text orientation and colors can all be selected.
Time scheduler	With a time scheduler, analog values defined as time-dependent are output; these values can be transferred to other functions, e.g. to a controller as a set-point input.
Time scheduler display	The time scheduler display displays the time scheduler's state and enables it to be controlled by the operator. This display consists of a trend region for tracking set points and process values, a state field and the associated faceplate.
Time synchronization	After the Freelance 2000 software has been installed the date and time in the PC are synchronized with the date and time in the Freelance 2000 process station.
Transition	<p>The transition criterion is formed by a variable (.RESULT) of data type BOOL. If this is set in the transition program to logical 1, the transition is kick-started.</p> <p>An element of the Sequential Function Chart that activates the following step (altern. situation) or steps (parallel situation). Within a transition in Freelance 2000, an FBD or IL program will continue to be processed until the transition criterion is satisfied.</p> <p>Condition that must be satisfied in order for a step to activate subsequent steps. See also step.</p>

Trend	The function of a trend is to provide a graphic representation of values using a time axis. In Freelance 2000 this is done by capturing the values in the process station and transferring them to the operator station in blocks.
Trend data acquisition	Function block that captures the signal values of the linked variables and transfers them in blocks to the operator station.
Trend display	The trend display is used to display values graphically using a time axis. A maximum of 6 trends can be shown in one trend display.
Trend window	Trend display on the engineering station that, like the value window, can be called up when in commissioning mode.
Undo	This function, which appears under Edit, can be used to reverse the last change, e.g. deletion of an object.
Unrunnable	Task state in the D-PS resource. The task is loaded into the process station and the automatic error detection system has detected an error such as division by zero. → Error handling
User FB variables	The terminals for the user-definable function blocks in FBD are declared using variables saved in the input and output bars in the FBD program. The first three numbers held in these variables then form the subsequent terminal designation.
User function block	A user function block is a function block created by the user that can be reused in other programs in the same way as other function blocks.
User function block pool	Within this pool the user-defined function blocks are configured in the project tree in the form of programs.
User menu	The user can define a menu that combines frequently used function blocks
User program	The sum of all the open- and closed-loop control tasks in the process, collected together as a D-PS resource and configured using DigiTool.
Value	Numeric value of a variable that must be entered in a specific format depending on data type.
Value window	Display in the commissioning phase allowing several variables in a variety of numeric formats to be displayed together.
Variable	A variable is the data carrier for a value or piece of information. The variables represent the link between the programs and displays; information based on a data type is transported from one location to another through a variable.
Variable list	The variable list contains all the variables configured in the project as inputs, outputs and flags from the programs and modules.
Variable window	A dialog within commissioning that enables one or more variables to be allocated to the value display or trend display.

## Glossary

Version	The Freelance 2000 software version number (e.g. V2.1) provides information on its release date and functional scope. Along with the Freelance 2000 software version number there is also a project version number which can detect and report whether the version of the user program loaded matches the version in the engineering station.
Version control	A version control check is carried out in order to check which project is loaded on the selected process station and whether this project matches the one currently open in DigiTool.
Version error	There are discrepancies for the user program between the process station running and the D-PS resource assigned. These version differences are reported irrespective of the resource's run state, in other words even when it is Stopped.
Vertical line	An element of the Sequential Function Chart enabling a vertical connection to be established between steps and transitions.
Visual acknowledgment	Only the messages in the message line are acknowledged. The message itself remains in the message list and must also be acknowledged there.
Waiting time	The waiting time TWA is the minimum duration of time the SFC program will remain in a step.
Warm start carried out	Warm restart carried out means that after the conditions for a warm restart were satisfied (power failure), all programs and function blocks were reset (initialized) and variables returned to their initial values.
Warm restart stopped	A warm restart was executed and the RUN/STOP switch is set to STOP.
Warning	Errors detected during the plausibility check of project objects that are not serious enough to prevent execution; the objects, however, are either incompletely configured or could be configured more simply.
Watchdog	Self-monitoring function in the CPU module.
Write	Transaction from DigiVis or the engineering station whereby a modified parameter from a tag is written to the process station. Unlike correction, such changes are not saved in the project file.
Zoom	4 zoom levels are provided in the Freelance 2000 graphic editor, each of these levels displaying the selected zoom region with greater magnification than the preceding level.
Zoom region	The zoom region is used to select the screen region to be displayed in the graphic editor's display area.

## **Features Added In Previous Versions**



**Contents**

<b>1</b>	<b>Changes in Versions 3.1 and 3.2</b>	<b>5</b>
1.1	General	5
1.1.1	Windows NT operating system	5
1.1.2	Modules	5
1.1.3	Redundancy	5
1.2	DigiVis	6
1.3	DigiTool	7
1.3.1	Project tree	7
1.3.2	FBD program	8
1.3.3	Ladder Diagram	8
1.3.4	Tag list	8
1.3.5	Variable list	8
1.3.6	Hardware structure	9
1.3.7	Commissioning	9
1.3.8	Trend	10
1.3.9	Logs	10
1.3.10	Graphic editor	10
1.3.11	General improvements	11
1.4	Function blocks	11
1.4.1	Send/receive blocks	11
1.4.2	Analog blocks	12
1.4.3	Binary blocks	12
1.4.4	Controller blocks	13
1.4.5	Acquisition blocks	14
1.4.6	Monitoring blocks	14
1.4.7	Control blocks	14
1.4.8	DigiBatch blocks	15
1.4.9	Remote control blocks	15
1.4.10	Sartorius balance block	15
<b>2</b>	<b>Changes in Version 3.3</b>	<b>16</b>
2.1	General	16
2.1.1	Process stations and racks	16
2.1.2	Modules	16
2.1.3	OPC	16
2.1.4	Licensing - Safety mode	16
2.2	DigiVis	17
2.3	DigiTool	17
2.3.1	Post-loadable libraries	17
2.3.2	Project tree	17
2.3.3	Variable list	18
2.3.4	Tag list	18
2.3.5	Commissioning	18
2.3.6	Graphic editor	18

## Features Added In Previous Versions

<b>3</b>	<b>Changes in Version 4.1</b>	<b>19</b>
3.1	General	19
3.1.1	User-defined function blocks	19
3.1.2	Modules	19
3.1.3	DigiNet P	19
3.2	DigiVis	19
3.3	DigiTool	20
3.3.1	Update	20
3.3.2	Project tree	20
3.3.3	Hardware structure	21
3.3.4	Commissioning	21
3.3.5	Graphic editor	21
3.3.6	Trend	22
3.4	Function blocks	22
3.4.1	Monitoring blocks	22
3.4.2	Interfacing Protronic / Freelance 100	22
3.4.3	Interbus blocks	23
3.4.4	Description of the block parameters	23
<b>4</b>	<b>Changes in Version 5.1</b>	<b>24</b>
4.1	General	24
4.1.1	FieldController	24
4.1.2	Profibus	24
4.1.3	Modules	24
4.2	DigiVis	25
4.3	DigiTool	25
4.3.1	Toolbars and context-sensitive menus	25
4.3.2	FBS and KOP editor	25
4.3.3	Sequential function chart	25
4.3.4	Variable list	25
4.3.5	Hardware structure	26
4.3.6	Commissioning	26
4.3.7	Documentation	26
4.3.8	PLC export	26
4.3.9	Display allocation	27
4.4	Blocks	27
<b>5</b>	<b>Changes in Version 5.2</b>	<b>28</b>
5.1	General	28
5.1.1	Length of tag names	28
5.1.2	Profibus	28
5.1.3	Sequence of Event Message	28
5.1.4	Function block diagram	29
5.2	Freelance <i>select</i>	29
5.2.1	Starter kit	29
5.2.2	Freelance <i>select</i> functional capabilities	30
5.2.3	Freelance <i>select</i> upgrade	30
5.3	Blocks	30

## 1 Changes in Versions 3.1 and 3.2

### 1.1 General

The compact control system known until now as DigiMatik will be marketed internationally under the new name **Freelance 2000** from software version 3 onwards.

