

## SIMATIC HMI

### ProTool V6.0 Service Pack 2 How to configure Windows- based systems

#### User Manual

Order number  
6AV6594-1MA06-1AB0

Preface, Contents	
Introducing ProTool	<b>1</b>
Getting started in ProTool	<b>2</b>
ProTool integrated in SIMATIC STEP 7	<b>3</b>
Creating and editing projects	<b>4</b>
Configuration techniques	<b>5</b>
Testing projects	<b>6</b>
Documenting and managing projects	<b>7</b>
Appendix, Index	

## Safety Guidelines

This manual contains notices which you should observe to ensure your own personal safety, as well as to protect the product and connected equipment. These notices are marked as follows according to the level of danger:



---

### Danger

indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury.

---



---

### Warning

indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury.

---



---

### Caution

used with the safety alert symbol indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury.

---

---

### Caution

used without safety alert symbol indicates a potentially hazardous situation which, if not avoided, may.

---

---

### Attention

indicates that unwanted events or status can occur if the relevant information is not observed.

---

---

### Note

draws your attention to particularly important information on the product, handling the product, or to a particular part of the documentation.

---

## Qualified Personnel

Only **qualified personnel** should be allowed to install and work with this software. Qualified personnel within the meaning of the safety notices of this documentation are persons capable of giving due consideration to safety engineering standards when using this software on systems..

## Trademarks

The registered trademarks of the Siemens AG can be found in the preface.

## Impressum

Editor and Publisher: A&D PT1

### Copyright Siemens AG 2002 All rights reserved

The transmission and reproduction of this documentation and the exploitation and communication of its contents are not allowed, unless expressly granted. Violators are liable for damages. All rights reserved, especially in the case of the granting of a patent or registration by GM.

Siemens AG  
Bereich Automation & Drives  
Geschäftsgebiet SIMATIC HMI  
Postfach 4848, D-90327 Nuernberg

Siemens Aktiengesellschaft

### Exclusion of liability

We have checked the content of this publication for compliance with the described hard and software. However, discrepancies cannot be excluded, with the result that we assume no guarantee for total compliance. The information in this publication is checked regularly, and any necessary corrections are included in the following editions. We would be grateful for any suggestions for improvement.

© Siemens AG 2002  
Technical data subject to change.

# Preface

## Purpose

This document forms part of the documentation package for the ProTool CS configuration software. It provides all information required for the creation and configuration of projects.

## Notation

Different font formats simplify orientation within the text:

“Output”	Names of dialog boxes, fields and buttons are shown in quotation marks.
“File” > “Edit”	Menu items are shown in quotation marks. Successive submenus/options are separated by arrows. The complete route to the menu option in question is always shown.

## History of the documentation

This documentation describes the configuration of Windows-based systems with ProTool.

The various issues of the documentation correspond to the following product versions:

Release 07/98	Configuring Windows-based systems. Valid for ProTool/Pro CS 5.0 or higher.
Release 01/99	Inclusion of MP 270 and editorial reworking of manual. Software runs under Windows 95, Windows 98, Windows NT 4.0 and higher. Valid for ProTool/Pro CS 5.1 or higher.
Release 12/99	Extensions and inclusions of new operating units. Software runs under Windows 95/98, Windows 2000 and Windows NT 4.0 and higher. Valid for ProTool/Pro CS 5.2 or higher.
Release 07/00	Extensions, new operating units and new PLC drivers. Applies to ProTool/Pro CS version 5.2 or higher plus Service Pack 2
Release 03/01	Incorporation of MP 370 and SIMOTION PLC drivers. Applies to ProTool/Pro CS version 5.2 or higher plus Service Pack 3

Release 12/01	Valid for ProTool CS 6.0 or higher.
Release 12/02	Valid for ProTool CS 6.0 Service Pack 2 or higher.

## Trademarks

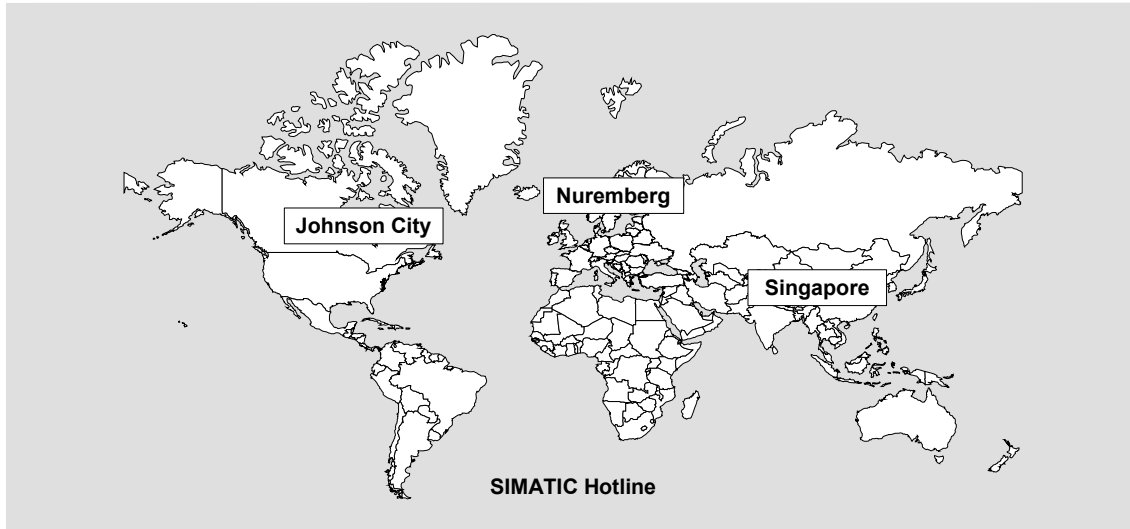
The following names are registered trademarks of the Siemens AG:

- SIMATIC®
- SIMATIC HMI®
- SIMATIC Multi Panel®
- SIMATIC Multifunctional Platform®
- SIMATIC Panel PC®
- HMI®
- ProAgent®
- ProTool®
- ProTool/Lite®
- ProTool/Pro®

The remaining trademarks in this publication may be trademarks, whose use by third parties for their own purposes could violate the rights of the owner.

## Customer and Technical Support

Available round the clock, worldwide:



<b>Worldwide (Nuremberg)</b> <b>Technical Support</b> (FreeContact)  Local time: Mon.–Fri. 8:00 to 17:00 Telephone: +49 (180) 5050-222 Fax: +49 (180) 5050-223 E-Mail: techsupport@ad.siemens.de GMT: +1:00	<b>Worldwide (Nuremberg)</b> <b>Technical Support</b> (fee-based, only with SIMATIC Card)  Local time: Mon.–Fri. 0:00 to 24:00 Telephone: +49 (911) 895-7777 Fax: +49 (911) 895-7001 GMT: +01:00	
<b>Europa/Africa (Nuremberg)</b> <b>Authorization</b>  Local time: Mon.–Fri. 8:00 to 17:00 Telephone: +49 (911) 895-7200 Fax: +49 (911) 895-7201 E-Mail: authorization@nbgm.siemens.de GMT: +1:00	<b>America (Johnson City)</b> <b>Technical Support and Authorization</b>  Local time: Mon.–Fri. 8:00 to 19:00 Telephone: +1 423 461-2522 Fax: +1 423 461-2289 E-Mail: simatic.hotline@sea.siemens.com GMT: -5:00	<b>Asia/Australia (Singapore)</b> <b>Technical Support and Authorization</b>  Local time: Mon.–Fri. 8:30 to 17:30 Telephone: +65 740-7000 Fax: +65 740-7001 E-Mail: simatic.hotline@sae.siemens.com.sg GMT: +8:00

The languages spoken by the SIMATIC Hotlines are generally German and English – the Authorization Hotline is also provided in French, Italian and Spanish.

## SIMATIC Customer Support Online Services

The SIMATIC Customer Support team offers you substantial additional information about SIMATIC products via its online services:

- General current information can be obtained
  - In the Internet under  
<http://www.siemens.com/simatic>
- Current Product Information leaflets, FAQs (Frequently Asked Questions), Downloads, Tips and Tricks can be obtained
  - In the Internet under  
<http://www.siemens.com/automation/service&support>

## Training Center

Siemens offers a number of training courses to familiarize you with the SIMATIC S7 automation system. Please contact your regional training center or our central training center in D 90327 Nuremberg, Germany for details.

Telephone: +49 (911) 895-3200

Internet: <http://www.sitrain.com>

E-Mail: [info@sitrain.com](mailto:info@sitrain.com)

## Other Sources of Assistance

In case of technical queries, please contact the Siemens representatives in the subsidiaries and branches responsible for your area.

The addresses can be found:

- In the Siemens Catalog ST 80
- In the Internet under  
<http://www.siemens.com/automation/partner>
- In the Interactive Catalog CA01  
<http://www.siemens.com/automation/ca01>

# Contents

<b>1</b>	<b>Introducing ProTool.....</b>	<b>1-1</b>
1.1	Overview of the documentation .....	1-4
1.1.1	ProTool for the whole SIMATIC HMI device family .....	1-5
1.1.2	ProTool for Windows-based systems .....	1-6
1.1.3	ProTool for graphical displays .....	1-8
1.1.4	ProTool for text-based displays .....	1-9
<b>2</b>	<b>Getting started in ProTool.....</b>	<b>2-1</b>
2.1	Starting ProTool.....	2-2
2.2	Configuring Windows-based systems .....	2-3
2.3	Notes on the configuration computer.....	2-4
<b>3</b>	<b>ProTool integrated in SIMATIC STEP 7 .....</b>	<b>3-1</b>
3.1	How to start ProTool in SIMATIC STEP 7 .....	3-3
3.2	SIMATIC Manager .....	3-6
3.3	Working with SIMATIC Manager .....	3-7
3.4	Changing the PLC and applying tags .....	3-8
3.5	How to integrate and copy STEP 7 projects.....	3-9
3.6	Connecting to STEP 7 Lite .....	3-10
3.7	How to restore the symbolic tag connections .....	3-11
<b>4</b>	<b>Creating and editing projects .....</b>	<b>4-1</b>
4.1	Fundamental considerations when creating a project.....	4-2
4.2	Components of a ProTool project.....	4-4
4.3	Objects in the project window .....	4-5
4.4	Steps to be taken when creating a project .....	4-6
4.5	Configuring behavior at run time .....	4-7
4.6	PLC drivers .....	4-9
4.7	Area pointers .....	4-11
4.8	Screen partitioning on the operating unit.....	4-14
4.9	Configuration notes for touch devices .....	4-17
4.10	Functions to provide support for service work .....	4-19
4.11	Editing projects .....	4-20
4.11.1	Convertible projects .....	4-21
4.11.2	Key assignment when converting.....	4-23
4.11.3	Example of how to convert an MP 370 project.....	4-26
4.11.4	Copying objects: Between projects and within a project .....	4-27
4.11.5	Replacing a project and project parts .....	4-30
4.11.6	Undoing and redoing actions .....	4-33
4.11.7	Undoing the last action .....	4-35
4.11.8	Redoing the last action .....	4-35
4.12	Retrieving project information .....	4-36
4.12.1	Display in "Cross-Reference" window .....	4-36
4.12.2	Display at "Project Information" .....	4-37

<b>5</b>	<b>Configuration techniques.....</b>	<b>5-1</b>
5.1	Create screens.....	5-2
5.1.1	Screen objects in ProTool.....	5-5
5.1.2	Libraries in ProTool.....	5-10
5.1.3	SIMATIC HMI Symbol Library.....	5-12
5.1.4	Colors of screen objects.....	5-13
5.1.5	Fonts.....	5-15
5.1.6	Custom configuration user interface in ProTool.....	5-16
5.1.7	Date/Time.....	5-17
5.1.8	Examples of how to create screens.....	5-18
5.1.8.1	How to paste a coolant pump in a screen.....	5-19
5.2	Configuring display elements and controls.....	5-20
5.2.1	Output fields (text/graphic list).....	5-22
5.2.2	Input fields (selection field).....	5-24
5.2.3	Input fields for date and time.....	5-26
5.2.4	Input fields for concealed password entry.....	5-27
5.2.5	Function keys.....	5-28
5.2.6	Trend view.....	5-30
5.2.7	Bit trigger functions.....	5-30
5.2.8	Bar graph.....	5-31
5.2.9	State areas.....	5-32
5.2.10	Buttons.....	5-34
5.2.11	State buttons.....	5-35
5.2.12	Using buttons as direct keys.....	5-37
5.2.13	Special features with direct keys.....	5-38
5.2.14	Switches.....	5-40
5.2.15	Slider controls.....	5-42
5.2.16	Analog display.....	5-44
5.2.17	Date/time display.....	5-46
5.2.18	Digital/analog clock.....	5-47
5.2.19	Configuring the date and time for Panels.....	5-48
5.2.19.1	How to display the date and time on the TP 170A.....	5-49
5.2.19.2	How to set the date and time on the TP 170A.....	5-55
5.2.19.3	How to synchronize the date and time with the PLC.....	5-62
5.2.19.4	How to configure the date and time for xP170B.....	5-66
5.3	Using tags.....	5-68
5.3.1	Properties of tags.....	5-71
5.3.2	Array tags.....	5-73
5.3.3	Tag settings.....	5-75
5.3.4	Updating tags.....	5-76
5.3.5	Address multiplexing.....	5-78
5.3.6	Multiplexing tags.....	5-79
5.3.7	Archiving tags.....	5-80
5.3.8	Functions to change tags.....	5-81
5.3.9	Functions to set tags bit by bit.....	5-82
5.3.10	Functions to change the value of a tag.....	5-83
5.3.11	Functions for the linear scaling of tags.....	5-84
5.3.12	Examples of configuring tags.....	5-85
5.3.12.1	How to configure a multiplex tag (address multiplexing).....	5-85
5.3.12.2	Structure of an archive for tags.....	5-87
5.3.13	Details on configuring tags.....	5-88
5.3.13.1	Saving STRING tags.....	5-88
5.3.13.2	Decimal places with tags.....	5-89



5.3.13.3	Address and type of a tag.....	5-91
5.4	Creating text or graphic lists.....	5-92
5.5	Creating graphics and text.....	5-93
5.5.1	Making optimum use of graphics.....	5-96
5.5.2	Static text.....	5-97
5.5.3	Vector graphic elements.....	5-98
5.6	Creating trends.....	5-99
5.7	Configuring schedulers.....	5-101
5.8	Creating reports.....	5-102
5.8.1	Printing messages in a report.....	5-106
5.8.2	Printing recipes in a report.....	5-107
5.8.3	Page numbers in reports.....	5-108
5.8.4	Example of creating reports.....	5-109
5.8.4.1	How to create a report.....	5-109
5.9	Messages.....	5-111
5.9.1	Displaying operating and process states.....	5-111
5.9.2	The components of a message.....	5-113
5.9.3	Message properties.....	5-115
5.9.4	Acknowledging messages.....	5-116
5.9.5	Settings for message classes.....	5-117
5.9.6	System messages.....	5-118
5.9.7	Logging messages.....	5-120
5.9.8	Displaying messages on the operating unit.....	5-121
5.9.9	Message view.....	5-124
5.9.10	Configuring message processing.....	5-126
5.9.11	Contents of message buffers.....	5-127
5.9.12	Communication areas for messages.....	5-129
5.9.13	Message procedure.....	5-130
5.9.13.1	Message bit procedure.....	5-131
5.9.13.2	Message number procedure ALARM_S.....	5-133
5.9.13.3	Display classes for ALARM_S messages.....	5-136
5.9.13.4	Setting the message procedure and selecting the display classes.....	5-138
5.9.13.5	Incorporating ALARM_S messages.....	5-141
5.9.13.6	Updating the operating unit.....	5-142
5.9.13.7	Resource consumption of messages.....	5-144
5.9.13.8	Communication sequence in the event of ALARM_S messages.....	5-145
5.9.13.9	Acknowledging ALARM_S messages.....	5-147
5.9.13.10	Printing ALARM_S messages.....	5-148
5.9.14	Examples of configuring messages.....	5-149
5.9.14.1	How to configure alarm messages.....	5-150
5.9.14.2	Message areas for SIMATIC S5.....	5-152
5.9.14.3	Message areas for SIMATIC S7.....	5-153
5.9.14.4	Examples of system messages.....	5-154
5.9.14.5	Event messages export file.....	5-155
5.10	Using functions.....	5-157
5.10.1	Events for triggering functions.....	5-159
5.10.2	Function parameters.....	5-161
5.10.3	Combining multiple functions.....	5-163
5.10.4	User-defined functions.....	5-164
5.10.5	Special features with conversion functions.....	5-165
5.10.6	Example of configuring a function.....	5-167
5.10.6.1	How to change the operating mode with a current display.....	5-168
5.11	Create archives.....	5-173
5.11.1	Archiving process data.....	5-174