#### 1.1.1 Windows NT operating system

**Windows NT 4.0**, as a system platform for the future, forms a foundation for a high degree of operational security and performance.

In the process of porting it to Windows NT 4.0, the software engineers also adapted the Freelance 2000 software for the use of unicode. Only the Windows applications in Freelance 2000 are affected by the conversion to unicode; the operating system in the process stations has not changed as a result.

#### 1.1.2 Modules

##### **CPU module DCP 10 for redundancy**

The new CPU module DCP 10 for redundancy is now available, enabling CPU redundancy to be configured.

##### **Digital input module DDI 04**

This module enables inputs to be connected directly for NAMUR initiators, 3- or 4-wire initiators or contact interrogation (make, break or changeover contacts).

##### **Rack DRA 02**

In a type DRA 02 rack the redundant CPU modules can be installed in slots 0 and 1 in the main rack and slot 0 in the supplementary rack.

#### 1.1.3 Redundancy

Version 3 offers the ability to increase the availability of the Freelance 2000 system by means of CPU redundancy.

The CPU redundancy has been implemented in such a way that users both of DigiTool and DigiVis see only one process station. The fact that a redundant process station has two CPU modules only needs to be taken into account initially during configuration and later when evaluating the state during the commissioning/servicing process. In all other situations a redundant process station behaves like a non-redundant process station with only one CPU module (single-machine model).

 In order to upgrade V3 with redundant operation, the I/O EPROMs of all I/O modules in the process station must be changed.

For V3 projects that are to be operated with no redundancy, non-redundant operation is possible with V2 I/O modules.

### 1.2 DigiVis

#### Number of operator stations

**10 operator stations** can be connected in a Freelance 2000 system.

#### Plant areas

Each point of measurement can be assigned to one plant area from a maximum of 15 (A...O). The plant areas are shown in the **message page** and the **signal sequence log**.

#### Diagnostic password

The system display contains information that is more important to some users than to others. Thus, certain items of data are provided in the system display which are of interest only to staff who have received specialized training in diagnostics. In order to avoid diagnostic data from being accessed by staff other than maintenance engineers, this data is protected by an access right - the diagnostics right.

#### Customizing alarm colors

To enable specific country-specific or company-specific requirements to be met, the facility is provided to change the display colors for the various message priorities.

Since a modification of this nature has an effect on the entire visual appearance of DigiVis and also includes aspects with relevance to safety, the person responsible for commissioning or the service technician must be involved in implementing it. The Freelance 2000 documentation always refers to the standard color settings.

#### Trend display

The window that is displayed can be positioned at a predetermined point in time on the curve. Changes made by the operator in DigiVis are stored. These settings are used to show the trend display the next time it is called. The display can be reset to the configured values.

#### Display of the number of hard copies

To provide the DigiVis operator with information on the status of hard-copy print jobs, a display has been incorporated in the DigiVis status line.

#### Logs

The output format for signal sequence logs (80-character display) has been changed.

If the remainder of the line is not long enough to output the old and new value of an operator action, then the output will wrap round over one or more new lines. The new value is displayed in line with the old value.

#### Headers and footers in logs

Headers and footers in logs can be configured freely. Fixed text, field references and variables can be used in header and footer text.

#### Reserving space for archives

After the software is downloaded onto the D-OS, the maximum disk space that could be required by the archives is set up straight away and permanently reserved.

### Input formats of data types

Additional input formats of data types have been provided.

Strings may now be entered without enclosing them in quotation marks. A date (**DT**) can now be entered without a leading **DT#** and time (**TIME**) without a leading **T#**. REAL numbers can now also be entered without a decimal point.

### Improved tag selection

The selection dialog for tags has been changed over to a combo-box with "type-ahead" functionality. The requisite tags can now be found and selected more quickly by entering a search string.

### Grouping of system error messages

All system error messages have been classified into so-called system message groups.

## 1.3 DigiTool

### 1.3.1 Project tree

#### New or changed project elements

(CONF)	Configuration	The numbers of operator stations and gateways should be specified here (default 5 D-OS and 0 D-GS).
(D-PS/RED)	Process station	A redundant process station is defined by two equipped CPU modules of type DCP 10.
(TASKLIST)	Task list	New object for separating the system tasks and user tasks.
(TASK/RED)	Redundant task	All subsidiary program lists and sequential function charts within this task are implemented with redundancy.



The resource needs to be initialized if boot parameters such as the number of objects, PRAM, I/O protocol etc. are changed in the resource header.

### Separating the user tasks from the system tasks

Additional structural elements have been introduced to make the project tree clearer.

**USRTask** User tasks can be set up under this node.

**SYSTask** Used for recording predefined system tasks. No other user-defined tasks can be set up under this node.

### Renaming display names globally

When a display name is changed in the project tree and confirmed by the user, that name is replaced throughout the project.

### 1.3.2 FBD program

#### Signalling other objects

When FBD objects are positioned above or below the visible portion of the screen, arrows pointing up and/or down appear on the scroll bar.

### 1.3.3 Ladder Diagram

The new programming language, Ladder Diagram, has been introduced. Ladder Diagram is a graphically-oriented programming language complying with IEC 1131-3.

The LD language comes from the field of electromagnetic relay systems and describes the flow of current through the individual networks of the program organization units (POU) of a PLC.

### 1.3.4 Tag list

#### Configuring the plant areas

In column **A** of the tag list a plant area can be assigned to each tag. A maximum of 15 plant areas (A...O) can be allocated.

#### Unused points of measurement displayed in red

Unused points of measurement are now displayed in **red** in the tag list (corresponding to unused variables in the variable list). Unused points of measurement can be selected and then deleted.

### 1.3.5 Variable list

In the **search filter** for the variable list, the unused variable names and the system variables can be shown and hidden.

The variable list can be **sorted by slot/channel**.

**New system variables** for redundant process stations.

Gross Automation, 1725 South Johnson Road, New Berlin, WI 53146, [www.ssacsales.com](http://www.ssacsales.com), 800-349-5827

### 1.3.6 Hardware structure

#### Preserving the allocation of resources and hardware

The resource allocation of a process variable is retained when the I/O channel allocation is deleted or changed.

#### Transmission rate on the station bus $\neq$ rack ID

From version 3 onwards the rack ID is no longer displayed in hardware structure to the left of the unit (previously rack ID).

#### Slot for the DCP10 in rack DRA 01 and 02

In a type DRA 02 rack the redundant CPU modules may be installed in slots 0 and 1 in the main rack, and in slot 0 in the supplementary racks.

A rack DRA 01 with a hardware index of 04 or above is capable of redundant operation, but with this rack the redundant CPU modules must be installed in slot 0 in the central unit and slot 0 in the supplementary racks.

#### Defining parameters for the link module DLM and the CPU module

In V3, parameters must be defined for the link modules. The label facilitates error diagnosis since the label and the associated short and long texts can be reported.

#### State information of the I/O modules

State information has been added in the I/O channel allocation for the link module and also for the CPU module; it should be evaluated using variables.

### 1.3.7 Commissioning

#### Uploading the current block parameters

The current working data from blocks can be modified using write utilities from commissioning, from DigiVis or via a gateway (e.g. DDE server). More than one item of working data can be saved using the Upload parameter function.

#### Rapid switch from commissioning to configuration mode

It is possible to switch rapidly from commissioning to configuration mode from an FBD program, IL program or SFC program, or from the hardware structure (system structure and station view), variable list and tag list.

#### New resource states

The transitions of the resource from **stopped** to **running** and vice versa are now represented by the states **starting** and **stopping**. These states are only visible when the processing of a task takes a considerable length of time.

#### Error handling at task level

The error message "Execution\_abort\_error" is sent if the task takes longer than 10 s to calculate the programs in this task once.

### State display of tasks in the project tree

**Partially running** has been added to the states displayed for a resource or task. The state 'running' is only adopted if all assigned program modules are being processed.

### Trend window and value window

The procedure for configuring the trend and value window has been improved for the user by storing configurations, a simplified way of deleting an entire configuration, and the retention of values when changes are made.

### Long-term error buffer on the process station

Each occurrence of an error on a process station is logged, which means that it is possible to analyze the cause of the error.

The content of the compiled error list is transferred to an ASCII file on the operator station (DigiVis PC). The error log is automatically read by DigiVis and written to an ASCII file when either of the following two alarms occurs: 'DPS boot: Self monitoring' or 'DPS boot: Fatal error'.

### 1.3.8 Trend

The color selection for the foreground, background and windows of the trend display (3 colors) and for each curve can be configured.

It is also possible to choose whether max. 3 or all trends in the value region of the trend display are shown.

### 1.3.9 Logs

A file run time > 24.8 days can now be configured in operation logs.

The plant areas in the **signal sequence log** can be shown and hidden.

### 1.3.10 Graphic editor

#### Import of bitmaps for excellent graphic displays on the operator station

#### Improved zooming in the graphic editor

The portion for display can be selected immediately after choosing *Zoom level ...* from the menu.

Selection of the current section can also be initiated using the right mouse button if the graphic editor is in "selection mode" (this does not apply in parameter definition masks).

#### Mandatory parameters in the graphic editor

The mandatory parameters in the masks of the graphic editor are now displayed in **red**, as per the general convention.

Gross Automation, 1725 South Johnson Road, New Berlin, WI 53146, [www.ssacsales.com](http://www.ssacsales.com), 800-349-5827

### 1.3.11 General improvements

#### **Selection of cross references from editors**

Cross references can now be selected direct from each editor (apart from the graphic editor).

#### **Switching to error locations after the plausibility check**

You can switch directly from the plausibility list to the point at which an error has been detected.

#### **Classes of plausibility errors**

Hints have been added to complement the plausibility check messages (errors and warnings).

#### **Storing the latest editing position**

Many editors now store the latest editing position.

#### **Return to project tree, expanding branches automatically**

If a program is quit via *Exit*, then the edited program is selected in the project tree. If the edited program is in a "contracted" tree structure, the project tree is "expanded" accordingly. This facilitates navigation in the project tree.

#### **Global cancel/exit from nested parameter input dialogs**

A parameter input operation that extends over more than one dialog mask can now be cancelled completely with CANCEL or completed fully with OK. Scrolling in such masks can only still be performed using the >>/<< buttons.

#### **Improved tag selection**

The tag selection dialog has been improved. The requisite tags can now be found and selected more quickly by entering a search string.

#### **Global renaming of tags**

When a tag name is changed in the tag list and confirmed by the user, all instances of that tag name are replaced.

## 1.4 Function blocks

### 1.4.1 Send/receive blocks

Send and receive blocks enable communication to take place with any other TCP/IP subscriber on an external computer, as well as communication between Freelance 2000 systems. These blocks use DigiNet S (Ethernet) as the transmission medium, and the protocol used is the TCP/IP sub-protocol UDP or, as an alternative option, UDP Broadcast.

All data types and data structures from Freelance can be transferred.

### 1.4.2 Analog blocks

#### Counter with analog input CT\_ANA

The basic value can only still be configured and used within the measuring range. In order to achieve greater precision the counter works internally with 64-bit REAL resolution.

#### Set point controller C\_ANA

The manual value is checked in conjunction with the weighting adjustment parameter. The bias can only be configured and operated within the range [-scale range .. +scale range]. In automatic operating mode the manual value is not allowed to be written.

#### Analog input converter AI\_TR

The plausibility check only still checks the default values (pins **DVL** and **DVH**) where these have been input in the parameters mask.

#### Analog input converter with transient evaluation AI\_TRT

The plausibility check only still checks the default value (pin **DV**) where this has been input in the parameters mask.

Transient monitoring (ON/OFF) can only still be changed by modifying the configuration.

#### Time scheduler TS

At each coincidence point with a run time  $\geq$  task cycle time, a coincidence point value within the scale range must be entered.

At each coincidence point with a coincidence point value, a valid run time must also be entered. When the error output **ERR** is set, the values on state output **STA** are redefined.

#### Time analog filter TFILT

The function of this new block is to filter analog signals with high resolution. It implements the functions dead time, mean time value, minimum time value and maximum time value.

### 1.4.3 Binary blocks

#### Operating time counter CT\_LT

To enable the counter to produce a 'genuine' count of operating hours, the facility has been provided for configuring a time resolution (second;minute;hour). In each different time resolution the counter operates with a precision to the nearest millisecond.

#### Frequency analog converter FAC\_D

The minimum frequency configured is observed in all cases at the block output.

#### 1.4.4 Controller blocks

##### All controllers

Set point ramps and correction value ramps can be activated in configuration either only in automatic mode or in both manual and automatic modes. Previously they were always active both in manual and automatic mode.

The set point limits are supplied dynamically with the values of the measuring range limits while a controller is being configured. However, these can be overwritten with valid values at any time. If the measuring range limits are reduced, then the set point limits follow suit.

Empty fields are no longer accepted as parameter entries for set point limits and correction value limits, unless they are interconnected.