5.11.2	Settings for an archive .....	5-177
5.11.3	Displaying archive data in a specified time window.....	5-178
5.11.4	Options for configuring archives .....	5-179
5.11.5	Examples of using archives .....	5-180
5.11.5.1	How to create an archive for messages .....	5-180
5.11.5.2	Structure of an archive for messages .....	5-181
5.11.5.3	Structure of an archive for tags.....	5-183
5.11.5.4	How to display archive data with a time window .....	5-184
5.12	Creating recipes .....	5-185
5.12.1	Using and configuring recipes .....	5-187
5.12.2	Recipe view.....	5-190
5.12.3	Recipe screen .....	5-193
5.12.4	Configuring recipes .....	5-194
5.12.5	Synchronization with the PLC.....	5-198
5.12.6	Editing data records .....	5-199
5.12.7	Structure of the "data mailbox" area pointer .....	5-200
5.12.8	Compatibility of recipes.....	5-201
5.12.9	Examples and scenarios for creating and configuring a recipe .....	5-202
5.12.9.1	How to create a recipe .....	5-203
5.12.9.2	How to configure a recipe view.....	5-205
5.12.9.3	Scenario 1: How to enter data records .....	5-207
5.12.9.4	Scenario 2: Manual production process .....	5-209
5.12.9.5	Scenario 3: Automated production process (job list) .....	5-211
5.13	Prompting the operator .....	5-212
5.13.1	Help information on the operating unit.....	5-213
5.13.2	On-screen keyboard .....	5-214
5.13.3	Icons for local function keys.....	5-216
5.13.4	Showing and hiding objects .....	5-216
5.13.5	Dynamic attributes .....	5-217
5.13.6	Light-emitting diodes on the operating unit.....	5-218
5.13.7	Tab order .....	5-219
5.13.8	Width and order of columns (screen objects) .....	5-221
5.13.9	Dynamic screen objects.....	5-222
5.13.10	Assigning operator permissions.....	5-223
5.13.11	Password administration.....	5-225
5.14	Creating VB scripts .....	5-227
5.14.1	Using ProTool objects in scripts .....	5-229
5.14.2	Internal script tags and ProTool tags.....	5-230
5.14.3	Debugging scripts in ProTool.....	5-231
5.14.4	Constraints with VBScript .....	5-232
5.14.5	Examples of creating VBScripts .....	5-234
5.14.5.1	Example of a subroutine .....	5-234
5.14.5.2	Example of a function .....	5-237
5.15	Configuring in foreign languages .....	5-240
5.15.1	System requirements for foreign languages .....	5-240
5.15.2	Languages in ProTool.....	5-241
5.15.3	System message languages.....	5-243
5.15.4	Language dependent keyboard assignment .....	5-244
5.15.5	Reference text.....	5-246
5.15.6	Steps to creating a multilingual project.....	5-248
5.15.7	Exchanging text with translators .....	5-250
5.15.8	Requirements for configuring in Asian languages .....	5-252
5.15.9	Constraints on projects in Asian languages .....	5-253

<b>6</b>	<b>Testing projects .....</b>	<b>6-1</b>
6.1	The Windows CE Control Panel .....	6-2
6.2	Downloading the executable project file .....	6-4
6.3	Simulation options .....	6-6
6.4	Simulation of the project with the simulator .....	6-7
6.5	Uploading projects .....	6-10
6.6	Directly accessing values on the PLC (Status/Force tag) .....	6-12
6.7	Troubleshooting .....	6-13
6.7.1	Error messages during compilation .....	6-13
6.7.2	Error messages while downloading .....	6-14
<b>7</b>	<b>Documenting and managing projects .....</b>	<b>7-1</b>
7.1	Printing project data .....	7-1
7.2	Project management with integrated operation .....	7-3
7.3	Managing projects in stand-alone operation .....	7-4
7.4	Constraints with printing .....	7-7
7.5	Example of documenting a project .....	7-8
7.5.1	How to create a customized printout .....	7-8

## Appendix

<b>A</b>	<b>HMI System Messages.....</b>	<b>A-1</b>
<b>B</b>	<b>Abbreviations .....</b>	<b>B-1</b>



# 1 Introducing ProTool

## ProTool components

SIMATIC ProTool consists of the ProTool CS (Configuration System) configuration software and a Runtime software for process visualization ProTool/Pro RT for example. Both systems can run on the Windows 98 SE, Windows Millenium, Windows 2000, and Windows NT 4.0 operating systems.

A separate Runtime software can also run on Windows CE devices.

- With ProTool CS, you create your projects on a configuration computer (PU or PC) in Windows.
- With ProTool/Pro RT, you let your project run on the operating unit and visualize the process. You also use ProTool/Pro RT on your configuration computer to test and simulate the compiled project file.

## Windows-based systems

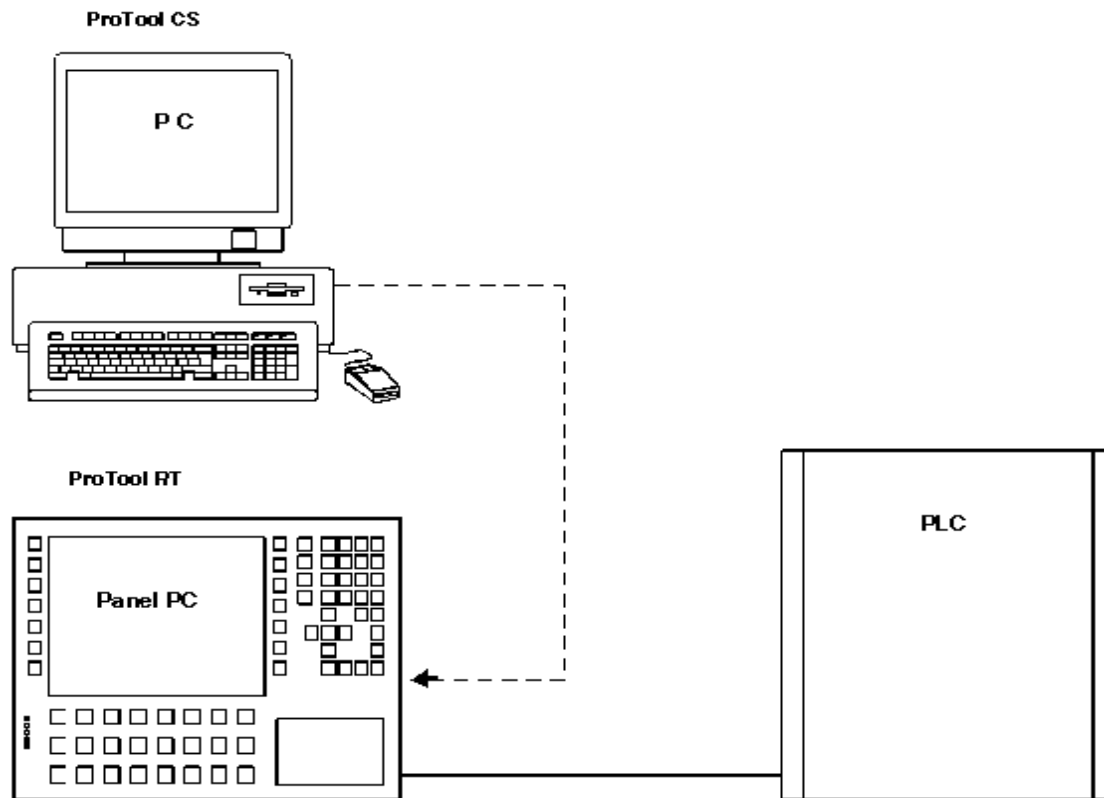
SIMATIC ProTool/Pro RT can be installed (plug and play) on the following Windows CE devices:

- Multi Panel, such as MP 270B, MP 370
- Panel, such as OP 170B, TP 170B, TP 170A, TP 270, OP 270
- Mobile Panel, such as Mobile Panel 170

SIMATIC ProTool/Pro RT can be run on the following Windows-based PC systems.

- SIMATIC Panel PC, such as PC 670, PC 670 TOUCH, PC 870, PC 870 TOUCH
- Standard PC

## A typical structure



*Example of PC as configuration computer and SIMATIC Panel PC as operating unit*

## Authorization

If you install the ProTool/Pro RT runtime software on a standard PC or on a SIMATIC Panel PC, you require authorization for unrestricted use. If authorization is not available, ProTool/Pro RT runs in Demo mode.

- PC: The authorization has to be ordered separately.
- SIMATIC operating units with enclosed runtime software: You purchased the authorization for SIMATIC operating units with enclosed runtime software (such as the Panel PC 670) with the unit; it is enclosed with the unit.
- Plug-and-play SIMATIC operating units: No authorization is required for plug-and-play SIMATIC operating units (such as TP 170B or MP 270).

## ProTool is easy to use and versatile

The fully graphical user interface of ProTool CS allows you to create object-oriented, symbol-based projects easily by mouse click. No special programming knowledge is required. ProTool features:

- User-friendly process visualization with a large selection of standardized input/output fields, bar graphs, trend views, raster and vector graphics and attributes with dynamic capability.
- An integrated message system.
- Archiving of process values and messages.
- User functions on the basis of Visual Basic Script.
- Drivers for connection to the SIMATIC S5, SIMATIC S7, SIMATIC 505 and many PLCs from other manufacturers.

## ProTool and SIMATIC STEP 7

ProTool can be integrated in the SIMATIC STEP 7 configuration software, thus allowing you to select SIMATIC STEP 7 symbols and data blocks as tags in ProTool. This not only saves you time and money but also eliminates the possibility of errors made when entering the same data several times.

You can use ProTool as a stand-alone application or integrated in SIMATIC STEP 7. In this documentation, attention is drawn at appropriate points to the fact that the procedures for the two configurations differ.

You will find more information about the configuration of ProTool with SIMATIC STEP 7 at the topic "Configuring with ProTool integrated in SIMATIC STEP 7".

## Offline configuration

With ProTool you create and edit your projects offline. The device need not yet be available at this time. The configuration computer displays the configured project data as it will subsequently be displayed on the device.

On completion of configuration you can download the executable project file from the configuration computer to the device.

## **1.1 Overview of the documentation**

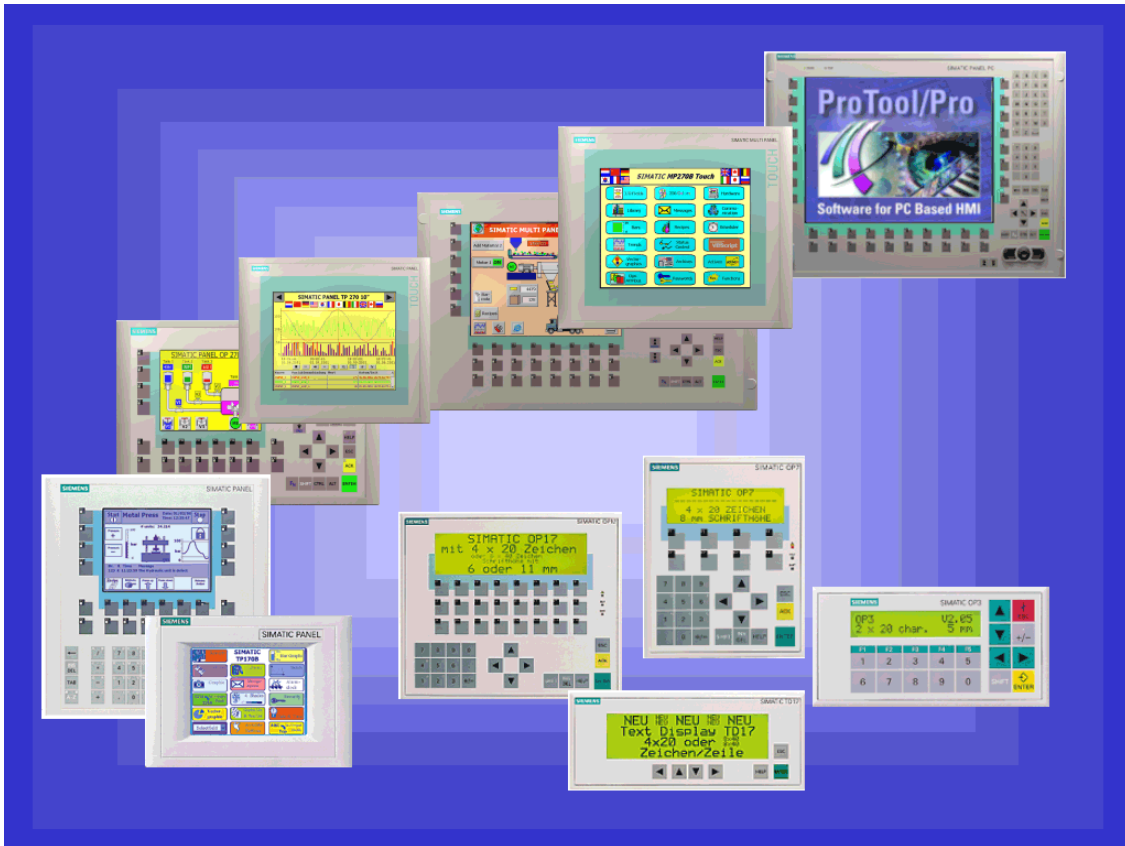
The SIMATIC HMI documentation is made up of a combination of manuals, instructions and online Help in keeping with the range of target groups. This chapter provides a broad outline.



## 1.1.1 ProTool for the whole SIMATIC HMI device family

### The SIMATIC HMI device family

The SIMATIC HMI device family is a complete family of text displays, operator panels, touch panels and Windows-based systems for efficient machine operation and monitoring. The performance and convenience of the devices are finely tuned to suit the individual demands made of them.



*SIMATIC HMI operating units*

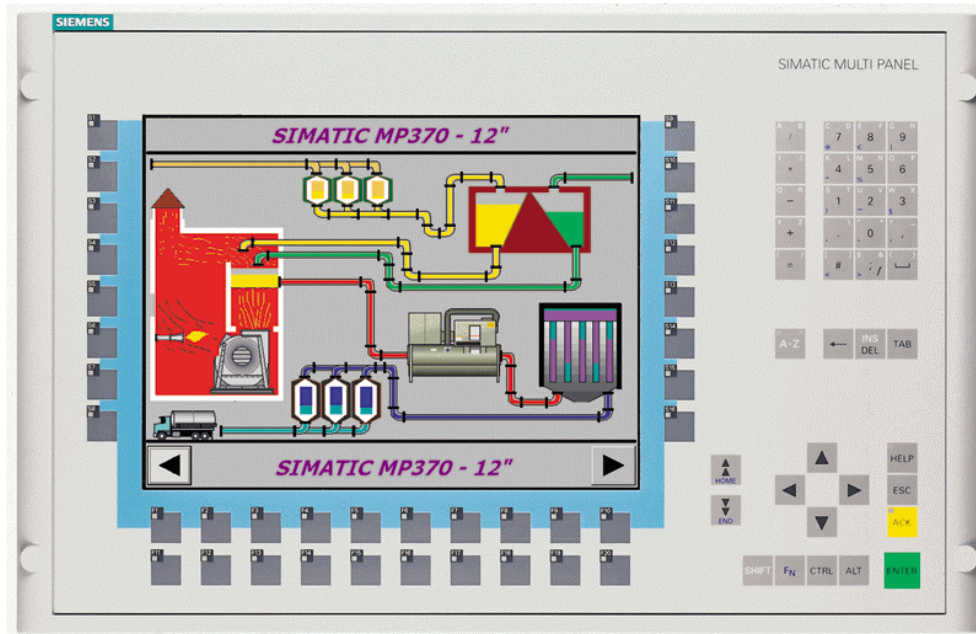
The great advantage is that you configure all the devices with the same configuration software.

## 1.1.2 ProTool for Windows-based systems

### Devices

The "ProTool - Configuring Windows-Based Systems" documentation tells you how to configure the following devices:

- Panels
  - TP 170A
  - TP 170B
  - TP 170B Color
  - TP 270 (6" and 10")
  - OP 170B
  - OP 270 (6" and 10")
- Mobile Panels
  - Mobile Panel 170
- Multi Panels
  - MP 270B
  - MP 270B TOUCH
  - MP 270
  - MP 370
  - MP 370 TOUCH
  - MP 370 15" TOUCH
- Panel PCs
  - FI 25
  - FI 45
  - PC 670 10"
  - PC 670 12"
  - PC 670 15"
  - PC 670 12" TOUCH
  - PC 670 15" TOUCH
  - PC 870 12"
  - PC 870 15"
  - PC 870 15" TOUCH
  - PC IL 70 12" TOUCH
  - PC IL 70 15" TOUCH
- SINUMERIK Panels
  - OP 010
  - OP 012
  - OP 015
- SIMOTION Panels
- PC



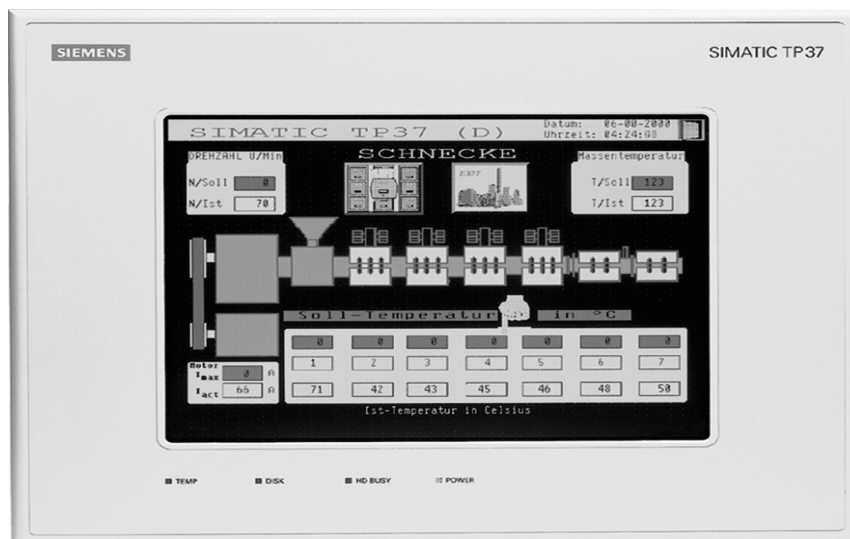
Example: MP 370

### 1.1.3 ProTool for graphical displays

#### Devices

The "ProTool - Configuring Graphics Displays" documentation tells you how to configure the following operating units:

- Graphics-oriented operator panels
  - OP 27
  - OP 37
- Touch panels
  - TP 27-6
  - TP 27-10
  - TP 37
- C7 devices
  - C7-626 (OP 25 with integrated S7-CPU)



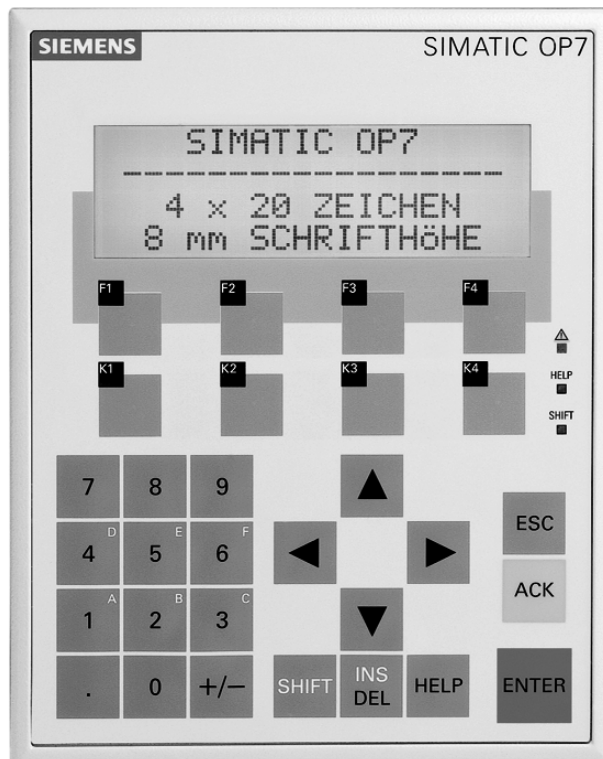
Example: TP 37

## 1.1.4 ProTool for text-based displays

### Devices

The "ProTool - Configuring Text-Based Displays" documentation tells you how to configure the following operating units:

- Text-oriented operator panels
  - OP 3
  - OP 7
  - OP 17
- Text displays
  - TD17
- C7 devices
  - C7-621 (OP 3 with integrated S7-CPU)
  - C7-623 (OP 5 with integrated S7-CPU)
  - C7-624 (OP 15 with integrated S7-CPU)
  - C7-633 (OP 7 with integrated S7-CPU)
  - C7-634 (OP 17 with integrated S7-CPU)



Example: OP 7



## 2 Getting started in ProTool

Here you learn how to start ProTool, configure Windows-based systems, and what you should look out for while you are configuring.