The effectiveness of the output limits with continuous controllers and point controllers has been overhauled.

##### Universal controllers and ratio controllers

The disturbance variable feedforward range for a DTB input has been expanded to -100.00.0% ... 100 .0%.

New parameter, "Disturbance variable monitoring", for specifying whether disturbance variables should be taken into account when displaying and checking the limit values of X, XD.

The disturbance variable processing for X has been corrected so that when the time function is configured to X (D-action and/or P-action effective to X) and disturbance variable DTB to X, a changed response may result.

##### Step controller C\_SS, C\_SU, C\_SR

The maximum task cycle time, in which the controllers concerned can be calculated, may not exceed 24 h.

The allowable ranges for motor positioning time, minimum pulse length, minimum pause length and dead time external feedback are restricted.

End position tracking (0%, 100%) even when no external feedback is available. Tracking of whichever values are of interest with internal feedback configured and external feedback available. State output STR is only still dependent on the value of the TRC input.

Secure control of end positions in automatic mode with external feedback. The continuous output variable has hitherto been limited to 0%..100% to make it impossible to completely reach end positions subject to min. pulse duration and motor positioning time. Steps have now been taken to ensure that the controller outputs remain set for as long as the target output variable required is  $\geq 100\%$  or  $\leq 0\%$ .

The control functions "Hold value", "Disable direction OPEN", "Disable direction CLOSED" are now also available with internal feedback.

Parameter for external feedback (mask 2) can now also be written in commissioning, with the result that it is possible to switch between internal and external feedback in commissioning.

##### Three-position controller C\_OS, C\_OU, C\_PU

The maximum task cycle time, in which the controllers concerned can be calculated, may not exceed 24 h.

### 1.4.5 Acquisition blocks

#### **Disturbance course log acquisition block DISLOG**

When the error output **ERR** is set, the values on state output **STA** are redefined.

#### **Trend acquisition block TREND**

When the error output **ERR** is set, the values on state output **STA** are redefined.

### 1.4.6 Monitoring blocks

#### **Connection monitoring M\_CONN**

The station type is no longer required for connection monitoring.

 It is not possible to monitor the connections to other process stations.

### 1.4.7 Control blocks

#### **Individual drive function for unidirectional drives IDF\_1**

End position monitoring is carried out irrespective of whether or not runtime monitoring is activated. The state local ignores any end position errors that may occur and resets them.

End position monitoring can be switched on and off.

The buttons for the correction command in the faceplate and in the operator line indicate the state at the block's output rather than the state of the input as before.

#### **Individual drive function for bidirectional drives IDF\_2**

End position monitoring is carried out irrespective of whether or not runtime monitoring is activated. The state local ignores any end position errors that may occur and resets them.

End position monitoring can be switched on and off.

A new response has been implemented for motors that are to be controlled. The type of device to be connected can be configured. In the faceplate, run direction STOP is signified by the STOP indicator flashing.

A dead time can be configured for the feedback inputs **FB1** and **FB0**.

The buttons for the correction command in the faceplate and in the operator line indicate the state at the block's output.

#### **Individual drive function for servodrives IDF\_A**

End position monitoring is carried out irrespective of whether or not runtime monitoring is activated. The state local ignores any end position errors that may occur and resets them.

End position monitoring can be switched on and off.

A dead time can be configured for the feedback inputs.

The buttons for the correction command in the faceplate and in the operator line indicate the state at the block's output.

**Analog dosing circuit DOS\_A**

In order to achieve greater precision the counter now works internally with 64-bit REAL resolution.

**Extended dosing circuit DOS\_E**

In order to achieve greater precision the counter now works internally with 64-bit REAL resolution.

**1.4.8 DigiBatch blocks**

**PLI function block**

Enhanced PLI function block (Phase Logic Interface) for interfacing to DigiBatch.

**1.4.9 Remote control blocks**

The blocks in the remote control library enable Freelance 2000 to be coupled with external systems. Interfacing is carried out over the serial interface of the CPU module (only DCP 10). The remote control protocol used is that conforming to IEC 870-5.

**1.4.10 Sartorius balance block**

For standard interfacing of Sartorius balances (not balance controllers), a block is provided with its own faceplate. The balance is linked by means of the communications module DCO01.

## 2 Changes in Version 3.3

### 2.1 General

#### 2.1.1 Process stations and racks

Under version 3.3 up to 10 process stations can be used (previously 5) with up to 5 racks (previously 3). A process station thus consists of one central unit and up to 4 I/O units.

The **D-GS resource** (gateway) has been reworked and is now also available as a redundant gateway.

#### 2.1.2 Modules

##### Frequency input module DFI 04

Processing of frequencies and pulses in the operating modes: Dosing circuit, event counting, frequency measurement, period duration measurement, pulse width measurement.

#### 2.1.3 OPC

The **OPC server** has been newly implemented. It enables data to be made available to a third-party OPC client.

**OPC client** functionality has been added to DigiVis. Data from a third-party OPC server can thus be processed in DigiVis.

Variables that are to be routed to the Freelance 2000 system via an OPC server are declared to the system using this function. These variables are not assigned to any process station, but remain assigned to the resource of type OPC server.

#### 2.1.4 Licensing - Safety mode

With a software update/upgrade or a license upgrade you receive a new authorization code that can be loaded into the Freelance 2000 system.

If a hard key is defective, the software goes into a safety mode (emergency mode). As with demo mode, Freelance 2000 can be used in this mode for 100 days. Within these 100 days the hard key can be replaced without affecting license rights.

If the hard key is removed after startup, then the software will likewise go into safety mode.

## 2.2 DigiVis

### **Messages with sound files in wave format**

For each process message, a separate sound file can be configured and played back over the sound system on the DigiVis PC.

### **Switch off horn**

The control room horn can be switched off in the message line by means of the HORN button. The configuration determines whether the field horn is switched off along with the control room horn, or whether it is controlled via a configured binary signal.

### **Overview display**

It is now possible to configure a free graphic as an overview display instead of the standard overview display. The display has been adjusted to conform with the new system limits.

### **System error messages**

System error messages have been added and updated.

## 2.3 DigiTool

### 2.3.1 Post-loadable libraries

When the project is opened, block versions are brought in line with one another. Only the classes of block used in a project are loaded onto the process station. This results in a greater amount of memory remaining available on the process station for the user programs. Classes of block required later can be post-loaded.

### 2.3.2 Project tree

#### **Adjustments for time zone / daylight saving changes**

The time zone and daylight-saving time are set for the entire project in the project tree, project element Configuration (CONF). This is now structured in the same way as the equivalent setting in Windows NT.

#### **Redundancy memory**

Redundancy memory has been added to the boot parameters of a redundant process station. This is the area of memory that is reserved for the transfer of redundancy data.

#### **Structuring of the operator station**

The "structure node" has been added to the operator station level. This project element allows the displays and logs on the operator station to be organized more clearly.

### 2.3.3 Variable list

The block of selected variables in the variable list is calculated via a task's process image.

### 2.3.4 Tag list

The tags exported from a tag list can be re-imported.

When importing tags it is also possible to import into the tag list files created using external applications (e.g. Microsoft Excel).

### 2.3.5 Commissioning

#### Loading I/O module EPROMs

In order to avoid the extremely time-consuming replacement of EPROMs on the I/O modules and a shutdown of the plant, EPROMs can be programmed with the system up and running and without involving any changes to the I/O modules' outputs. However, this is only possible with relatively new I/O modules that are already equipped with flash EPROMs.

### 2.3.6 Graphic editor

#### Storing the graphic as a bitmap

A complete graphic can be stored as a bitmap, as can the separate objects that make up a graphic.

#### Self-animated object

In version 3.3 and later a new dynamic graphic object is available.

Up to 8 static graphic objects are displayed cyclically in quick succession. Binary process variables can be used to switch the alternating display on and off, to alter the speed of the display and to make the display invisible.

#### Enhanced buttons

Any button, be it a separate button or part of a button field, can be 'labeled' with any static graphic element in addition to text.

#### Color map

The colors available for graphic displays have been changed from version V3.3 onwards. This means that color sequences are now available, e.g. for 3D displays.

When older projects are imported, the colors available previously are translated as effectively as possible to the new color values. However, it is also an option to continue using the existing color values. The color map can be selected before a project is imported.

## 3 Changes in Version 4.1

### 3.1 General

#### 3.1.1 User-defined function blocks

The user-defined function blocks (UFB for short) provide the option of creating custom function blocks. Function blocks can thus be created that are tailored to the specific requirements of the field.

A distinction is made between classes and instances for working with UFBs.

The functionality and visual appearance of a UFB is determined by the user-defined function block class. This contains the entire program set up by the user with its functions, function blocks and variables, the faceplate, the parameter mask and the help text. UFBs can be locked by the user with a password.

To use a UFB class, instances of the class are formed. Each user-defined function block instance has one.

The user-defined function block faceplate is produced in the faceplate editor. The faceplate editor provides all the same functions as the graphic editor.

#### 3.1.2 Modules

##### Communication module DCO 01



Changes have been made to the EPROM of the DCO 01 which have made it incompatible. When changing from older versions, the EPROM of V4.1 must be used.

#### 3.1.3 DigiNet P

New terminating resistors for the CAN bus (100  $\Omega$  in place of 120  $\Omega$ ).

## 3.2 DigiVis

### System display

The system display features the same appearance as the hardware structure in DigiTool. The dialogs are made up of tabbed dialogs.

### Quick-select dialog

The quick-select dialog makes it possible to rapidly access favorite displays, trend displays, logs etc. The quick-select dialog can be called up as an additional window on the screen. This window contains a set of buttons. Each function can also be called via keyboard operation rather than using the buttons.

### **Video RAM**

DigiVis organizes the 5 most recently called displays in video RAM. It is now possible to move in both directions (forwards and backwards) within the video RAM. This video RAM can also be implemented in the form of ring memory.

### **Freelance 2000 logo**

The Freelance 2000 logo is displayed in the message line.

## **3.3 DigiTool**

### **3.3.1 Update**

#### **Names for variables and tags**

From version V4.1 variables and tags are no longer allowed to share the same names. When importing from earlier versions, any variable names that are the same as tag names are suffixed with “\_var”.

#### **Statistics blocks**

Interface blocks for serial communications are configured in the hardware structure in versions from V4.1. The statistical information is made available to the programs via pin-compatible statistics blocks. When importing from older versions, the interface blocks are automatically replaced by the corresponding statistics blocks.

### **3.3.2 Project tree**

#### **External time server**

Up to 3 external stations can be time-synchronized via an operator station or gateway station. A DCP gateway can time-synchronize other DCP gateways or Maestro UX stations.

#### **Cross references**

As well as branching to the points where a cross reference is used, it is now also possible to switch to the point at which a cross reference is defined.

#### **Plausibility check**

If a sequential function chart (SFC), a trend acquisition block (TREND), a disturbance course log acquisition block (DISLOG) or a time scheduler (TS) has no corresponding display, this will cause a plausibility warning to be generated.

### 3.3.3 Hardware structure

#### Tree view and graphic view

The hardware structure now has two views (tree view and graphic view) in which the hardware is configured.

#### Toolbar and shortcut menu

A toolbar and shortcut menus (right mouse button) have been added to the hardware structure.

#### I/O editor

The new I/O editor replaces the previous channel assignment.

All channels defined in the I/O editor can be used directly in the programs with <TagName>.<Channel name>.

#### Standard names

Each object is assigned a standard name when it is added to the hardware structure.

#### Station type

A new station type for redundant process stations has been added. The redundant connection is now added automatically.

### 3.3.4 Commissioning

*Load changed objects* has been optimized to give shorter load times.

### 3.3.5 Graphic editor

#### Increased flexibility of graphic objects

Bar graph, fill level and graphic symbol can also be scaled by means of variables.

A reference line has been introduced for bargraph and fill level.

Configured message texts can be displayed with the alphanumeric display.

All actions can now be performed with any graphic object (other than trend window).

#### Operator actions

The following operator actions can be performed: display selection, tag selection, writing of variables and acknowledgment of messages.

Any value can now be entered in DigiVis when writing to variables.

#### Message display

The state of one or more messages can now be displayed instead of the bit variables. The messages can be displayed in the Freelance 2000 message colors.

### **Message type symbol**

The graphic object message type symbol has been introduced for displaying the message types.

### **Macros**

The management of the macro libraries has been overhauled.

In addition to the previous static macros, dynamic macros can now also be created. Macros are created in macro mode in the graphic editor. All graphic objects can be used in macros. All attributes of the graphic objects can be defined as parameters for animating the macros.

### **Text**

Text for user faceplates can be selected from a text list.

### **3.3.6 Trend**

The description length for each trend has been increased to 16 characters.

## **3.4 Function blocks**

### **3.4.1 Monitoring blocks**

#### **Message block M\_GEN**

The message block is used to create messages with a predefinable message value. This block can also generate messages from external devices. For this purpose, an external time stamp can be attached. The acknowledge status of the message (coming and going) is available at outputs.

### **3.4.2 Interfacing Protronic / Freelance 100**

The integration of Protronic process controllers in Freelance 2000 is in the form of standard function blocks. It is implemented with the following blocks:

- Blocks for describing the communication interface.
- Block for describing the process controller as a communication subscriber.
- Block for describing the controller-specific functionality of a channel on the process controller with specific parameter masks in DigiTool and a dedicated faceplate in DigiVis.

The Protronic range comprises the process controllers Protronic 100/Freelance 100, Protronic 500 and Protronic 550. The process controllers can be operated as stand-alone instruments in the process environment, but also in an interconnected system with other Protronic controllers or interconnected with overlaid systems.

### **3.4.3 Interbus blocks**

The function of the Interbus blocks is to transfer I/O data to an Interbus link module. The transmission medium used by the blocks is DigiNet S (Ethernet). The Interbus link module is a complete Interbus master. All data types and data structures from Freelance 2000 can be transferred.