## 2.1 Starting ProTool

### Starting ProTool

After ProTool has been installed, you will find a folder on the Start menu called Simatic, containing the following icons:

#### ProTool CS V6.0



Calls online Help.



Starts the configuration software.



Shows late-breaking information on the configuration software.



Starts Setup in order, say, to change the ProTool language or to add a component.

#### ProTool RT V6.0



Copies the compiled project file in compressed form.



Starts the runtime software.



Shows late-breaking information on the runtime software.



Starts the simulation software.



Starts the loader.

#### ProSave 6.0



Opens ProSave as a stand-alone application.



## 2.2 Configuring Windows-based systems

### Getting Started

If you are not all that familiar with the ProTool visualization software, we would recommend you read the short guide and get started with ProTool with the help of the example. The printed version is enclosed with the manual.

### Requirements for working with the brief introduction

To do the exercises for ProTool CS in this brief introduction, you require

- a PC as a configuration computer
- the ProTool software package ProTool incorporates the ProTool CS configuration software and the ProTool/Pro RT Runtime software.
- an operating unit, such as a SIMATIC Panel PC.

### Other documents on ProTool CS

You will find the electronic manuals on the ProTool CD at:

Docs\..\UsersManual\_Win.pdf

You can find all the information contained in the manual in ProTool's online Help.

## 2.3 Notes on the configuration computer

### Notes on the configuration computer

So that you can work as efficiently as possible when you are configuring, give due consideration to the following notes for your configuration computer:

- **Close applications**  
Close all other programs. In this way more main memory will be available for ProTool.
- **Configurations on a local drive**  
Do not start projects from networks, but from your local drive. Network access times are frequently too long.
- **Large/small Windows fonts**  
In configurations that you create on a configuration computer with small Windows fonts and then open on a computer with large windows fonts or start on the operating unit (Runtime), the text on configured buttons appears too large. Therefore, when different computers are being used, make sure that the Windows fonts set are identical.  
The following applies to devices in the Windows CE family:  
As a general rule, set small Windows fonts on the configuration computer so that the WYSIWYG functionality is not lost.
- **Available disk storage**  
Your drive should have free disk storage space. Take note of Microsoft's recommendations for the Windows system in this respect.
- **More working memory**  
Memory requirements rise in proportion to the number and size of the screen objects you are using. An increase in available main memory reduces processing times considerably.
- **System resources**  
Make sure when you are working with ProTool that you have 20 to 40 MB of main memory available and at least 65% free system resources. You will find information on this topic in Explorer by choosing "Help" > "About". Increase the size of your main memory with additional RAM or virtual memory, as necessary. Additional system resources are released by closing other applications.

- Environment tag TEMP  
In Windows, a temporary file is created for files when they are opened and processed. You decide where this temporary file is stored by means of the TEMP environment tag.

The path of the TEMP environment tag is normally set in the `autoexec.bat`® file. If the TEMP environment tag is not placed in the `autoexec.bat` system file, the temporary file is stored in the local Windows directory. If an error message is issued by ProTool that indicates an error upon write access, the storage space on the drive might be too small for the temporary file.

In Windows 2000, the environment variable "TEMP" is set at "Start" > "Settings" > "Control Panel" > "System" > "Advanced" tab > "Environment variables" button.

In Win NT 4.0, you set it at "Start" > "Settings" > "Control Panel" > "System", "Environment" tab.



### 3 ProTool integrated in SIMATIC STEP 7

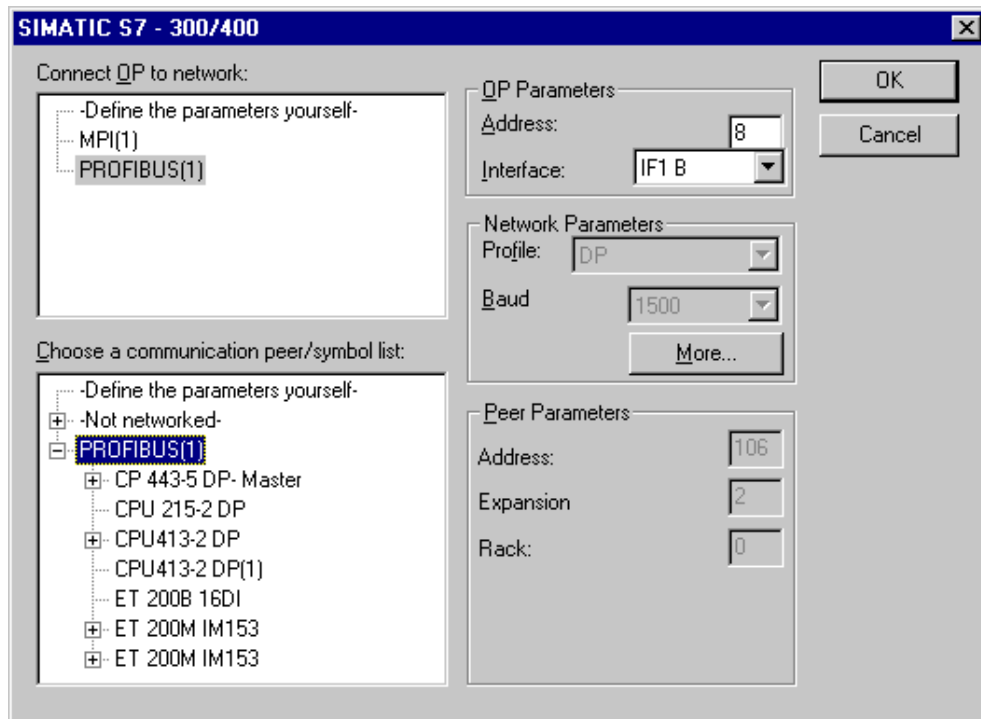
#### Requirements

If the PLC you are using is a SIMATIC PLC and you have installed STEP 7 configuration software on your system, you can integrate ProTool in SIMATIC STEP 7.

#### Advantages of STEP 7 integration

With integrated configuration, you access the database that you created when you configured the PLC in SIMATIC STEP 7. You then have the following advantages:

- You can use SIMATIC Manager as a central point for creating, editing and managing ProTool projects.
- The communication parameters of the PLC are created as default values when you create your ProTool project and reset when SIMATIC STEP 7 is changed.



*Example of the "Driver Parameters" dialog box with STEP 7 integration: network parameters and parameters of the peer are set as default values.*

- When configuring tags and area pointers, you can directly access the STEP 7 symbols in ProTool. Simply select the STEP 7 icon in the ProTool tag dialog box to which you wish to connect a tag. Changes to the symbol table in SIMATIC STEP 7 are updated in ProTool.
- You assign symbolic names once only and use them in SIMATIC STEP 7 and ProTool.  
For this purpose, you must assign a name to a DB (data block) in the SIMATIC STEP 7 symbol table. If this is not the case, this DB is not offered for selection in the ProTool "Tags" dialog box.
- ALARM\_S messages configured in SIMATIC STEP 7 are supported by ProTool and can be output to the operating unit.

### Switching STEP 7 integration from ProTool

A ProTool installation with integration in SIMATIC STEP 7 provides you the menu item "ProTool Integration in STEP 7" under the "File" menu in ProTool. This allows you to integrate ProTool in SIMATIC STEP 7 or to operate it in stand-alone mode. After changing to integration mode, ProTool is automatically restarted.

Integration of ProTool in SIMATIC STEP 7 is still possible in "ProTool Setup", irrespective of this.

In integrated mode it is also possible to copy single configurations from STEP 7. You can then edit them in ProTool stand-alone mode (e.g. on another configuration computer).

### Integrating specific ProTool configurations and copying them from STEP 7

Configurations generated in ProTool stand-alone mode cannot be called directly in SIMATIC Manager. To be able to implement such configurations in a STEP 7 project, you need to integrate them.

Execute by selecting the menu item "File" > "Integrate in STEP 7 project" in ProTool.

Vice versa, if you want to use ProTool in stand-alone mode to edit projects created in SIMATIC STEP 7 in integrated mode and without having SIMATIC STEP 7 available, you must first generate a copy of these configuration files offline of STEP 7.

To do this, select the menu item "File" > "Copy from STEP 7 project" in your ProTool application.

---

#### Note

When you open a stand-alone project, there can no longer be automatic synchronization of the database between SIMATIC STEP 7 and ProTool. You therefore have to ensure that the shared database is resynchronized (for example, by re-integrating) if changes are made to the STEP 7 program.

---

### 3.1 How to start ProTool in SIMATIC STEP 7

#### General

There are two ways in which you can start ProTool in SIMATIC STEP 7:

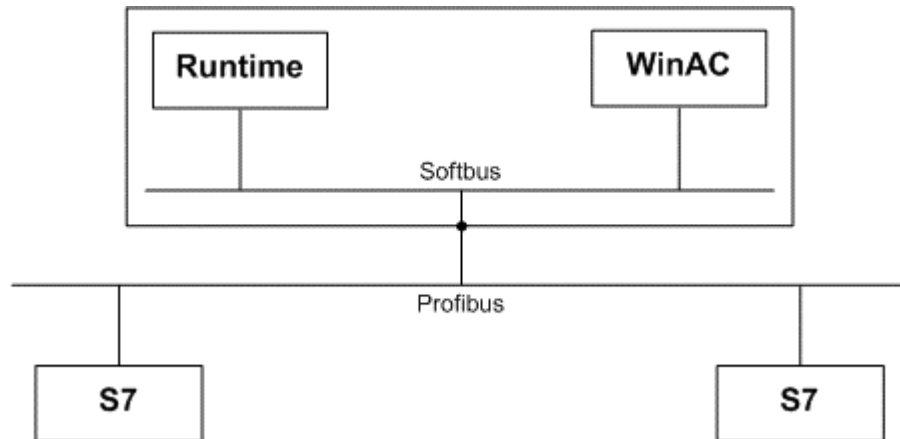
- By means of SIMATIC Manager
- Directly in Windows

#### Starting ProTool with SIMATIC Manager

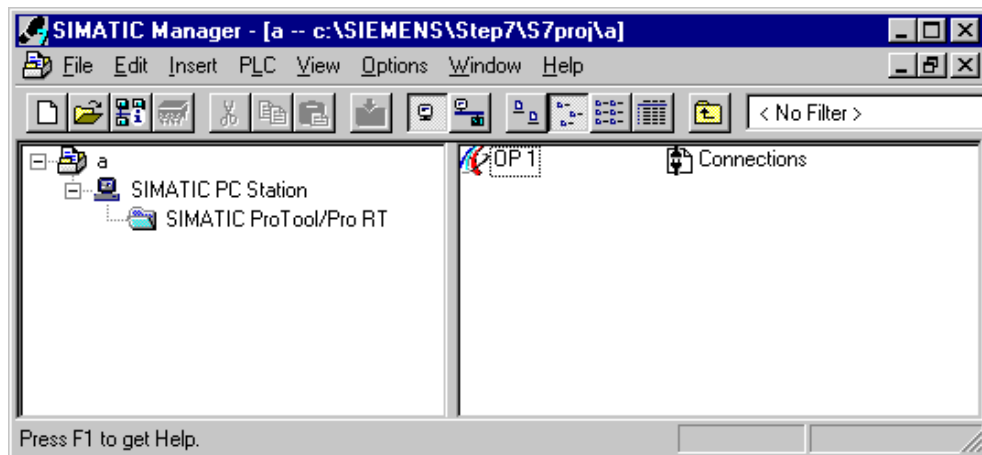
Procedure:	
1	Start SIMATIC Manager.
2	Select a STEP 7 project, or create a new STEP 7 project.
3	Choose the "Paste" > "Station" > "SIMATIC OP" option from the menu, and enter a name for the ProTool project.
4	Start ProTool by double-clicking the ProTool project. Then select the operating unit for the project.

#### Starting ProTool on a SIMATIC PC station

If you install a WinAC PLC on a PC station on which the Runtime software is installed, you can address it in ProTool Runtime as a PLC. In addition, you can communicate over a network with SIMATIC PLCs outside the PC station.



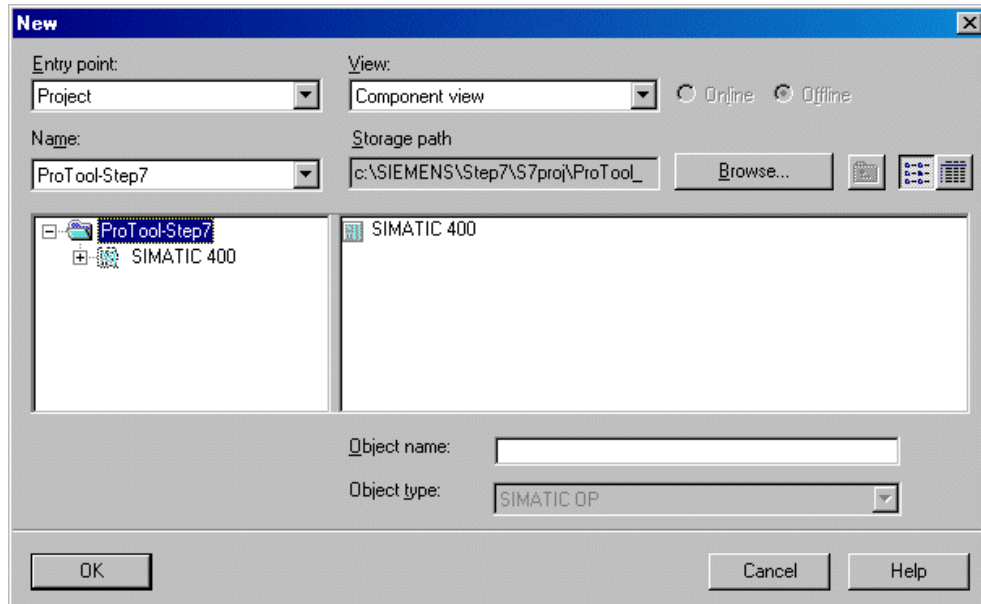
Procedure:	
1	Start SIMATIC Manager.
2	Select a STEP 7 project, or create a new STEP 7 project.
3	Choose the "Paste" > "Station" > "SIMATIC PC Station" option from the menu, and enter a name for the SIMATIC PC station.  Note: Also take note of the online Help in STEP 7 and the documentation on the SIMATIC NET CD-ROM. More information is available in the ProTool Information System at "Technical data and system limits".
4	Select the "Open Object" from the shortcut menu of the SIMATIC PC station (right click).  "HW Config" is started and the PC station is opened in it.
5	Select the "Paste" > "Hardware Components" option from the menu. The "Hardware Catalog" dialog box appears.
6	Select "SIMATIC PC Station" > "HMI" and drag the entry "SIMATIC ProTool/Pro RT" to a spare slot in the "PC" dialog box. Do not use slot 2, since it is reserved for Soft CPUs (WinAC).
7	Close the "HW Config" application.
8	Start ProTool by double-clicking the ProTool project. Then select the operating unit for the project.





## Starting ProTool in Windows

Start ProTool directly under Windows. Choose "File" > "New" from the menu to open a dialog box in which you can select a STEP 7 project and create a ProTool project in it. You then select the operating unit.



Example of the "New" dialog box for integrating STEP 7

## 3.2 SIMATIC Manager

### Requirement: ProTool integrated in SIMATIC STEP 7

If you are not working with ProTool integrated in SIMATIC STEP 7, SIMATIC Manager is available to you for your ProTool projects.

### SIMATIC Manager

As will be familiar to you from SIMATIC STEP 7, you can use SIMATIC Manager as a central point for managing your projects. You save time since you can perform the following tasks without opening ProTool:

- Compile and download ProTool projects
- Start ProTool RT
- Export and import blocks of text for translation
- Perform language settings
- Copy or move ProTool projects
- Archive and unarchive ProTool projects as part of SIMATIC STEP 7 projects

For further information refer to the documentation on SIMATIC Manager.

---

#### Note

ProTool's Project Manager is not available to you in the event of integrated operation. ProTool data can no longer be viewed independently in this event, since the data is always linked to a STEP 7 project. It therefore has to be managed and backed up using this application.




---

### 3.3 Working with SIMATIC Manager

#### Advantages

You can deal with many tasks more efficiently in SIMATIC Manager than in ProTool. Since there is a separate documentation for SIMATIC Manager, only a short summary will be presented here of the menu commands available to you for your ProTool projects in SIMATIC Manager.

For further information refer to the documentation on SIMATIC Manager.