### **3.4.4 Description of the block parameters**

A description of the parameters has been added to manual DigiTool (2).

# 4 Changes in Version 5.1

## 4.1 General

### 4.1.1 FieldController

Version V5.1 provides a FieldController for interfacing field buses to Freelance 2000.

The FieldController operates like a conventional Freelance 2000 process station via DigiNet S (Ethernet). The Freelance 2000 I/O modules can be used via a CAN module. Remote I/Os can be connected using an open field bus standard such as Profibus DP. A combined total of up to 10 process stations and FieldControllers may be operated in a Freelance 2000 system.

The FieldController has a modular case that enables it to be operated without ventilation at temperatures between 0 and 60 °C. The power supply (115/230 V AC or 24 V DC) is a plug-in unit that is integrated in the case. The Ethernet interface is supplied by plug-in modules (10BASE2 or AUI). Up to 4 field bus modules can be connected per FieldController. There are modules available for Freelance 2000 CAN bus, Profibus DP and serial protocols (e.g. Modbus).

The FieldController complies with the following standards: CE mark, NAMUR-EMC, IEC950, IEC1010, IEC1131-2, CSA, CSA-NRTL.

### 4.1.2 Profibus

Freelance 2000 supports Profibus in the form of PROFIBUS-DP Master Class 1. Any kind of Profibus slave can be connected to Freelance 2000 and configured via a Profibus slave object. The Profibus slave object enables DigiTool to define parameters for any field device that can in principle be configured via Profibus.

Configuration and parameterization are based on the device master data (GSD file) that is imported when a new Profibus slave object is imported. The data for the Profibus devices are treated exactly like I/O components. Custom parameter masks can be set up for the Profibus devices as for user-defined blocks.

Existing device configurations can be reused with the aid of a template manager.

### 4.1.3 Modules

#### Digital input module DDI 05

This module enables up to a maximum of 32 binary signals in a voltage range up to 120 V AC / 230 V AC to be linked.

## 4.2 DigiVis

### **Movable faceplates**

All the faceplates can be moved to any position within the display area of the monitor.

### **Messages with sound files**

There are three options for processing sound files in the operator station:

- a) priority-controlled, oldest message,
- b) priority-controlled, most recent message,
- c) time order.

## 4.3 DigiTool

### **4.3.1 Toolbars and context-sensitive menus**

Toolbars and context-sensitive menus (right mouse button) are now available in all areas of DigiTool.

### **4.3.2 FBS and KOP editor**

A shortcut menu in the FBD and KOP editor enables the operator to switch into a "line draw" mode.

### **4.3.3 Sequential function chart**

A tag can be allocated to each criterion in the criteria windows. In the SFC display, the faceplate for the assigned measuring point can be called up directly.

### **4.3.4 Variable list**

#### **Leading numbers**

Variable names are now also allowed to begin with numbers. However, the variable name must contain at least one letter.

#### **Channel assignment**

The assignment of variables to I/O channels can now also be edited in the variable list.

### 4.3.5 Hardware structure

#### Templates

Sections of the configured hardware structure can be stored in a template manager for reuse. Templates can be reused in the same project as well as in other projects.

#### Direct use of I/O components

I/O data from field devices can also be used directly in programs.

#### Objects

The hardware structure organizes the 20 most recently called objects in a cache. It is possible to move in both directions (forwards and backwards) within the cache.

#### Graphic views

Graphic views for the field bus configuration have been added to the hardware structure.

### 4.3.6 Commissioning

When the variables or terminals in a block are overrun in FBD and KOP programs, then the current values are displayed directly.

### 4.3.7 Documentation

#### Hardware documentation

The hardware can also be documented selectively. This means that only the selected sections can be documented in the hardware structure.

#### Cross references

Cross references in FBD programs are no longer documented. The documentation of cross references is performed through the cross reference list.

### 4.3.8 PLC export

An external tool (DELTAPLC) is available for PLC export; this determines the difference between two PLC files, and saves it in PLC format. In this way, just the changes in the configuration can be transferred to other systems (e.g. Maestro UX).

### 4.3.9 Display allocation

To a certain extent, display allocations can be specified automatically by the system. This serves to reduce the effort for configuration.

When display allocation is called up, and when a plausibility check is carried out on an operator station, the tags, displays and logs contained in the configuration database are checked. For each tag, usage is established in the instances of all display types and log types.

When usage of a tag is discovered, the display or log is indicated as an entry for the display allocation. If no usage and no cross reference is found, this is indicated by the entry "<undefined>".

## 4.4 Blocks

Several different sound files (WAV files) can be entered for each message for the purpose of sound output. The separate sound files are separated by a '+'. The input field remains the same length as before, i.e. 100 characters. Any number of sound files may be entered as long as they fit within the space available.

# 5 Changes in Version 5.2

## 5.1 General

### 5.1.1 Length of tag names

It is possible to use tag names that are 16 characters in length (KKS) rather than 12-character names. The standard tag name length is 12 characters; a special order must be placed in order to use 16-character tag names.

### 5.1.2 Profibus

Additional configuration and commissioning facilities have been provided for Profibus objects.

- For DPV1 parameters, data structures from the PROFIBUS-PA Profile for Process Control Devices can be used directly.
- The design of the combo boxes for creating the individual dialogs has been enhanced.
- The bus address and identification number of all slaves connected to the master can be read.
- The configuration data of the slaves can be displayed and read from the device.
- All configured DPV1 parameters can be read from the device and transferred to the project database.
- The system message for diagnostic values can be deactivated.

### 5.1.3 Sequence of Event Message

Selected binary signals can be logged with a time stamp in the correct chronological order.

For this purpose, the 'Time stamp' function is activated in module DDI 01. When this function is activated, an extra binary value and a time value are made available at the output of this hardware block in addition to the 'normal' 32 binary outputs. For each channel, when there is an edge change to the binary value on these outputs the current value and associated time (resolution = update cycle of module) are stored.

A sequence of events monitoring block M\_SOE is configured in an FBD program. This block can be used to transfer binary values with time stamps to a DDI 01 in a signal sequence log. For this purpose, when parameters are defined for this block a DDI 01 module is allocated and the channels that are to be monitored are defined.

With each cycle of the user task the binary values and time stamps of the DDI 01 module are read. When the stored binary values and their time stamps are read, the time stamp function is re-activated in the DDI 01 module, i.e. the next edge change of this binary value can be detected and stored.

Each time the function block M\_SOE detects a value change on a monitored channel, a sequence of event message is generated with the binary value and associated time stamp; this sequence of event message can be processed further in the signal sequence logs.

Sequence of event messages are not shown on the DigiVis message line or message page.

### 5.1.4 Function block diagram

The editor for function block diagram programs has been provided with a new interface. The draw area is now 10 x 10 pages. The lines for linking variables are no longer used. Instead, the editor now contains a freely-positionable element variable, as used already in the ladder diagram editor.

FBD programs from earlier versions of Freelance 2000 are converted automatically.

## 5.2 Freelance *select*

Freelance *select* with the FieldController *select* is a fieldbus starter kit aimed at providing customers with an initial entry into fieldbus technology. For an attractive entry-level price one receives a fully-functional FieldController with a Profibus module. The entire software package is also provided, thus enabling a Profibus line to be operated with a master and slaves.

### 5.2.1 Starter kit

The FieldController is equipped with a power supply (115/230 V AC), a 10Base2 Ethernet module and a Profibus module; the starter kit also includes a DigiTool license and a DigiVis license.

It also includes two hard keys and the appropriate authorization code for DigiTool *select* and DigiVis *select*. All the documentation is provided in the form of a CD.

Entry is made easier for the first-time user by a pre-prepared project which can be used to gain an understanding of the first steps in configuring a project.

### 5.2.2 Freelance *select* functional capabilities

The FieldController *select* has the same external design, but is identified by the system as FieldController *select*. In its standard form the FieldController *select* is provided with one Profibus *select* module, and can be upgraded with both a CAN-3 and a SERIAL fieldbus module. The Profibus *select* module can only be used in a FieldController *select*.

In total, DigiTool *select* supports the following:

- 1 operator station (DigiVis) and 1 gateway (DDE, OPC, Maestro)
- 2 user tasks
- 8 control circuits
- 20 trend data acquisition blocks

All other functions are the same as for a standard DigiTool license.

### 5.2.3 Freelance *select* upgrade

Freelance *select* can be upgraded to a full system, naturally using the FieldController *select*, which remains in the existing system.

 If, for example, the process stations, gateways and/or operator stations are upgraded, then both DigiTool *select* and DigiVis *select* must also be upgraded. When you order an upgrade, your license will also be upgraded .

## 5.3 Blocks

The new sequence of event monitoring block **M\_SOE** processes sequence of event signals (sequence of events).

### TCP/IP send and receive blocks

The TCP/IP interface is configured under Hardware Structure. New statistics blocks are provided for processing statistical data further in user programs.

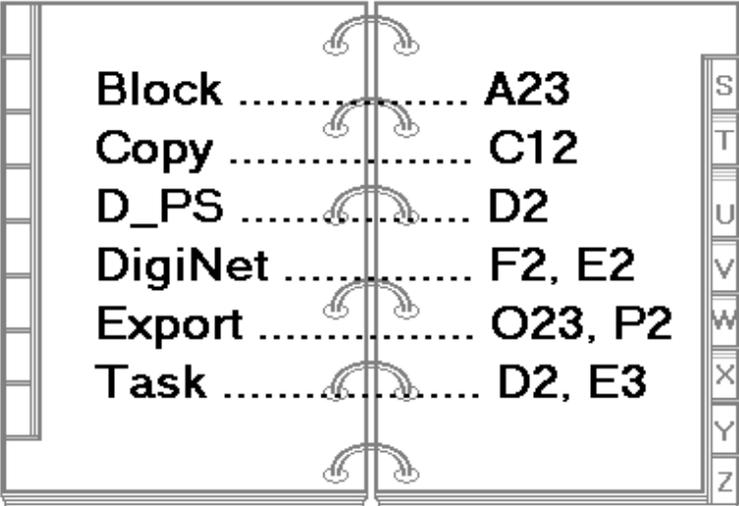
### Telecontrol blocks

The functionality of the telecontrol editor (export/import of the telecontrol list) has been transferred to the interface blocks.





# X Index



The image shows a stylized index page for the letter 'X'. The page is framed to look like an open book with a central binding and curved lines representing the pages. On the right side, there is a vertical index bar with letters S, T, U, V, W, X, Y, and Z. The main content lists terms and their corresponding page numbers:

Block .....	A23
Copy .....	C12
D_PS .....	D2
DigiNet .....	F2, E2
Export .....	O23, P2
Task .....	D2, E3



**A**

Access rights .....	D-22, D-23
Acknowledgement .....	A-19
Activate/deactivate .....	E-33
Adding components .....	E-29
Authorization disk .....	B-5, B-15
Automatic startup.....	D-43

**B**

Batch .....	A-11
Bit .....	E-38
bit length .....	E-38
Block	
export .....	D-21
import.....	D-21
import as redundant.....	D-21
Boot Parameters.....	A-18

**C**

Check .....	D-52
Check all.....	D-52
Clipboard	
viewer.....	E-22
Closing	
Project.....	C-18
Cold start task.....	D-45
Colors .....	D-57
Comment.....	D-55
Commissioning.....	F-5
Project tree.....	F-12
Commissioning/Configuration (changeover) .....	F-11
Common display pool.....	D-13
Communication links .....	D-26
configuration .....	D-25
Communications connections .....	E-34
Compress .....	D-18
Configuration .....	D-12
CONF .....	D-25
Configuring the communication links.....	D-25
Context menu .....	E-20
Creating New Projects.....	C-10
Cross references .....	E-24
Cycle time.....	D-43
Cyclical task.....	D-15, D-41

**D**

Data type .....	E-38
DCP gateway .....	D-34
DDE gateway .....	D-34
Default names	
Master objects .....	E-30
Modules .....	E-30, E-31
Remote process controller .....	E-31
Serial interface .....	E-31
Slave modules .....	E-30
Slave objects .....	E-30
Default task .....	D-15, D-41
Deleting	
Project .....	C-19
Demo mode .....	B-5
Detail view .....	E-18
Device view .....	E-17
D-GS resource .....	D-28, D-33
Diagnostic password .....	D-33
DigiBrowse .....	A-19
DigiCheck .....	B-24
DigiLock .....	C-22, D-22
DigiTool Configure .....	B-20
DigiTool Help .....	B-22
Display pool	
show target stations .....	D-41
Disturbance course log .....	D-14, D-49
DMSAPI .....	D-35
Document types .....	G-50
Documentation .....	G-6
Changing the column width .....	G-43
Cover page .....	G-15
Crossreference	
Tags .....	G-26
Variables .....	G-22
Display access .....	G-34
Documentation settings .....	G-17
Drawing footer .....	G-7, G-10
Drawing header .....	G-7
Footer title .....	G-11
Hardware scope .....	G-28
Hardware structure .....	G-28
Header title .....	G-9
Index .....	G-16
Manager .....	G-43
Menu Structure .....	G-44

Message processing .....	G-32
Object parameter .....	G-12
Plausibility check errors .....	G-33
Printer setup.....	G-42
Project tree.....	G-18
Scope.....	G-15, G-35
Sorting the output.....	G-36
Structured variables .....	G-33
Tags list.....	G-24
Variables list.....	G-20
D-OS resource.....	D-28
D-POOL Global display pool .....	D-28
D-PS resource .....	D-28
<b>E</b>	
Emergency mode .....	B-5
Error handling .....	D-47
Error list .....	D-55
Error task .....	D-47
Event log.....	C-21
Exiting DigiTool.....	C-20
Expand .....	D-18
Export	
Backup.....	C-17
block.....	E-22
Exported variables of the resource.....	F-18
<b>F</b>	
FBD program .....	D-15, D-49
FDT / DTM-Support.....	A-9
Field contents .....	G-59
Field names	
drawing footer .....	G-54
drawing header .....	G-54
Field titles .....	G-53
FieldController Redundancy.....	A-9
Flash rate.....	D-33
FTP access.....	B-9
Full expand .....	D-18