Task	Call
Compile a project	Choose "Edit" > "Compile" from the menu
Download a project to an operating unit	Choose "System" > "Load" from the menu or "System" > "Load" from the station shortcut menu or click 
Start ProTool RT	Choose "Edit" > "Start Runtime" from the menu
Export and import blocks of text for translation	Choose "Options" > "Manage Multilingual Text" > "Import" or "Export" from the menu or choose "Manage Multilingual Text" > "Import" or "Export" from the shortcut menu
Perform language settings	Choose "Options" > "Manage Multilingual Text" > "Language" or "Delete Language" from the menu or choose "Manage Multilingual Text" > "Language" or "Delete Language" from the station shortcut menu
Copy or move projects	Choose "Edit" > "Copy" or "Cut" from the menu or choose "Copy" or "Cut" from the station shortcut menu or click  or  or drag and drop.
Archive and unarchive SIMATIC STEP 7 projects including	Choose "File" > "Archive" or "Unarchive" menu

## 3.4 Changing the PLC and applying tags

### General

If you change the PLC (for example, when changing to the new model), the tag connections are not lost. Tag names up to 80 characters long ensure, when tags are applied from SIMATIC STEP 7, that the symbol names are applied in full.

### Changing the PLC

The connection between a S7-CPU and a ProTool PLC might be lost. This means that there is then no connection between the STEP 7 symbols and ProTool tags, either.

You will find more information about this in "Restoring symbolic tag connections".

### Applying tags from SIMATIC STEP 7

The tag names, which you apply to the ProTool project from SIMATIC STEP 7, are formed from the last component of the STEP 7 overall symbol (the tag name "testname" is derived from "RotAxis.Speed.XDirection.testname", for example).

Identical tag names are given an index, starting at "1", for unique identification. The index always remains the same for the tag. Unsupported characters within the tag name are replaced by underscore characters ("\_").

If you have since changed a tag name applied in that in that form, you are asked the next time you apply the tag name whether you would like to retain the old tag name or whether it should be adjusted to the STEP 7 symbol.

### Applying arrays from SIMATIC STEP 7

If you are using a SIMATIC S7 300, a SIMATIC S7 400 or a SIMOTION PLC, you can also apply whole arrays from SIMATIC STEP 7, in addition to tags. For this, you must be operating ProTool integrated in SIMATIC STEP 7.

---

#### Note

If you are using the SIMATIC 300/400 PLC driver and would like to apply arrays, perform the following steps:

- In the "Symbols" dialog box of the Tag dialog box, select the first array element.
  - At "Number of elements", enter the number of elements contained in the array.
-

## 3.5 How to integrate and copy STEP 7 projects

### Computer without SIMATIC STEP 7

Perhaps you have only installed both ProTool and SIMATIC STEP 7 on your workstation machine but would like to take a completed project with you on a Notebook computer or a PU to a machine on site. In this case you can unintegrate an integrated project and work with it as a stand-alone project in ProTool. The data of the PLC programming in SIMATIC STEP 7 are then no longer available.

Conversely, you can re-integrate a project which you edited as a stand-alone project.

---

#### Note

When you open a stand-alone project, there can no longer be automatic synchronization of the database between SIMATIC STEP 7 and ProTool. You therefore have to ensure that the shared database is resynchronized (for example, by re-integrating) if changes are made to the STEP 7 program.

---

### Integrating a project

Procedure:	
1	Open ProTool.
2	Choose "File" > "Integrate" from the menu.
3	Enter the directory or the name of the *.pdb file that you would like to integrate.
4	Connect a PLC in ProTool to the corresponding CPU in SIMATIC STEP 7. The project is integrated and the connection of the tag to the symbols of the PLC can be established by choosing "Edit" > "Restore Symbols".

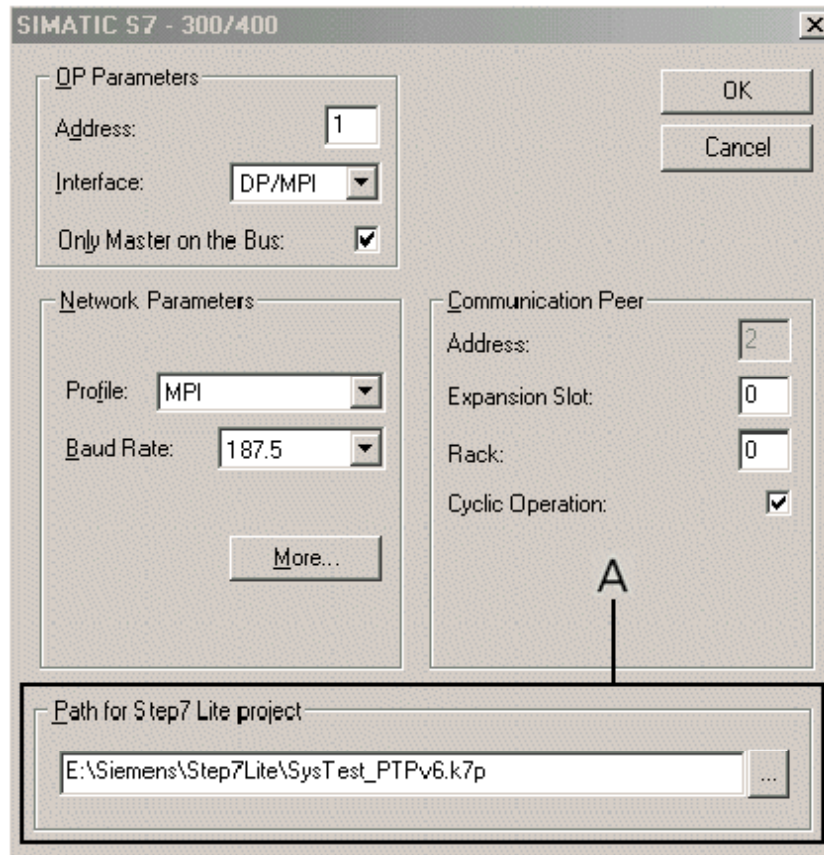
### Copying a project from STEP 7

Procedure:	
1	Open the project in ProTool.
2	Select the menu item "File" > "Copy from STEP 7 project".
3	Enter a directory and filename. The configuration is saved as *.pdb file. You can copy and move it in the same way as any other file.

## 3.6 Connecting to STEP 7 Lite

### General

ProTool will automatically detect a STEP 7 Lite installation on your configuration computer. If you create a new project and use the "SIMATIC S7 300/400" driver, a selection is added to the Parameters dialog box (area A):



Here you can select a STEP 7 Lite project. The symbol information is stored in the project, for example. If the CPU was configured in STEP 7 Lite, its address has already been saved in the project.

### Constraints

The following constraints apply to STEP 7 Lite:

- STEP 7 Lite supports only one CPU.
- When you work with ProTool integrated in SIMATIC STEP 7, STEP 7 Lite is not supported.

For further information, please consult your STEP 7 Lite documentation.

## 3.7 How to restore the symbolic tag connections

### Background

The connection between a S7-CPU and a ProTool PLC might be lost. This means that there is then no connection between the STEP 7 symbols and ProTool tags, either.

The following causes are possible for this:

1. Reuse of a ProTool project in another STEP 7 project.

---

#### Note

If the S7-CPU is copied together with the ProTool project within STEP 7 to a new STEP 7 project, the connections are retained.

2. Integration of a ProTool project into a STEP 7 project.
3. Replacement of an S7-CPU in the STEP 7 project.  
The CPU swop within STEP 7 HWCONFIG is not normally affected by this.

---

#### Note

The ProTool tags concerned are reported in the output window when the project is opened.

### Procedure:

Perform the following steps to restore the connections between ProTool tags and STEP 7 symbols.

Procedure:	
1	In ProTool CS, in the "Properties" dialog box, connect the ProTool PLC to the corresponding STEP 7 CPU by means of the "Parameters" button.  When doing so, you receive a message to the effect that the connection of the tag to the STEP 7 symbols has been restored.
2	In ProTool CS, choose "Edit" > "Restore Symbols".
3	From then on, the following cases are possible for the STEP 7 symbol used in the ProTool tag:  <ol style="list-style-type: none"> <li>1. A STEP 7 symbol that is used is present</li> <li>2. A STEP 7 symbol that is used is present but the address in the S7 program is different</li> <li>3. A STEP 7 symbol that is used is not present in the STEP 7 symbols, but another STEP 7 symbol having the same address is present</li> <li>4. A STEP 7 symbol that is used is not present and there is no other STEP 7 symbol that has the same address</li> </ol> In case 1, the ProTool tag is linked to the STEP 7 symbol.  In case 4, an error message is issued, and the ProTool tag cannot be linked and has to be corrected manually.

<b>Procedure:</b>	
4	<p>In case 2 and 3, you have the following options:</p> <ul style="list-style-type: none"> <li>• "Connect" button: The ProTool tag is linked to the STEP 7 symbol.</li> <li>• "Disconnect" button: The ProTool tag is identified as not being symbolically linked.</li> <li>• "Jump" button: This can be used to postpone the decision to a later run The ProTool tag remains in the error state and is not connected.</li> </ul>
5	After ProTool Tag Management is executed, the symbolic connection is restored to the extent possible.

**Note**

The ProTool tags that could not be successfully connected gain are read out to the output window.

**Tag connection for SIMOTION****Note**

The connection to symbols is normally lost as a result of renaming objects in SIMOTION SCOUT. You will find more information in the section "Process communication" > "SIMATIC S7" > "SIMOTION".



## 4 Creating and editing projects

Here you obtain an overview over:

- the ProTool project structure
- the procedure for creating a project
- copying objects
- displaying project information

## 4.1 Fundamental considerations when creating a project

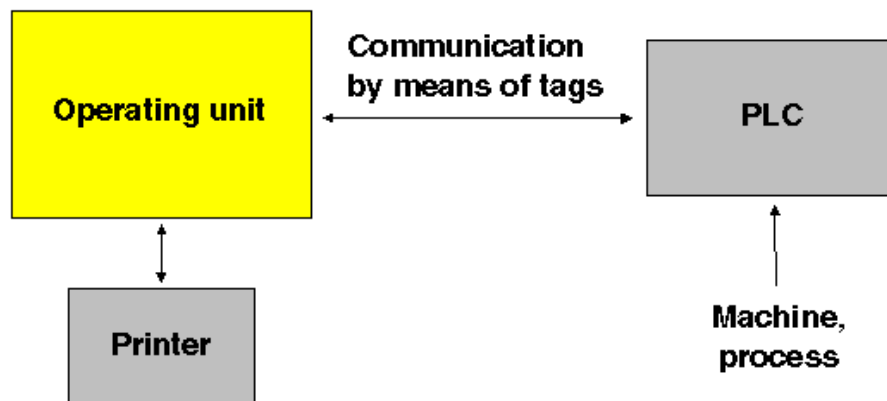
### Objective

You want to make it possible to operate and visualize a machine or process. To do this, you map the machine or process on the operating unit as accurately as is necessary.

### System configuration

Communication takes place between the operating unit and the machine or process by means of "tags" via the PLC. The value of a tag is written to a memory area (address) on the PLC, from where it is read by the operating unit.

The following diagram provides an overview of the fundamental structure:

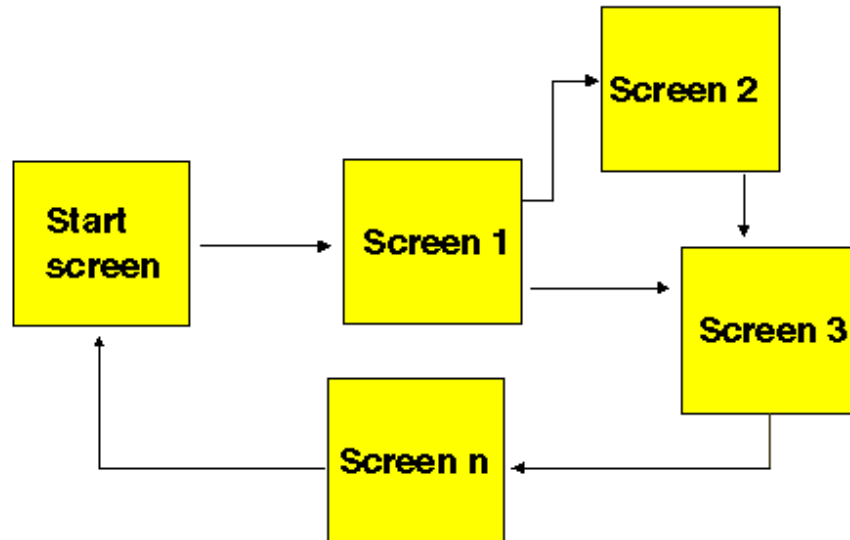


*A typical structure*

## Defining an operating philosophy

Consider which process values or states of the machine you want to display on the operating unit or work with and which object types you will have to configure for the purpose.

1. Create a user interface Screens are the central components of the project, with which you visualize the states of the machine or process and create the prerequisites for operating the machine or process.



### Example of screen creation

2. You can create a number of screens with display elements and controls - for switching between screen, for example - for the user interface with which the operator is subsequently faced on the operating unit.
3. Process data entry and transfer: In order to enter process data or set new values, you configure input/output fields, for example. This is also possible as a symbolic display.
4. Report process states: To acquire and log process and operating states on the operating unit or to archive them, configure messages.
5. Define communication areas A defined address area on the PLC for data interchange with the operating unit is addressed by means of an area pointer. The number of area pointers available varies depending on the selected operating unit.

---

### Note

Disable the "Cyclic operation" option for the following CPUs of distributed systems in the PLC dialog box:

- IM151 (ET200 S CPU)
- BM147 (ET200 X CPU)

In this way you avoid possible communication problems.

---

## 4.2 Components of a ProTool project

### What is displayed in the project window?

When you open a new or an existing project, the project window opens.

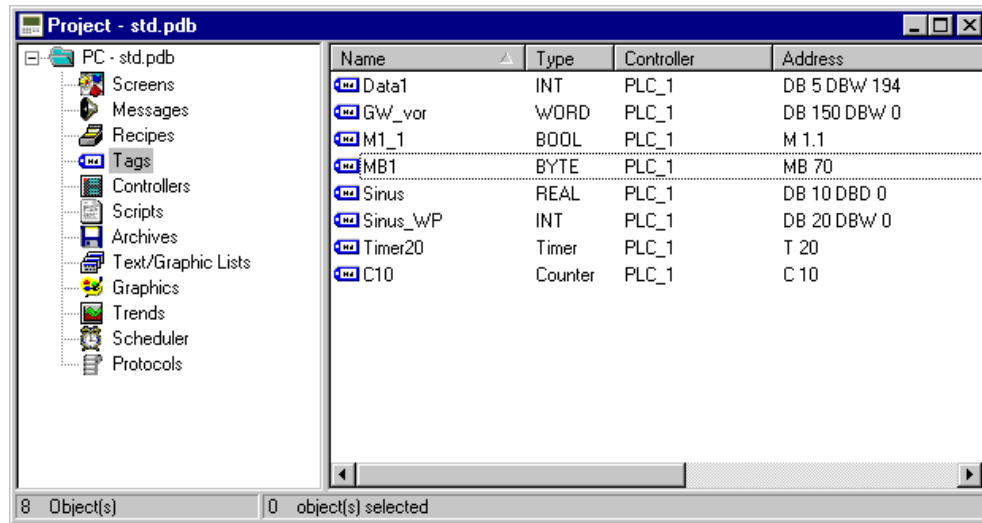
The project data of a ProTool project is stored in the form of objects. The objects in a project are arranged in a tree structure.

The left side of the project window displays object types that belong to the project and you can configure for the selected operating unit. The object types contain objects with settable properties, which are shown in the right section of the project window. The objects that you can configure depend on the type of the operating unit.

The various objects are linked directly in ProTool with the tool required to edit them.

The project window is structured as follows:

- The title bar contains the project name.
- Depending on the operating unit, the configurable object types are displayed in the left half of the window and the objects you create in the right half.


















*Example of a project window with tags*

### Note

If you maximize the project window, tabs are displayed for the open windows along the bottom border to enable you to change easily from one window to another.

## 4.3 Objects in the project window

### Overview Objects in the Project Window

Object symbol	Object type	Object name (default)
	ProTool project	PRO1
	Screens	PIC_1
	Messages	---
	Recipes	STRUCT_1
	Tags	VAR_1
	PLCs	PLC_1
	Scripts	SCRIPT_1
	Archives	ARCHIV_1
	Text or graphic lists	LIST_1
	Graphics	GRAPHIC_1
	Trends	CURVE_1
	Scheduler	ALARM_1
	Reports	REPORT_1
	Multiplex tag	MUXVAR_1
	Area pointers	---

## 4.4 Steps to be taken when creating a project

### Principle

To create a project, perform the following steps:

1. Create a new project ("File" > "New" or "File" > "Open"). Choose "File" > "New" to create a new project. The project assistant guides you through a number of selection dialogs.
2. Select your PLC: Select a driver for your PLC. Only those drivers with which the selected operating unit can be run are displayed.
3. Enter information about the project in the "summary", should you wish to. Click the "Create" button to open the project window.
4. Set the communication areas (project window: "Area pointers"): Shared communication areas (area pointers) have to be set for certain functions such as "bit message procedure" and "synchronous data record transfer" to be used.
5. Create your project: This involves the most work. You can approach it in one of two ways: Either you create all the individual parts first and then link them to form a meaningful structure (the bottom > up approach), or you begin by designing a structure and then fill it with the individual elements (the top > down approach). To do this, you basically have to perform the following steps:
  - Partition the display on the operating unit.
  - Create a user interface with display elements and controls.
  - Configure tags to allow data exchange with the PLC.
  - Configure messages to receive information about the machine status or the process.
6. In addition, you can configure additional objects, such as scripts, depending on the operating unit.

---

### Note

If you modify and save the project several times, the project file gets larger (\*.pdb). Reduce the size of the file by choosing "Save As" from the menu. This causes the file structure to be reorganized.

---

## 4.5 Configuring behavior at run time

### Constraints

Depending on the device you happen to be configuring, not all options listed in the "Settings" dialog box will be available. All the settings that are possible are listed in the table below.

### What are the possible settings?

To set how the project should behave at run time and the options open to the operator, you can perform the following settings at "System" > "Settings":

Setting	Explanation
User version	The user version identifies the version of the project.
OP password	Here you enter the administrator's password and the time after which the administrator has to log on again if there has been no operator input (time-out).
OPC server	Select this check box in order to set the operating unit as the server in the OPC network in the executable project. Conversely, OPC clients can recognize the operating unit as the server in the OPC network.
Display: Lock Task Switching	This option allows you to stop the operator calling up a different program on the operating unit. The operating unit is then only available to the ProTool project at run time.  If you have selected "Lock task switching", it is essential to configure a key with the function <code>Exit_Runtime</code> in the project. Otherwise you will not be able to exit ProTool/Pro RT or Windows. The key combination CTRL + ALT + DEL has no effect either.

Setting	Explanation
Display: Full Screen	<p>You should decide right at the start of a project how you want to display it at runtime, so that you can create your screens in the correct size.</p> <p>When you select this check box, the project is displayed on the operating unit while making full use of the screen size. This setting is useful for most projects. With the "Full Screen" setting, you have a somewhat larger area available for your screens and the fixed window, as the border and title bar are missing from the window.</p> <p>If the "Full Screen" setting is deselected, the project runs in a Windows-type window with border and title bar. The operator can vary the size of the window, he can minimize or maximize the window, put the window in the task bar or close it, and thus exit ProTool/Pro RT.</p>
Display: Use On-Screen Keyboard	<p>If you select this check box, the on-screen keyboard will be displayed automatically at run time when you select an input field. This is practical with non-touch devices and mouse-operated devices.</p>
Display: Show Tooltips	<p>If you select this check box, the operator is shown tooltips at run time for the input fields you configured (such as limit values).</p>
Cursor control as for v.5.10	<p>If you select this check box, the cursor control acts in the same way as in ProTool 5.10.</p>

Of course, depending on the project requirements, you can combine settings for run time. This allows you, for example, to ensure that a system is only available in a specific ProTool project.



## 4.6 PLC drivers

### PLC drivers available for Windows-based systems

Select the PLC driver with the project assistant in the "Select PLC" dialog box.

You can also define or later edit the parameters there if you select PLC "Properties" in the project window.

---

#### Note

You will find information about the various PLC drivers at "Start" > "SIMATIC" > "ProTool" > "ProTool Information System" in the topic "Process communication".

---

You can select the following drivers for Windows-based systems:

SIMATIC S5 - AS511

SIMATIC S5 - DP

SIMATIC S7-300/400

SIMATIC WinAC

SIMATIC S7-200

SIMATIC 500/505

SIMATIC 500/505-DP

SIMOTION

OPC (Client/Server)

Allen-Bradley DF1 (PLC5/SLC500)

Allen-Bradley DH485

LG GLOFA-GM

MITSUBISHI FX

MITSUBISHI Protocol 4

Modicon Modbus

Telemecanique Uni-Telway

GE Fanuc SNP/SNPX

Omron Host-Link/Multilink

### Repercussions for tags

The address of a tag depends on the PLC you are using. The way in which the address of a tag with a PLC connection is displayed depends on the PLC you selected.

You select the available data types and data formats in the "Tags" dialog box at "Type" or "Format".

You will find an overview of the available data types and data formats that you can select for the PLC driver concerned in the "Communication for Windows-Based Systems" user's manual.

## 4.7 Area pointers

### What are area pointers for?

A defined address area on the PLC for data interchange with the operating unit is addressed by means of an area pointer.

You create area pointers in the project window at "Area pointers". You can edit area pointers with all the commands available in the shortcut menu for example, copy and then paste in another project.

### What area pointers are available?

The number of area pointers available varies depending on the selected operating unit.

For information on how large the area pointers should be when you create them, and the structure they have to have, refer to the "Communication for Windows-Based Systems" user's manual.

The overview below lists the various area pointers and what they are used for. The area pointers are sorted in alphabetical order.

Area pointers	Explanation
User version	The user version identifies the version of the project. A version check is performed on the PLC by means of this area pointer.
Job mailbox	The PLC uses this data area to transfer jobs to the operating unit in order to trigger specific functions (to display a screen, for example).
Event messages	You can configure an event message for each bit in this data area. The bits are assigned to the message numbers in ascending order.  As soon as the PLC sets a bit in this data area, the operating unit recognizes that the assigned event message has "arrived". Conversely, the operating unit interprets the message as "departed" after the bit is reset on the PLC.
Screen number	In this data area the operating unit stores information on the current screen. You can evaluate this information in the PLC program to call another screen, for example.
Data mailbox	The data mailbox is a data area on the PLC. It is used for synchronization during data record transfer from the operating unit to the PLC.

Area pointers	Explanation
Date/Time	The operating unit writes the date and time in this data area in accordance with a PLC job. This data can be evaluated by the PLC program.
Date/Time PLC	The operating unit reads from this data area in the set date/time acquisition cycle to synchronize with the PLC.  The date and time must be stored in the PLC in the format <code>S7_DATE_AND_TIME</code> .
Coordination	The PLC can use this data area to prompt the status of the operating unit (for example, startup of the operating unit, current operating mode and communication readiness).
Trend request	The PLC can use this area pointer to determine which bit-triggered trend is currently being displayed on the operating unit.
Trend transfer area 1	This data area is used to trigger trends. As soon as the PLC program sets the bit assigned to the trend and the trend communication bit in the trend transfer area, the operating unit detects the trigger and, depending on the configuration, reads out either a value or the entire buffer.
Trend transfer area 2	This data area is required when you configure trends with a switch buffer. The data area is structured in the same way as the "trend transfer 1" data area.
LED assignment	The PLC can use this area pointer to drive the light-emitting diodes on the function keys of the operating unit.
OP Acknowledgement	The operating unit uses this area pointer to indicate to the PLC which alarm messages have been acknowledged on the operating unit.
PLC Acknowledgement	Messages can be acknowledged from the PLC using this area.
Alarm messages	You can configure an alarm message for each bit in this data area. The bits are assigned to the message numbers in ascending order.  As soon as the PLC sets a bit in this data area, the operating unit recognizes that the assigned alarm message has "arrived". Conversely, the operating unit interprets the message as "departed" after the bit is reset on the PLC.

---

**Note**

The following applies to the "Coordination" area pointer:  
You can configure more than one "Coordination" area pointer in ProTool 5.2 + SP3 or higher.

---

## 4.8 Screen partitioning on the operating unit

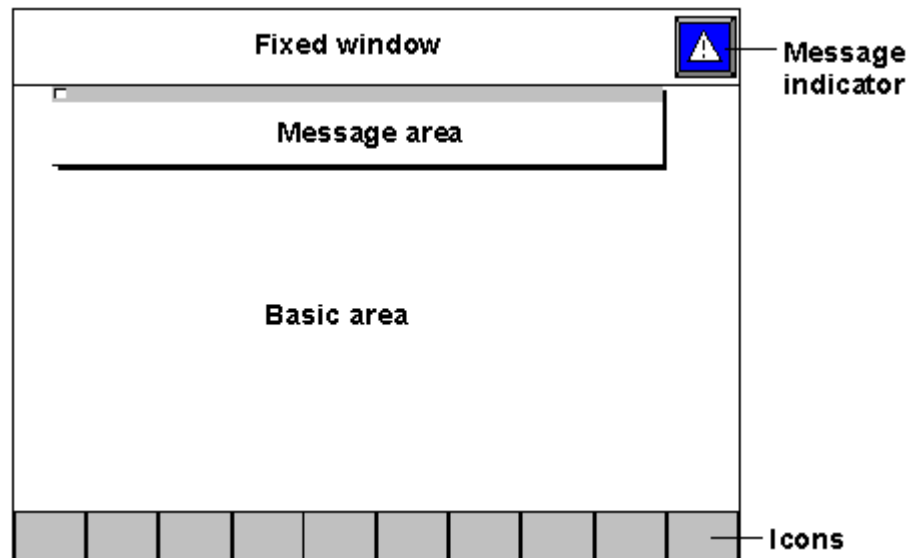
### Display areas

Depending on the operating unit for which you create your project, you can partition the display into different areas. These areas can either be moved (message area and message indicator) or they are immobile (fixed window, icons for function keys).

### Where do you set the areas?

Choose "System" > "Screen/Keys" to set the areas. The settings made here apply to the whole project, So set the areas before you begin configuration, to the extent possible.

The example below shows one possible way of partitioning the display:



*Example of partitioning the display on the SIMATIC Panel PC*

### Basic area

The basic area is the entire screen area without the fixed window. If you change the size of the fixed window, the size of the basic area automatically changes as well. All the other areas are superimposed on parts of the basic area. You configure process screens in the basic area. This means that the contents of the basic area changes for each screen you call.

### Fixed window

The fixed window is a window that is always displayed. It uses up part of the basic area. You can enable and disable the fixed window by choosing "System" > "Screen/Keys" from the menu. If it is switched on, it is displayed during screen configuration and thus reduces the size of the area available for screens. The size of the fixed window can be changed, but its position is always the same at the top of the screen. The fixed window is configured with the "Screens" editor.

### Message indicator

The message indicator is a symbol indicating alarm messages that are still applicable on the operating unit. You can enable and disable the message indicator by choosing "System" > "Screen/Keys". You cannot change the size, but you can the position.

The message indicator can be used with touch devices.

### Message area

The message area is the area in which messages are displayed on the operating unit. By choosing "System" > "Screen/Keys", you can set separately for alarm and event messages whether they are displayed on the message line or in a message window, or whether they are not displayed at all.

- Message line: A configured message line is always present irrespective of the screen that has been selected. It shows the most recent message to arrive.
- Message window: If one of the display modes "Window/Window", "Window/Line" or "Window/Off" is configured, the message window for alarm messages is automatically opened as soon as an alarm message arrives. To open the message window for event messages by means of an operator input, in your project the function `Display_Event_Message_Window` must be linked to a control.

### Icons

On SIMATIC HMI operating units having an integrated keyboard (for example, the OP 170B, MP 270 or Panel PC 670), you can place icons on the screen for function keys having locally defined functions. This is only possible, however, for the function keys arranged directly around the screen.

To find out how to assign keys globally or locally, refer to the topic "Function keys".

**Setting the display for the operating unit**

By choosing "System" > "Settings", you can set how the compiled project will be displayed on the operating unit for PCs and SIMATIC Panel PCs, for example. You can start the project either in full screen mode or in a window that is smaller than the screen. For full screen mode, check the option "Full Screen" at "Display".

In Full Screen mode, more space is available to you even while you are configuring since the window frame is hidden. Therefore, set Full Screen mode as a matter of course before you start configuring if your target device supports this option.



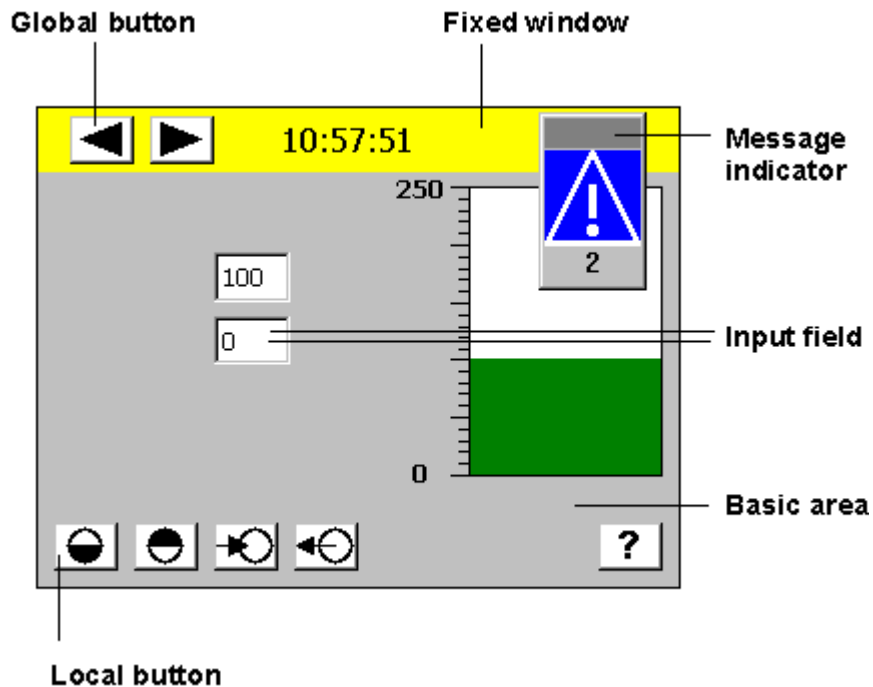
## 4.9 Configuration notes for touch devices

### Screen partitioning basics

Before you start setting touch device controls, set the basic partitioning of the screen on the operating unit. Give consideration to where messages and input windows will be opened in order to prevent unintentional overlapping of buttons and input fields.

Dimension the controls, such as buttons, large enough for them to be operated properly. For display elements, such as output fields, use font sizes that are large enough to allow comfortable reading.

The figure shows an example of screen partitioning on the TP 170B.



Example of screen partitioning on the TP 170B

### Global buttons

Position buttons which you want to be available globally in every operating situation in the fixed window. Buttons in the basic area and their assigned functions are available only locally on the screen concerned.

### Overlapping controls

Mutual overlapping of control elements can result in undefined operating states and

**On-screen keyboard**

To read in values, the touch device automatically displays an on-screen keyboard as soon as the operator touches an input field, for example. The keyboard is automatically hidden when the input is terminated. Depending on the configured input field, the on-screen keyboard features keys for purely numerical input or for entering alphanumeric characters.

**Special features of touch devices**

Input fields and input/output fields are touch-sensitive on touch devices. Configure a border for the field, or select a background color that is different from the background color of the current screen. In this way the touch-sensitive area of the field can always be recognized on the operating unit.

Help text is particularly helpful with input fields and input/output fields, since the assignment of input field to input window on the touch device is not always directly evident from the position of the input window.

## 4.10 Functions to provide support for service work

### Purpose

In your project, you should provide controls to make service work easier on startup and during operation.

It is advisable to configure these controls in a separate screen, with their selection protected by a password. In this way, you prevent operators who are not service personnel changing basic settings by mistake.

### Functions for service work

The following tasks when starting up a project or when the project is running, can be simplified by using functions:

- Opening the Windows CE Control Panel: The service technician can use the Windows CE Control Panel to set the printer, and data and time, or configure the network card.
- Changing modes: At project run time, the operating mode is usually set to "Online". In order to test a new project first without a connection to the PLC, the operating mode must be changed to "Offline", to transfer a new project to "Download mode".
- Switching languages: Where projects involve several languages, the service technician can set the language required by the actual installation when starting up or while the project is running.
- Display software version: The Hotline needs the software version of the installed ProTool components to efficiently process your inquiries.
- Displaying Task Manager: The service technician can change to other applications on the operating unit by using Task Manager.
- Opening Internet Explorer: Using Internet Explorer, the service technician can change from the world-wide web from an operating unit Windows CE.

## 4.11 Editing projects

### Useful functions when editing projects

If you edit projects in ProTool, there are several ways in which you can make work easier for you.

#### **Converting projects**

If you wish to create similar projects for different devices, you can convert a project into a project for another destination device. In this way you quickly create a basis for a new project.

In this instance, it is possible, for example, to convert a project for graphics displays to a project for Windows-based systems.

#### **Copying objects**

You can copy previously created objects from one project to another project. You do not have to create the same objects twice when you edit several projects.

#### **Undoing and redoing actions**

You can undo many of the actions performed in ProTool when, for example, you have entered a wrong value or have edited a screen object incorrectly. Similarly, you can redo actions you have undone.

#### **Substituting a project and project parts**

To use an existing project, say, as a basis for a new project or to reuse previously configured parts of a project, you can substitute whole projects in ProTool or only certain selectable project parts in a destination project.

### 4.11.1 Convertible projects

#### Windows-based system > Windows-based system

You can use menu command "File" >"Convert" to convert a ProTool project that you created for a Windows-based device to a project for a different Windows-based device.

It is also possible, within a device class, to convert from one resolution to another, or between different destination devices having different display sizes.

---

#### Note

When you convert an existing configuration for another destination device, close the converted configuration and open it again before you make changes.

---



---

#### Note

Projects for the TP 170B Mono can be converted to a project for the TP 170B Color, and the other way round.

---

You can cross convert projects for the following devices:

	TP 170A	TP 170B	OP 170B	TP 270	OP 270	MP 270	MP 270B	MP 370	OP 37/Pro	PC	PC 670	PC 870	FI 25	FI 45	OP 010	OP 012	OP 015	PC IL70	P0 12	P0 15	
TP 170A		x	x	x	x	x	x	x	x	x	x	x	x	x					x	x	x
TP 170B	x		x	x	x	x	x	x	x	x	x	x	x	x					x	x	x
OP 170B	x	x		x	x	x	x	x	x	x	x	x	x	x					x	x	x
TP 270	x	x	x		x	x	x	x	x	x	x	x	x	x					x	x	x
OP 270	x	x	x	x		x	x	x	x	x	x	x	x	x					x	x	x
MP 270	x	x	x	x	x			x	x	x	x	x	x	x					x	x	x
MP 270B	x	x	x	x	x	x		x	x	x	x	x	x	x					x	x	x
MP 370	x	x	x	x	x	x	x		x	x	x	x	x	x					x	x	x
OP 37/Pro	x	x	x	x	x	x	x	x		x	x	x	x	x					x	x	x
PC	x	x	x	x	x	x	x	x	x		x	x	x	x	x	x	x		x	x	x
PC 670	x	x	x	x	x	x	x	x	x	x		x	x	x					x	x	x
PC 870	x	x	x	x	x	x	x	x	x	x	x		x	x					x	x	x
FI 25	x	x	x	x	x	x	x	x	x	x	x	x		x					x	x	x
FI 45	x	x	x	x	x	x	x	x	x	x	x	x	x						x	x	x
OP 010																	x	x			
OP 012																x		x			
OP 015																x	x				
PC IL 70	x	x	x	x	x	x	x	x	x	x	x	x	x	x						x	x
P0 12	x	x	x	x	x	x	x	x	x	x	x	x	x	x					x		x
P0 15	x	x	x	x	x	x	x	x	x	x	x	x	x	x					x	x	

---

#### Note

Take note of the system limits of the device when you convert projects.