**G**

Gateway.....	A-16
DCP.....	D-34
DDE.....	D-34
OPC.....	D-35
TRN.....	D-35
UNI.....	D-35
Gateway station.....	D-13, D-33
redundant.....	D-13
General hints for operating.....	A-6
Global display pool D-POOL.....	D-28
Global variables of the resource.....	F-17
GMT.....	F-22
Graphic display.....	D-14, D-49
Graphic view.....	E-11
Graphics.....	A-18
Group display.....	D-14, D-49

**H**

Hardcopy.....	D-56
Hardware structure.....	E-5
Hart Support.....	A-9
Hotline.....	B-5

**I**

I/O Editor.....	E-36
IL program.....	D-15, D-49
Import	
Backup.....	C-15
block.....	E-23
Project.....	C-14
Incorrect objects.....	D-52
Initialize	
all.....	F-32
Insert.....	E-25
Insert above.....	D-17
Insert below.....	D-17
Insert documentation job.....	G-45
Insert next level.....	D-17
Installation	
Upgrade.....	B-18
IP address.....	E-44

**K**

Keys (recurring) ..... A-7

**L**

LatCRcv task ..... D-47

LatCSnd task ..... D-47

Lateral communication ..... D-47

LD program ..... D-15, D-49

Link configuration ..... D-25

## Load

    Message configuration ..... F-27

    Project ..... F-25

    selected objects ..... F-28

    Variables ..... F-27

    Whole station ..... F-26

Local time ..... F-22

**M**

Maestro CS resource ..... D-28

Maestro NT ..... A-10, D-35, D-37

Maestro NT as operator station ..... D-12

Maestro RTDS ..... D-38

Maestro UX ..... D-34

## Memory

    Requirements ..... C-14

Menu structure ..... D-9

Multitasking ..... D-43

**N**

Network protocol ..... B-8

New print job ..... G-14

Nodes ..... D-10, D-18

Number of Logs ..... A-14

**O**

## Objects

    moving, deleting ..... D-19

Objects of the resource ..... F-19

Once-only task ..... D-15, D-41

OPC gateway ..... D-35

OPC server ..... D-13, D-39

OPC server resource ..... D-28

## Open

    Project ..... C-12

Operating system .....	B-5
Operation log .....	D-14
Operation with keyboard.....	A-6
Operation with mouse.....	A-6
Operation with mouse or keyboard.....	A-6
Operator plant log.....	D-49
Operator station.....	D-12
Operator station programs	
displays and logs .....	D-49
Overload .....	D-43
Overview display.....	D-14, D-49

## P

Parameter	
upload .....	F-29
P-FB pool of user FBs .....	D-28
PhaseX .....	A-11
Ping.....	B-9
Plausibility check .....	D-53, E-23
PLC mode.....	D-43
Pool.....	D-13, D-22
Pool of user FBs P-FB.....	D-28
Print .....	A-19, G-40
Print job.....	G-6
Process image.....	D-43
Process station .....	D-12, D-28
redundant.....	D-12
Processing sequence .....	D-48
Profibus	
view .....	E-16
Program	
information .....	F-21
Program list.....	D-15, D-48
Program-documentation .....	G-18
Programs, displays and logs .....	D-49
Project	
Comment.....	C-20
Comment.....	C-20
Header .....	C-19
version.....	F-16
Project documentation.....	D-52
Project objects .....	D-10
Project tree	
search .....	D-18

**R**

RedSt task .....	D-47
Redundancy Link Module RLM 01 .....	A-9
Redundant	
process station .....	D-12
task .....	D-15, D-41
Redundant gateway station .....	D-13
Resource	
AC800F .....	D-28
D-GS .....	D-28
D-OS .....	D-28
D-PS .....	D-28
FieldController .....	D-28
gateway station .....	D-33
GWY .....	D-33
Maestro CS .....	D-28
MAESTRO-CS .....	D-37
OPC server .....	D-28, D-39
process station .....	D-28
user-defined function blocks .....	D-40
Resource allocation .....	E-32
RTDS	
Maestro NT .....	D-38
Run task .....	D-46

**S**

Save	
Project .....	C-19
Screen Resolution .....	A-12
Search	
project tree .....	D-18
Sequential function chart display .....	D-14, D-49
Services .....	B-9
Set system time .....	F-23
SFC program .....	D-15, D-49
Show target stations .....	D-41
Side effects .....	D-10, F-15, F-28, F-33, F-34
Signal sequence log .....	D-49
Signal sequence log 1 .....	D-14
Signal sequence log N .....	D-14
Standard names	
Hart channel .....	E-31
Starting	
DigiTool Setup .....	B-10
Project .....	C-6

State line.....	D-8
long .....	D-56
Project tree.....	D-8
Station display of a redundant FieldController .....	E-15
Station numbers.....	B-13
Station view .....	E-13, E-14
Stop task.....	D-47
Structure node .....	D-14, D-50
Summertime .....	D-26
Synchronize system time.....	F-22
System	
Limits.....	A-10
object.....	E-46
Time .....	F-22
view .....	E-12
<b>T</b>	
Task.....	D-15, D-41
cold start.....	D-45
cyclical.....	D-15, D-41
default .....	D-15, D-41, D-43
error.....	D-47
LatCRcv .....	D-47
LatCSnd .....	D-47
once-only.....	D-15, D-41
redundancy start .....	D-47
run .....	D-46
stop .....	D-47
system.....	D-45
warm start .....	D-46
Templates Module .....	A-9
Time scheduler display.....	D-14, D-49
Time zone .....	D-26, F-22
Toolbar.....	A-13
Button.....	C-26
Tree view .....	E-8
Trend	
Display .....	D-14, D-49
Gateway .....	D-35
Server.....	A-11
Window .....	F-42
TRN gateway.....	D-35
True Color.....	A-12, C-25
Type ahead.....	A-15
Types of scripts .....	A-5
Types of symbols .....	A-5

**U**

UNI gateway .....	D-35
Upload .....	F-29
User function block pool .....	D-13, D-40
User groups .....	D-24
configuration .....	D-24
User tasks.....	D-41
User-defined function blocks .....	D-40

**V**

Value	
window .....	F-41
Variable .....	E-39
window .....	F-38
Variables and Tag List.....	A-15
Variables for drawing footer .....	G-56
Version	
check.....	F-16
number.....	F-16
Virtual keys .....	A-8
Virtual memory size .....	B-9
Visual orientation hints .....	A-5

**W**

Warm start task .....	D-46
Wave files .....	B-12, B-21



Gross Automation, 1725 South Johnson Road, New Berlin, WI 53146, [www.ssacsales.com](http://www.ssacsales.com), 800-349-5827