---

## Graphics display > Windows-based system

You can also create a new project for a Windows-based system (such as an MP 370 or Panel PC 670), by choosing the "File" > "Convert" option from the menu, from an existing project for a device having a graphics display (such as an OP 37). Projects of different resolution are also mutually convertible.

All objects that can be mapped to the new destination device can be converted. If the functionality of a given object is not supported on the destination device, the object will possibly only be converted in part or not at all. ProTool will tell you in the system message window if there are any objects and functions that cannot be converted.

---

### Note

If a project is converted from a graphics device to a 6 inch Windows CE device, the maximum number of visible list elements in a symbolic selection field should be changed to six. This ensures a neat display in every viewing position.

---

---

### Note

You will find more information about converting projects in the documentation called "Configuration support for upgrading - migration to Windows-based devices" [Projektierungsunterstützung für Umsteiger - Migration zu Windows-basierten Geräten].

---

## Settings for conversion

You can influence the conversion result by choosing "File" > "Convert" > "Settings".

- **Smoothing**  
If the "Smooth" setting is enabled, an anti-alias filter for replacing missing pixels due to interpolation is used when increasing or decreasing the size of a graphic. This setting results in softer transitions in photo-realistic displays. In the case of drawings, the display quality is enhanced when zooming out of graphics in particular, since the complete screen information is retained.
- **Converting resolution**  
Select this setting if you would like to convert projects between different operating units having different resolutions or display sizes.

## 4.11.2 Key assignment when converting

### Mapping keys when converting projects

This table provides an overview of how the keys are mapped to each other when a project is converted for another operating unit.

#### Examples:

You have configured a function on F20 on the MP 270; following conversion, this function is triggered with F8 on an MP 370 and with SHIFT + F8 on a standard PC.

If you have used F15 for a FI45 project, this function is no longer available on a standard PC following conversion.

#### Note

When converting in ProTool CS, the keys are imaged on each other irrespective of the key code.

MP 370 PC 670/PC 870 FI 25/FI 45 P0 12K/P0 15K	MP 270 OP 37/Pro MP 270B OP 270 10"	OP 270 6" OP 170B	Standard PC
F1	F13	F9	SHIFT F1
F2	F14	F10	SHIFT F2
F3	F15	F11	SHIFT F3
F4	F16	F12	SHIFT F4
F5	F17	F13	SHIFT F5
F6	F18	F14	SHIFT F6
F7	F19	--	SHIFT F7
F8	F20	--	SHIFT F8
F9	--	--	SHIFT F9
F10	--	--	SHIFT F10
F11	--	--	SHIFT F11
F12	--	--	SHIFT F12
F13	--	--	--
F14	--	--	--
F15	--	--	--
F16	--	--	--
F17	--	--	--
F18	--	--	--
F19	--	--	--
F20	--	--	--

<b>MP 370 PC 670/PC 870 FI 25/FI 45 P0 12K/P0 15K</b>	<b>MP 270 OP 37/Pro MP 270B OP 270 10"</b>	<b>OP 270 6" OP 170B</b>	<b>Standard PC</b>
S1	F1	F1	F1
S2	F3	F3	F3
S3	F5	F5	F5
S4	F7	F7	F7
S5	F9	--	F9
S6	F11	--	F11
S7	--	--	--
S8	--	--	--
S9	F2	F2	F2
S10	F4	F4	F4
S11	F6	F6	F6
S12	F8	F8	F8
S13	F10	--	F10
S14	F12	--	F12
S15	--	--	--
S16	--	--	--
--	K1	K1	--
--	K2	K2	--
--	K3	K3	--
--	K4	K4	--
--	K5	K5	--
--	K6	K6	--
--	K7	K7	--
--	K8	K8	--
--	K9	K9	--
--	K10	K10	--
--	K11	--	--
--	K12	--	--
--	K13	--	--
--	K14	--	--
--	K15	--	--
--	K16	--	--
HELP	HELP	HELP	--
ESC	ESC	ESC	--
ACK	ACK	ACK	--



<b>MP 370 PC 670/PC 870 FI 25/FI 45 P0 12K/P0 15K</b>	<b>MP 270 OP 37/Pro MP 270B OP 270 10"</b>	<b>OP 270 6" OP 170B</b>	<b>Standard PC</b>
ENTER	ENTER	ENTER	--
PAGE UP	PAGE UP	PAGE UP	--
PAGE DOWN	PAGE DOWN	PAGE DOWN	--

### 4.11.3 Example of how to convert an MP 370 project

#### Objective

You want to convert an MP 370 project to a PC project.

#### Procedure:

Procedure:	
1	Open the existing MP 370 project.
2	Choose "File" > "Convert" > "Settings" from the menu. In the "Conversion Setting" dialog box, set whether the graphics should be smoothed ("anti-aliasing") and the resolution should be adapted to the destination device. Select "Convert resolution", too, if you want to convert projects between devices of different screen resolution. During conversion, all screens will then be adapted to the resolution of the destination device.
3	Close the dialog box by clicking OK.
4	Choose "File" > "Convert" > "Start". The "Save As" dialog box appears.
5	Enter a directory and a new name for the new project.
6	From the "Device type" list, select the destination device for your conversion, in this instance PC. The list displays all available destination devices for the source project you want to convert.
7	Select the resolution of the destination device from the "Resolution" list.
8	Click "Save". The project is converted and opened in the project window.

#### 4.11.4 Copying objects: Between projects and within a project

##### Principle

You can cut or copy selected parts of your project and paste them from one project to another via the Clipboard. For example you can copy text and fields from the alarm message editor to the event message editor, or graphic elements from one screen to another.

A requirement for successfully copying from one project to another is that the system limits of the destination project are not exceeded.

You can also copy objects between projects with different resolution. If the destination system uses a lower resolution than the source system, then the icons of soft keys, for example, are cleanly adjusted to the new resolution by an anti-aliasing mechanism.

##### Preparations for new projects

Before starting to copy objects from an existing project, you should without fail carry out the following global settings in the new project. This will ensure that no loss of data occurs due to different settings during copying.

- At "System" > "Screen/Keys", adjust the partition of the screen display to that in the source project.
- Make the option for Full Screen mode the same as in the source project by choosing "System" > "Settings" from the "View" menu.
- Make the name and driver of the "PLC" the same as in the source project.
- Make the language settings the same as in the source project by choosing "System" > "Language Assignment".

##### Pasting objects in the destination project

To paste objects from the Clipboard to the destination project, choose "Edit" > "Paste" from the menu.

The object is pasted from the Clipboard to the destination project. If there is already an object of the same name in the destination project, the object is pasted under a new name.

---

##### Note

With "Paste", ProTool always checks the lower-level objects (such as the limit value tags of a tag which has been copied) to ensure that existing objects are reusable.

If there is already an object of the same name in the destination project, the object to be pasted will be renamed if necessary. It is given the next available name in the destination project.

---

**Example:**

Tag VAR\_4 will be renamed VAR\_11 if tags VAR\_1 through VAR\_10 already exist in the destination project.

**What is copied?**

You can copy the following via the Clipboard:

- All objects listed in the project window, such as screens, text or graphic lists, tags etc.
- Objects from screens (trends, fields, graphics etc.) in the screen editor.
- Messages and objects from messages (message text, fields, Help text etc.) in the message editor.
- Area pointers
- Global and local function key assignments

An object is copied complete with its attributes and all cross-referenced objects. Special situations encountered during copying are reported in the "system message window" at "Clipboard". This gives you information about any objects that have not been copied or renamed, for example.

**Special situation with screens**

If the object to be copied refers to a screen that does not exist in the destination project, the underlying screen is not copied; instead a blank screen is created as a dummy if the destination project does not contain a screen that can be reused.

This ensures that you will not copy the complete source project by mistake along with the start screen.

Afterward, when you paste the screen via the Clipboard, the dummy screen in the destination project will be automatically replaced by the proper screen.

**Special feature with tags**

Perform the steps listed below so that tags with a PLC connection can be inserted completely in the destination project when copying from one project to another:

- Assign a self-explanatory symbolic name for the PLC in the source project and do not use the system default setting (for example, `PLC_1`).
- Copy the PLC object to the destination project first so that it is present in both projects with identical parameters. Then copy the tags belonging to this PLC.

**Special feature of recipes**

With Windows-based systems, you can also copy recipes via the Clipboard. You will find more information about this at "Compatibility of recipes".

**What is not copied?**

These objects are not copied:

- Objects that are unknown in the destination project (such as functions or command buttons when copying from TP 170 to OP 170)
- Character sets
- In the case of multilingual projects, only the languages available in the destination project are copied. No new languages will be created. If necessary, create them before copying.
- Settings at "System" - for example, Language Assignment or Full Screen mode.

## 4.11.5 Replacing a project and project parts

### Purpose

If you would like to introduce new or updated standard components (say, from ProAgent) of one project in your project, you can do so with the ProTool's Replace function.

### Replacing a complete project

By choosing "File" > "Replace" > "Setting..." from the menu, you can set the object types which should be taken into account when replacing whole projects. To do this, select the object types you want.

The settings will be saved in the project; if this project is used as a source project for the substitution, the settings are taken input.

If you choose "File" > "Replace" > "Start" from the menu, you can select a project that you want to replace your currently open project. The source project must be of the same type as the destination project. You should save the active configuration before doing so.

Replacement of a complete project cannot be undone.

Using the object name, a check is made for all the objects present in the source project as to whether an object having the same name is already present in the destination project.

If there is no such object present, a new object is created with the name used in the source file. If the object is already present in the destination project, it will be replaced by the object in the source project.

Object linking is not changed by the substitution; merely the contents of the object are inserted and additional linking added.

Should errors occur in the project during the substitution operation, they are reported in the same way as other errors when the project is compiled.

---

### Note

PLCs and permanent windows are not replaced.

---

### Replacing objects

You can also copy individual project objects from a source project and paste them in another project with the "Insert/Replace" function:

- By choosing "Edit" > "Copy", you copy as many objects you like from the source project to the Clipboard.
- Choose "Edit" > "Insert/Replace" from the menu all objects in the destination project having the same name are replaced.

The replacement of individual objects and their linked (referenced) objects can be customized in a settings dialog box. This dialog box is displayed for every Replace action.

In the dialog box, you set which objects you wished to have replaced as referenced objects with replacement via the Clipboard. Select the object types you require. This setting is not saved locally for the project but globally for the application.

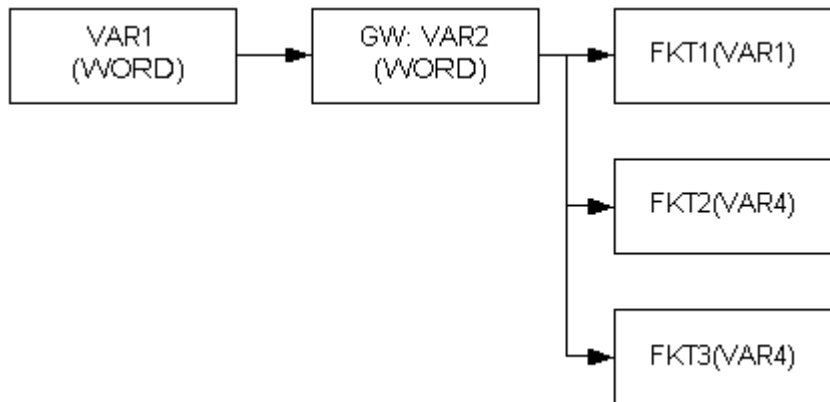
### Example

An archive has been configured for an archive. If this tag is now copied from the source project to the Clipboard and replaced in the destination project, the referenced archive object is replaced if you have selected the "Archive" object type in the "Replace Setting" dialog box and the archive object is already present with the same name in the destination project.

If the archive object is not in the destination project, a new object is created and linking of the tags expanded.

#### Destination project

The tag VAR1 (WORD data type) has a limit tag VAR2. This in its turn references the three functions FKT1, FKT2 and FKT3.



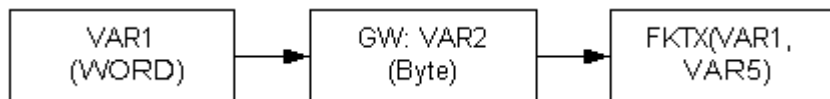
#### Source project

Tag VAR2 (BYTE data type) references the function FKTX.



#### Result

Limit value tag VAR2 has been replaced by tag VAR2 from the source project.



Object linking is not changed by the substitution; merely the contents of the object are inserted and additional linking added. All the objects referenced by the inserted objects are also pasted in the destination project.

Should errors occur in the project during the substitution operation, they are reported in the same way as other errors when the project is compiled.

The replacement of individual objects can be undone. Further, Replace is supported differently by various objects, including between projects of different types; thus, for example, you can paste objects from a graphics project in a Windows-based project.

### What you must bear in mind

You cannot insert more objects with Replace than the number allowed by the quantity structure of the destination project.

If the destination project has a smaller range of functions than the source project, it is possible that not all objects will be applied.

Only objects having the same name are replaced. If, for example, you wish to replace a recipe with 50 entries in the destination project with a recipe having 500 entries, the entries not contained in the destination project will be created.

Object attributes that have not been defined in the source project will be assigned default values in the destination project.

### Scenario 1

Accepting objects from other projects without breaking up the project structure and without time-consuming reconfiguration.

#### Example

You want to renew the contents of a picture without overwriting the contents of the tag on the screen.

1. Copy the required screen for the substitution in the source project to the Clipboard.
2. In the destination project, replace the screen of the same name in the destination project by choosing "Edit" > "Insert/Replace" from the menu.
3. Deselect the "Tags" object type in the Settings dialog box. Tags referenced on the screen by means of input/output fields, for example, are then not replaced.

### Scenario 2

Updating projects issuing from a common "basic project".

#### Example

All standard screens have been revised in the basic project, or new ones have been added to it. For example, screen elements have been inserted or deleted and tag fields adapted. All derived projects now have to be updated.

1. In the basis project, select all object types you want to have replaced in the destination project by choosing "File" > "Replace" > "Setting..." from the menu. If, say, only screens, tags and archives have been modified, only these object types should be selected.
2. Save the basic project.
3. In the destination project, select the basic project by choosing "File" > "Replace" > "Start..." from the menu. In the Settings dialog box that then opens you then select the object types for substitution again. The types already set in the destination project are offered as defaults. Object substitution starts after you quit the dialog box by clicking OK.



## 4.11.6 Undoing and redoing actions

### Purpose

During configuration, it may become necessary to cancel actions which have been performed, or to reconstruct actions which have been discarded. The two commands "Undo" and "Redo" on the "Edit" menu are used for this purpose.

- Undo  
The "Undo" command (key combination Ctrl + Z) cancels the last action performed. If you keep selecting the command, you can cancel up to 30 successive modifications.
- Redo  
The "Redo" command (key combination CTRL+ Y)revokes the last action canceled, thus restoring the status before the last "Undo" command was executed.

### Principle

Each active editor (project window, screen editor, drivers) has its own undo history. Thus, for example, if three screens are opened at the same time, three separate Undo Histories will be created. When a screen is closed, the actions listed in the accompanying History are deleted. When the project is saved, all the Undo Histories for the current project are deleted.

The last recorded action is displayed in abbreviated form in the menu. The tooltips contain detailed texts for the "Undo" and "Redo" buttons and for the status bar.

#### Example of menu

Undo:        VAR\_5 edited        Ctrl + Z  
Redo:        PIC\_2 edited        CTRL+ Y

#### Example of tooltip/status bar

Undo:        property edited of tag VAR\_5  
Redo:        contents edited of screen PIC\_2

Until the corresponding undo history is deleted, deleted objects in the cross-reference continue to be listed as used objects. The status of these objects is shown in brackets after each object - for example, PIC\_5 (deleted).

## General Information

The commands "Undo" and "Redo" only work with actions taken since the last time the project was saved. If, for example, you move a screen object and then save your project, you cannot later cancel this action.


These are some of the rules that apply to "Undo/Redo":

- Settings in dialog boxes (properties of a field) can only be canceled in their entirety. It is not possible to open the dialog box and discard individual entries.
- With multilevel dialog boxes, only changes to the primary object are recorded. Modifications to underlying objects, or creations or deletions cannot be reversed.  
Example (project window):  
"Edit Tag" > "Edit Limit Value Tag"  
You can only undo changes to the tag here.
- "Undo/Redo" is special to ProTool. With a project integrated in SIMATIC STEP 7, the Undo buffer cannot be accessed by a higher-level SIMATIC Step 7 Undo Manager.

## 4.11.7 Undoing the last action

### Possible procedures

To cancel your last action in ProTool, choose one of the three following options:


- Choose "Edit" > "Undo" from the menu.  
The last action that can be undone (canceled) is shown in abbreviated form after the menu command. A longer description is given in the status bar.
-  Click the "Undo" button on the toolbar.  
This opens a tooltip which shows you the last action that can be undone (canceled). You are given the same information in the status bar.
- Press CTRL and Z simultaneously.  
In contrast to the first two options, you are not given any feedback about which action has been canceled.

If you keep executing the "Undo" command, you can successively cancel all the recorded modifications.

## 4.11.8 Redoing the last action

### Possible procedures

To restore your last canceled action in ProTool, choose one of the three following options:

- Choose the "Edit" > "Redo" command from the menu  
The last undone action is shown in abbreviated form after the menu option. A longer description is given in the status bar.
-  Click the "Redo" button on the toolbar.  
This opens a tooltip which shows you the last canceled action. You are given the same information in the status bar.
- Press CTRL and Y simultaneously.  
In contrast to the first two options, you are not given any feedback about which action has been restored.

If you keep executing the "Redo" command, you can successively restore all the recorded cancellations.

## 4.12 Retrieving project information

### Tools

The following tools are available to you for displaying or storing information on a project.

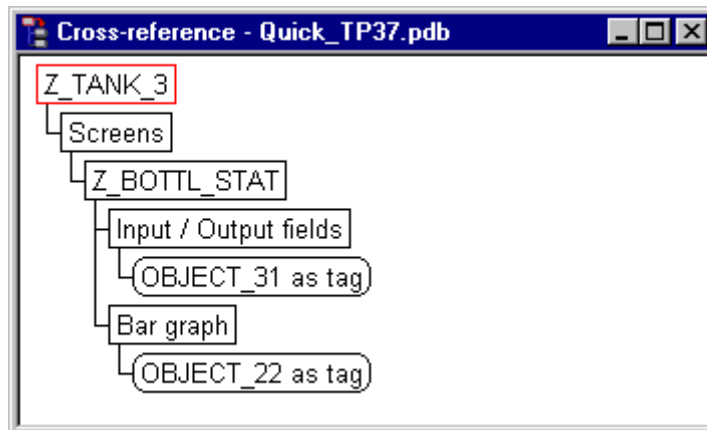
- Cross references
- Project information

### 4.12.1 Display in "Cross-Reference" window

#### Usage

When you have to add to or modify a project and need to check where and how a particular object is being used in your project, you open the "Cross-Reference" window.

You select an object in this window, and all the references to this object in the project are then displayed to you.



*Example from a project*

You open the "Cross-Reference" window by choosing the "View" > "Cross-Reference" option from the menu. The active object is displayed with a red border around it.

The selected object is at the uppermost level, and all the objects in which the selected object is used appear under it.

The cross-reference list also contains objects in the current Undo history (undo and restore actions). The status of these objects is shown in brackets after each object - for example, PIC\_5 (deleted).

---

#### Note

You can use the "Cross-Reference" window efficiently for troubleshooting.

---

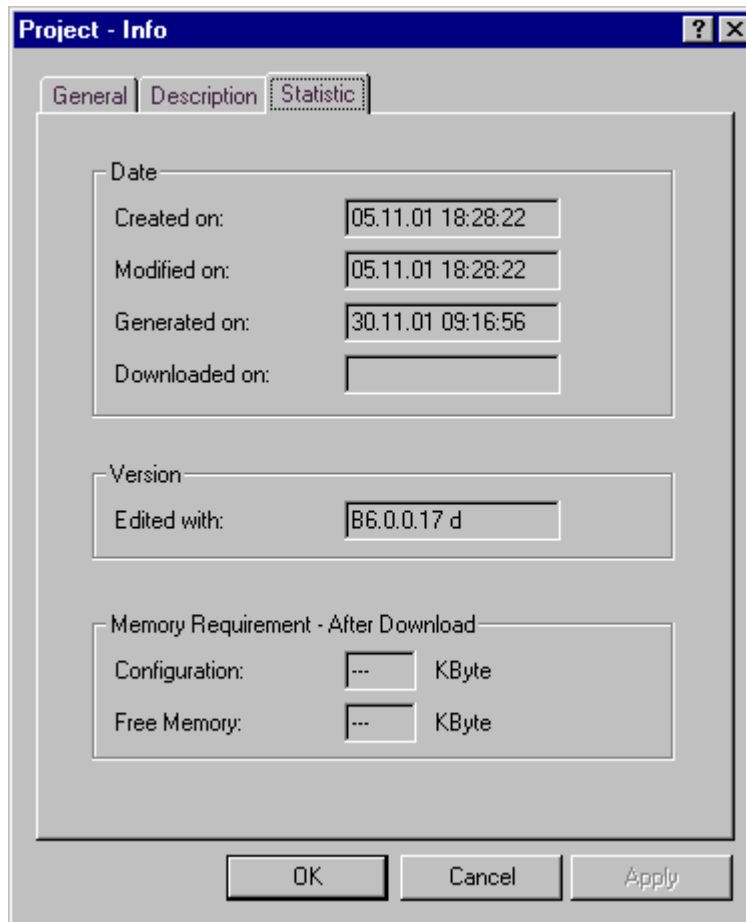
## 4.12.2 Display at "Project Information"

### Usage

To obtain information about a project when projects change or are adapted, open the "Project Information" dialog box. To do so, choose "File" > "Project Information" from the menu.

The "Project Information" dialog box displays general "project data" and the "memory requirements" for the project. Project information is spread according to subjects over three tab controls:

- General
- Description
- Statistics



*Example relating to the "Statistics" tab*

**General**

The "General" tab shows information on the device type, project name, path name of the stored project file and person who created the project. You fill in the "Created by" field and all the other fields are updated automatically by ProTool when the project is saved.

**Description**

The "Description" tab contains an input field for the project description. Here you can enter any information you like that are important for your project.

**Statistics**

The "Statistics" tab shows when the project was created, modified, generated and downloaded, the ProTool version last used to edit the project and the memory required by the project after it has been downloaded to the flash memory on the operating unit. The memory requirement is determined and displayed after every download operation.

## 5 Configuration techniques

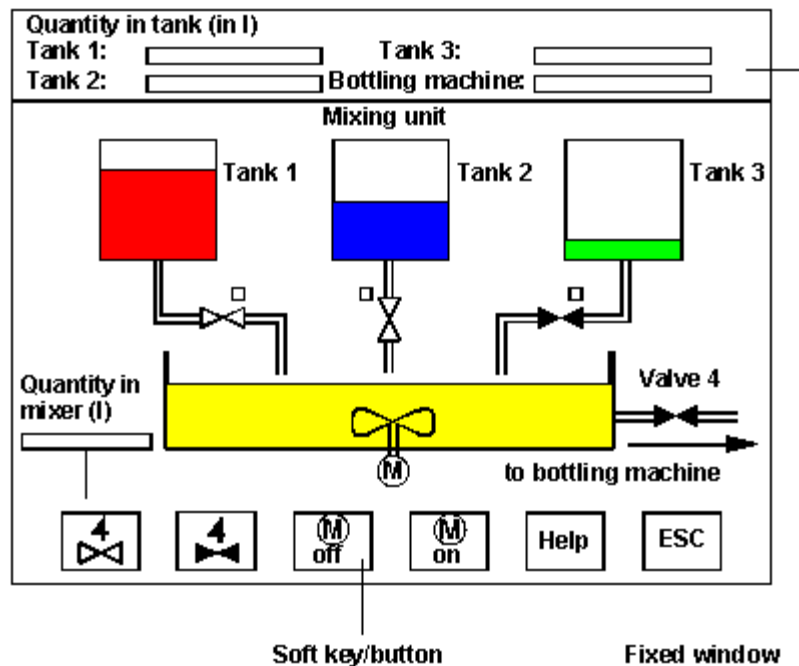
Here you will learn how to:

- Create screens
- Use tags
- Configure controls and display elements
- Configure schedulers and trends
- Create reports
- Configure messages
- Use functions
- Create archives and recipes
- Configuring operator prompting
- Creating VB scripts
- Configuring in foreign languages

## 5.1 Create screens

### Example

Screens are an image of the process. You can display processes on screens and specify process values. The figure below shows an example of a mixing unit for producing different juices. Ingredients are filled into a mixer from different tanks and then mixed. The liquid levels in the tanks and in the mixer are displayed. The intake valves can be opened and closed by means of the operating unit. The motor for the mixer can be turned on and off in a similar manner.



Example of a Screen: Mixing Station

### Components of a screen

A screen can consist of static and dynamic components. Static components - for example, text and graphics - are not updated by the PLC. Dynamic components are linked to the PLC and visualize current values from the PLC memory. Visualization may take place in the form of alphanumeric displays, trends and bar graphs. Dynamic components are also inputs made by the operator on the operating unit and written to the PLC memory.

The connection to the PLC is established by means of tags.



## Screen editor

Screens are created with a separate editor in ProTool. The operating unit is displayed when you call the screen editor. Open the screen editor by:

- Double-click on "Screens" in the left half of the project window to create a new screen.
- Double-click in the right half of the project window on an existing configured screen to open the screen for editing.

By choosing "View" > "User Interface" or the buttons on the toolbar, you can zoom in and zoom out of the display in steps. If, for example, you wish to edit details you can do it much more simply by zooming in on them.

---

### Note

If possible, you should always configure in 100% Mode, since the display will correspond to the display as shown later at run time and you thus obtain a most precise preview.

---

To obtain an overview of the complete area that you can process, select "Complete screen area" in the "User Interface" dialog box or on the toolbar.

Screens are stored under a symbolic name. You enter a name by choosing "Edit" > "Properties" from the menu. This name has to be specified when you edit, reference or delete the screen. In addition, screens are numbered automatically.

## Start screen

Declare one screen in every project as the "start screen". The start screen is the screen that is displayed after the operating unit has started up.

## Fixed window

The fixed window is the window that is always flush with the top border of the operating unit screen. By choosing "System" > "Screen/Keys" from the menu, you can open and close the fixed window and adjust its height by dragging with the mouse. Since the contents of the fixed window do not depend on the current screen, you can output important process tags or the date and time to it, for example.

You configure the contents of the fixed window in the screen editor. To access the fixed window, click it with the mouse.

## Soft keys/buttons

Soft keys are function keys with screen-dependent assignment. Buttons are virtual keys on the operating unit screen. You can configure both soft keys and buttons for specific screens. When configuring, you assign functions to these controls. You can illustrate the task of a control by means of a graphic or text.

You can use soft keys and buttons to open another screen, turn a motor on and off or display the message buffer, for example.

## Select screen

Every configured screen has to be integrated into the control process so that it can be opened at run time on the operating unit. For this there is the `Select Screen` function. You can assign this function to an input field, a function key or a button, for example. As parameters, specify the screen name and the field in which you want the cursor to be positioned following a change of screens. This means that a screen can be displayed by means of an input field or a function key or button.

With input fields, soft keys and buttons, the function can be used only locally on that screen. Should you wish the function to be available globally, you have to configure the function on a globally assigned function key or on a button positioned in the fixed window.

## 5.1.1 Screen objects in ProTool

### Introduction to screen objects

Screens consist of individual objects. There are different types of object, which you can use at will when configuring a screen. You can determine the number and type of the objects, as well as their position and size.

### Libraries

Libraries are collections of preconfigured screen objects. In the screen editor, you can insert any library object into the current screen and edit it. You can integrate suitable screen objects from a screen to a library.

---

#### Note



Under the illustrated object, there are other pre-configured screen objects for Windows CE devices, Panel PCs and standard PCs in the "SIMATIC HMI Symbol Library".



---









### Vector graphics










A vector graphic is a rectangle or a line, for example. You will find an overview of objects for vector graphics in the topic "Vector Graphic Elements".









### Available screen objects

The following table contains you short descriptions of the most important screen objects. Not all the screen objects listed here are necessarily available for every destination device.

Screen object	Name	Description
	Graphics	With the graphic field, you can use graphics from other graphics programs in your project.
	Text	You can configure static texts with the text box. Various formats are available. You can display text in different fonts, enter several lines of text and align it vertically or horizontally.

Screen object	Name	Description
	Output field	With the output field you can display process values directly. Different display formats - for example, decimal and binary - are possible. Output fields can be formatted (such as 3D effects, colors, flashing, etc.).
	Input field (for alphanumeric values)	With the input field, the operator can enter values of different formats directly on the operating unit. You can also set limit values for the input or protect the input field by means of a password.
	Date/Time	With Date/Time you can display and set the system time numerically.
	Symbolic output field	A symbolic output field shows text in place of a numerical value. The assignment between the text and the values of the tags is configured in a text or graphic list (text list or graphic list).  Example: Instead of the values 0 and 1, you can output the texts <code>Motor OFF</code> and <code>Motor ON</code> on the operating unit.
	Selection field (input field for symbolic values)	In selection fields, the operator selects a block of text for the input from a "text list".  Example: Instead of the values 0 and 1, the operator can select the text <code>Motor OFF</code> and <code>Motor ON</code> to control the motor.
	Viewing a graphic	With the graphic view, you can dynamically position graphics from other graphics programs in your project.
	Graphic list	The graphics list shows a graphic from a list irrespective of the PLC status.  Example: Instead of the values 0 and 1, you can display a graphic for a closed valve and a graphic for an open valve.  

Screen object	Name	Description
	Selecting a graphic	In graphics selection fields, the operator selects a graphic for the input from a "graphics list". Example: With a graphics selection list, you can design the user interface independently of any language, for example.
	Button	Buttons are objects that trigger configurable functions by clicking a mouse or using a touch-sensitive screen. Buttons can be freely assigned a graphic or a block of text.
	Status button	The status button is used to input and display two states: ON and OFF or Pressed and Not pressed. You can configure it with or without latching.
	State area	The status area is a display element. Depending on the configuration, you can configure a warning indication or read out the status of a unit of equipment that cannot be seen from the operating unit (such as a motor).
	Switches	The switch is used to input and output a binary status. It can only be switched on or switched off.
	Invisible button	With invisible buttons, their contours are not visible. The button can, for example, be placed over a graphic, to allow indirect operation of graphic elements. You trigger configurable functions by touching an invisible button.
	Trend view	You can present several trends representing value histories simultaneously in trend view. The axes of the trend view can be configured (scales, units, etc.).
	Bar graph	Bar graphs display a value from the PLC as a rectangle. You can use it to display fill levels, for example.
	Message view	In the message view, the operator can visualize selected messages or Message events from the message buffer or message archive. To display the message archive, you must have configured a message view. All the features offered for message display by default only access the volatile message buffer.

Screen object	Name	Description
	Single message view	The single message view features a subgroup of the functions of the message view. You can use it to implement, for example, a message line in a screen.
	Status/Force	Using the "Status/Force Tag" screen object, can read and write directly to individual address areas on the connected SIMATIC S5 or SIMATIC S7.
	Password list	In ProTool you can protect entry operations by means of passwords. Passwords have to be assigned for permission levels 0 (no password) to 9 (administrator). The password list is required to enter the passwords for the different permission levels on the operating unit.
	Recipe view	With recipe view, the operator can display and edit data records on the operating unit.
	Digital/analog clock	With digital/analog clock, you can either display the system time on the operating unit as a clock with hands, or digitally.
	Slider controls	Using the slider, the operator can enter and read off numerical values.  When entering values, the operator slides the slider, for example, with the mouse or the arrow keys, to the desired position.  When it is used as a display element, the position of the slider indicates a current value from the PLC.
	Analog display	Use the analog display to display numerical values as a pointer instrument.
	SIMATIC HMI Symbol Library	This is a comprehensive library containing Graphics covering the areas of engineering and production.  The library contains, among other things:  valves, motors, tanks conveyor belts heat exchanger units and heating devices ISA symbols

---

**Note**

The following applies to the TP 170A, TP 170B and TP 270-6":

If you change from one input field to another within a screen as a result of an operation, thus resulting in the on-screen keyboard being displayed, the "Exit field" event for the previous field is not triggered immediately but only after the on-screen keyboard has been closed.

---

## 5.1.2 Libraries in ProTool

### Purpose

Libraries are collections of preconfigured screen objects. They add to the number of screen objects available and enhance the effectiveness of your configuration by repeated and multiple use of "finished parts".

### Standard libraries

Once ProTool is installed, the following libraries are available to you in the `\Library` directory:

File name	Content	Suitable for devices
Bmp-Graphics.lib	Technical standard symbols	All
Vector-Graphics.lib	Pipes, pumps, tanks and valves	All Windows systems, except TP 170A
TOUCH-Switches.lib	Buttons	All Windows systems, especially for touch-operated devices, except TP 170A, TP 170B, Mobile Panel 170
TP 170A-TOUCH-Switches.lib	Buttons	All Windows systems, especially for touch-operated devices such as TP 170A, TP 170B, Mobile Panel 170, TP 270, MP 270 TOUCH, MP 370 TOUCH Note: All objects in this library are displayed in gray tones.
TP 170B-Color-Touch-Switches.lib	Buttons	TP 170B, TP 270, MP 270B TOUCH, MP 370 TOUCH, MP 370 15" TOUCH

You can add further libraries to this directory yourself and then use them in ProTool. Files must be of the `*.lib` type and have the same format as a ProTool library.



## Managing libraries



As soon as you open a screen in a project, you have access to the menu options under "Edit" > "Libraries" and to the icon button.

Once you open the first library, the library window on the screen is active. The first library opened and all those subsequently opened are shown together here, as a tree structure.

The following functions are available to assist you in managing your libraries:

- Adding a library: You insert a file (of the \*.lib type or in ProTool library format) in the \Library directory. Use the "New" option on the shortcut menu to create a new, empty library. Save a library under a different name and then edit it.
- Renaming or deleting a library: Use the Windows functions in Explorer of the library.
- Deleting library objects or folders: Use the shortcut menu of the library.
- Restructuring a library: Edit the overview by dragging with the Windows functions. Use "New Folder" on the shortcut menu to add a new folder. To rename folders or library objects, click twice or press "F2".

---

### Note

You cannot edit a write-protected library.

---

## Editing library objects

You cannot edit library objects directly, you can only edit them in a project.

Paste the object in question to a screen in your project, edit it there and move it back to the associated library. Delete the original object and give the one you have just added the same name as the old one.

### 5.1.3 SIMATIC HMI Symbol Library

#### Purpose

There is also the "SIMATIC HMI Symbol Library", in addition to ProTool's own libraries. You open this library by means of an icon button on the "Screen Objects" toolbar.

The SIMATIC HMI Symbol Library provides you with more options for the varying and realistic design of screens in your project. You can find virtually all screen objects (graphics) in a whole number of categories dealing with subjects on technology and production, such as:

- valves, motors, tanks
- production lines
- factory buildings
- 3D ISA symbols
- refrigerator units and heating installations

#### Settings

In the screen editor, you can paste any object from the SIMATIC HMI Symbol Library into the current screen and edit it. As well as the usual tabs which you get for all screen objects (Color, Attributes, Functions, Position, Name, Help text and Enable), there are two more tabs for the SIMATIC HMI symbol library:

- Symbol: Selecting the required symbol
- Style: Settings for color fill, transparency and alignment of the screen object, as well as configuration of a tag:  
You can modify the appearance and manner in which a symbol and its background are displayed.  
You can mirror the symbol and/or rotate in steps of 90°.  
You can give the cursor above the screen an additional lightning symbol to make the configuration of a function clear to the operator, for example (only when using the mouse).  
You can configure a tag to the symbol and thus control the display of colors as a function of a limit value range not being achieved or being exceeded.

The preview automatically shows you how the symbol will appear in ProTool/Pro RT.

## 5.1.4 Colors of screen objects

### Color attribute

You can assign one or more colors to each screen object created in ProTool. Depending on the screen object, you can configure different colors for the following color attributes, for example:

- Background
- Foreground
- Exceeding the limit values

### Colors

You can allocate a color to each color attribute. Several colors are available to you for this assignment:

- Default colors  
16 default colors are permanently defined and cannot be changed.
- Project colors  
You can select any 8 project colors you like from a range of colors. These colors are globally available throughout the project. To define project colors to your range of colors, choose "Tools" > "Colors" from the menu or double-click in the "Colors" toolbar on one of the project color fields in Colors.
- Different colors  
For each color attribute, you can also select a specific color from a range of colors. This color applies to the current color attribute only. It is not globally available in the entire project and cannot be selected on the "Colors" toolbar. To assign a specific color, click the "Different Color" button on the "Color" tab, .

### "Colors" toolbar

The "Colors" toolbar is available to you in the screen editor for rapid selection of colors. It contains the following elements:



- 1 = selected foreground color
- 2 = selected background color
- 3 = 16 invariable standard colors
- 4 = 8 definable project colors

---

**Note**

Screen objects (message views, trend graphics, recipes, buttons, etc.) are also affected by the color settings on the control panel. In Windows 2000 the default color settings on the Control Panel have been changed compared to earlier operating systems (Windows 98 and Windows NT). Screen objects, for example, are shown in a lighter gray than the background.

To adjust it again, select the "Appearance" tab at "File" > "Settings" > "Control Panel" > "Display".

Select the setting "Windows Classic" from the list shown at "Scheme".

---

## 5.1.5 Fonts

### Purpose

In all screen objects which contain static or dynamic texts, you can assign parameters to the appearance of the texts, to mark the significance of individual texts within a screen. For example, you can select the font, the font style and the font size and set additional effects, such as strikethrough and underscore.

### Settings

Perform all the settings in the Properties dialog box of the relevant screen object on the "Font" tab. In the case of screen objects with a number of texts to which parameters can be assigned (such as message view, status/control, recipe view) you can set different font attributes for titles and basic text, for example. The fonts available depend on the actual device that has been set. Which font properties are available will depend on the font selected.

---

### Note

Text markings, such as font style (bold, italic, etc.), and effects (strikethrough, underscore) always refer completely to the selected screen object property. This means, for example, that you can display all the titles within a screen object as bold, but not the individual letters or words in a title.

---

## 5.1.6 Custom configuration user interface in ProTool

### Custom configuration user interface

To configure display elements and controls in a user-friendly manner, you can create the configuration user interface in ProTool so that it satisfies your requirements.

Select "View" > "User Interface" from the menu while the screen is open . You can perform the following settings in the "Interface" dialog box:

- At "Zoom", set the zoom factor of the current screen in percent.
- At "Field grid", set the density of the grid on the current screen.
- At "Display", select whether and how you want to have the grid displayed.

---

#### Note

You can perform the zoom functions in conjunction with CTRL and the numeric keypad, too:

CTRL + "+": Zoom in  
CTRL + "-":  
CTRL + "\*":

---

## 5.1.7 Date/Time

### Usage

The date and time details can be displayed and changed on the operating unit.

The following facilities are available for this:

- **Date/time display:** You can use the date/time display for screens and logs. This means that you can adjust the date and time on the operating unit or in tags. You can use tags to assign start values for the date and time.
- **Digital/analog clock:** The digital/analog clock is purely a display element. This allows you to choose between digital and analog display.
- **PLC jobs:** You can use PLC jobs 14 and 15 to set the date and time on the operating unit, and PLC jobs 40 and 41 to download the date and time to the PLC.
- **Input/output field:** Configure an input/output field with a tag in a time or date format if a date/time display is not available for your operating unit. When an input field is used for the date and time, the operator can enter the date and time on the operating unit.

---

### Note

The actual date and time format depends on which setting you have selected in your Windows system.

---

### **5.1.8 Examples of how to create screens**

In this example we will show you how you insert a coolant pump from a library in a screen.





### 5.1.8.1 How to paste a coolant pump in a screen

#### Objective

You can use the SIMATIC HMI symbol library to paste in a screen, say, a coolant pump that flashes as a function of the value of a certain tag.

#### Procedure:

Procedure:	
1	Open the screen in question.
2	 <p>On the "Screen objects" toolbar, click on the adjacent button and then in the screen, at the place where you want the object to appear. The "Symbol Library" dialog box opens.</p>
3	<p>On the "Symbols" tab, select the "pumps" category at the bottom left The "coolant pump" is selected on the right-hand side.</p> 
4	<p>On the "Style" tab, change the "Foreground mode" to "Shading" and set the "Background type" to "Opaque". The foreground color cannot be changed in the "Shaded" Foreground mode.</p>
5	Select on the "Color" tab the colors for the foreground, background and flashing of the coolant pump.
6	Select the "Attributes" tab to perform the settings for flashing.
7	Add a decimal value of 0 and assign, at "Flashing", <i>Off</i> and <i>Red</i> as the foreground color and <i>Green</i> as the background color.
8	Add a decimal value of 1 and assign, at "Flashing", <i>On</i> and <i>Green</i> as the foreground color and <i>Red</i> as the background color.
9	Click "Force" and link a control tag to the attributes. Confirm by clicking OK.
10	Configure an input field for the control tag on the same screen.
11	Declare the screen to be your start screen in the project and compile the project.
12	Start "ProTool Runtime". ProTool/Pro RT appears on the screen.
13	Enter the value 1 in the input field. The coolant pump starts flashing.

#### Note

If you enable flashing on the "Color" tab and on the "Attributes" tab, the setting enabled at "Attributes" takes precedence.

## 5.2 Configuring display elements and controls

### Purpose of display elements

The operator uses display elements to monitor the machine or the system on the operating unit. You can view current information, such as actual values from the PLC, process and operating modes and malfunctions, on the operating unit as a numerical value, in plain language or as a graphic.

### Purpose of controls

The operator uses controls on the operating unit to intervene directly in the process. He uses them, for example, to specify setpoints, trigger functions, open screens and acknowledge messages.

You can assign a password level to controls in order to prevent operation by unauthorized persons.

### Display elements in ProTool

Two types of display element are available in ProTool:

- **Static display elements**  
Static display elements are pieces of text and graphics that are not linked to the PLC. They do not react to user inputs and cannot be modified at run time on the operating unit. Use static display elements, for example, for system screens or for textual or graphic explanations for controls and dynamic display elements.  
Static display elements are
  - Static text
  - Vector graphics
  - Graphics
- **Dynamic display elements**  
Dynamic display elements can be linked to the PLC by means of tags. They visualize current values from the PLC in alphanumeric or graphic form. Dynamic display elements can change their display spontaneously at run time on the operating unit without the operator intervening. Use dynamic display elements for all tasks associated with monitoring the process, a machine or the system.  
Dynamic display elements are
  - Output fields
  - State area
  - Bar graphs
  - Trend view
  - Date/time display
  - Digital/analog clock
  - Analog display
  - Objects from the SIMATIC HMI Symbol Library

### Controls in ProTool

The following controls are available in ProTool:

- Input fields
- Function keys
- Buttons
- State buttons
- Switches
- Slider controls

Dynamic attributes can be used to indicate - for example, by means of a change of color or by flashing - that an operator input is expected at the control in a certain situation. In addition, you can display and hide control elements controlled by an event on the operating unit to make operation easier.

---

#### **Note**

To many flashing and transparent objects can have a negative impact on the screen change frequency and updates.

---

## 5.2.1 Output fields (text/graphic list)

### Purpose

Output fields display current values from the PLC on the operating unit. The values may be output optionally in numerical, alphanumeric or symbolic form.

### Numerical and alphanumeric output

Output fields for numerical or alphanumeric values show the value as a number or as text. A numerical value, for example, is the number 80 as the actual value of a temperature. An example of an alphanumeric value is the text string Valve\_12.



You configure output fields for numeric and alphanumeric values on the open screen in the "Screen Objects" toolbar by choosing the button shown.

### Symbolic output

Output fields for symbolic values do not display the true value but, optionally, a text string or a graphic from a text or graphic list. For example, you can store the two states of a valve in a text list or in a graphic list. When the valve is open, the output field then points, for example, to the text string OPEN or to a corresponding graphic.

By using output fields for symbolic values, you eliminate misinterpretations on the part of the operator to a large extent, since a symbolic value often presents a state more lucidly than an abstract numerical value, for instance. With graphics, there is the added advantage of your being able to configure the symbolic output independently of any language.



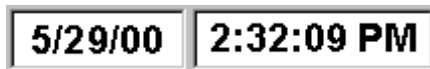
You configure output fields for text on the open screen in the "Screen Objects" toolbar by choosing the button shown.



You configure output fields for graphics on the open screen in the "Screen Objects" toolbar by choosing the button shown.

### Output field for date and time

Using the output field for date and time, you can display the calendar dates and time specifications in the selected format. The format depends on the language currently set on the operating unit. It complies with international conventions. The figure shows an example of one output field for the time and one for the date for English-speaking countries.



The output field for the date and time is especially useful for operating units such as the TP 170 A, whose range of functions does not include the "Display Date/Time" screen object.

**Array tags**

You can configure array tags for input and output fields. Using an index, you can address a specific input or output field in this way. You can use array tags configured for input and output fields for mapping a recipe structure to the screen configuration, too.

## 5.2.2 Input fields (selection field)

### Purpose

In input fields the operator enters values on the operating unit that are transferred to the PLC. The values can be optionally numerical, alphanumeric or symbolic. If you define limit values for the input field tag, operator inputs on the operating unit that are outside the specified range of values are rejected.

### Input fields

In input fields for numerical and alphanumeric values you enter the value one character at a time on the operating unit. A numerical value, for example, is the number 80 as the setpoint for a temperature. An example of an alphanumeric value is the text string `Valve_12`.



Input fields for numeric and alphanumeric values are created on the open screen on the "Screen Objects" toolbar by choosing the illustrated button or at "Insert" > "Input Field".

You can configure, among other things, the following displays for input for use with special tasks:

- Password  
The string you enter is displayed in the input field with placeholders (\*).
- Date and time  
The values you enter for the date and time are formatted depending on the language.

### List boxes

In list boxes you do not enter the value one character at a time but select this value on the operating unit from a list. On the list, you assign the value to every symbolic entry of a tag during configuration. You can configure two different types of selection field:

- Text  
The selection field consists of a text list. For example, you can turn a motor on and off by means of the two entries `ON` and `OFF`.
- Graphics  
The selection field consists of a graphics list. This means that you can configure language-independent selection lists, for example.

By using list boxes, you prevent misinterpretations to a large extent, since the operating unit accepts only the configured values on the text list.



You create a selection field for text on the open screen in the "Screen Objects" toolbar by selecting the illustrated button or by choosing "Insert" > "Selection Field" from the menu.



You create a selection field for a graphic on the open screen in the "Screen Objects" toolbar by selecting the illustrated button or by choosing "Insert" > "Graphic Selection Field" from the menu.

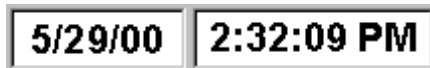
### Array tags

You can configure array tags for input and selection fields. Using an index, you can address a specific input or selection field in this way. You can use array tags configured for input and selection fields for mapping a recipe structure to the screen configuration, too.

### 5.2.3 Input fields for date and time

#### Purpose

Using the input field for date and time, you can enter calendar dates and time specifications in the selected format. The format depends on the language currently set on the operating unit. It complies with international conventions. The figure shows an example of one input field for the time and one for the date for English-speaking countries.



#### Usage

The input field for the date and time is especially useful for operating units such as the TP 170 A, whose range of functions does not include the "Display Date/Time" screen object.

#### Tag linking

Depending on the display you chose - `date` or `time` - you can configure a tag of the relevant type for the input field. The operating unit writes the values you enter to the tag. In addition, the tag presents you with the option of assigning a start value to the input field.

---

#### Note

In order to be able to change calendar dates and specifications of times on the operating unit with the input field, the configured tag must have a connection to a SIMATIC S7. The system time on the operating unit cannot be directly changed by means of the input field but must be implemented by means of the "Date/Time Display" object.

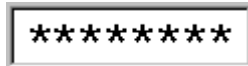
---



## 5.2.4 Input fields for concealed password entry

### Purpose

With the input field for passwords, the operator can conceal passwords when they are entered on the operating unit. The string you enter is displayed in the input field with placeholders (\*). The figure shows an example.



### Usage

The input field for passwords is particularly suitable for the following applications:

- You want specifically to log on in the start screen of your operating unit, for example, without calling a password-protected function beforehand which will automatically open the login window.
- You want to log in on the TP 170 A.

### Tag linking

You can configure a `STRING` type tag for the input field. The operating unit writes the string you enter to the tag.

If you link the tag to the function `Logon_User`, the operating unit deletes the tag contents as soon as it detects a valid password has been entered. This means that unintentional evaluation or display of the password elsewhere is avoided.

Should you still require the password as a tag in order, for example, to evaluate the tag on the PLC, configure for the "Change Password" event the global function `Write_Password_To_Tag`. This makes sure that the tag only contains the valid password and not input that could be faulty.

## 5.2.5 Function keys

### Purpose

A function key is a key on the operating unit for configuring a function assignment. You can assign one or more functions to any function key in ProTool. The functions are triggered as soon as the key on the operating unit is pressed. The function key assignment may be locally or globally significant. You can assign a password level to function keys in order to prevent operation by unauthorized persons.

---

### Note

Do not operate several function keys/soft keys or buttons simultaneously. Unintended actions might otherwise be initiated.

---

### Global assignment

Globally assigned function keys always trigger the same function, irrespective of the current control situation. You can open a specific screen, for example, display queued messages or print the contents of the screen.

By using globally assigned function keys, you cut your configuration effort considerably since you do not have to assign individual functions to global keys on every screen.

You assign function keys globally in the "Screen/Keys" dialog box, which you open by choosing "System" > "Screen/Keys" from the menu. To do so, click in the "Screen/Keys" dialog box on the function key you require. You can specify which function you have assigned to a specific key on labeling strips, as required, or you assign an informative icon to the function key.

---

### Note

By choosing "System" > "Screen/Keys" you can specify global settings for keys that apply to all screens. The icons appear in each configured screen provided the settings are not subsequently overwritten by an individual screen. In that case, the changes apply only to that particular screen.

---

### Local assignment

Locally assigned function keys trigger different actions on the operating unit depending on the screen. A function key whose assignment may vary from screen to screen is referred to as a "soft key".

Assign soft keys locally in the screen editor. Click the function key you require for this on the screen. With most SIMATIC operating units, you can assign to any soft key an icon that illustrates the soft key function by means of text or a graphic. Icons can, of course, only be assigned to soft keys positioned directly on the screen of the operating unit.

You can also assign soft keys globally. A global assignment is active on all screens to which you do not assign a screen-specific function. Use globally assigned soft keys, for example, to change from any screen to the same system screen.

---

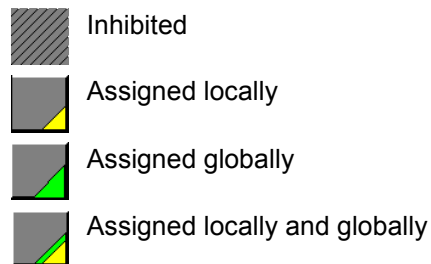
**Note**

Do not assign functions that have to be permanently available on the operating unit to soft keys.

---

**Displaying assignments**

You can see in the "Screen/Keys" dialog box and in the screen editor how your configured soft keys are assigned:



Depending on whether it is a case of function keys, soft keys or soft keys with LEDs, the display can vary slightly from the representation.

**Special feature upon change of screen**

The following special points apply, since the key assignment changes after a change of screens:

- All functions are executed immediately on leaving the screen when the event "Release key" occurs, even when the key is still pressed.
- When the function "Set Bit When Key Is Pressed" is configured for a function key, the bit is reset immediately on exiting the screen.

## 5.2.6 Trend view

### Purpose

A trend view is a dynamic display element. Trend views make possible a particularly lucid form of continuous display of process data on the operating unit.

You can display several different trends simultaneously in one trend view on the operating unit.



You configure a trend view on the open screen in the "Screen Objects" toolbar by selecting the button shown.

## 5.2.7 Bit trigger functions

### Principle

When a screen containing a trend view with one or more bit-triggered trends is opened on the operating unit, the operating unit sets the bits assigned to the individual trends in the "trend request area" in each case. From the bits set, the PLC can evaluate which trends are currently being displayed on the operating unit.

If the PLC program now sets both the bits assigned to the trends and the trend communication bit in the "trend transfer area", the operating unit detects the trigger. It resets the trend bits and the "trend communication bit" and, depending on the configuration, reads out one value or the entire buffer.

As soon as the conditions for a new trigger have been met, the PLC program sets the trend bits and the trend communication bit again.

Once the screen has been deselected, the operating unit resets the bits set in the trend request area.

Define the trend request area and trend transfer area in the project window at "Area pointers". You will find a description of these data areas in the "Communication Manual".

---

### Note

The position of the communication bit in the trend transfer area depends on the PLC you set. Do not use this bit for the identification of bit-triggered trends.

---

## 5.2.8 Bar graph

### Purpose

A bar graph is a dynamic display element. Bar graphs display a value from the PLC as a rectangular area. This means that you can tell at a glance on the operating unit how far the current value is from the limits or whether a specified setpoint has been reached. Bar graphs can be used to display fill levels or quantities, for example.



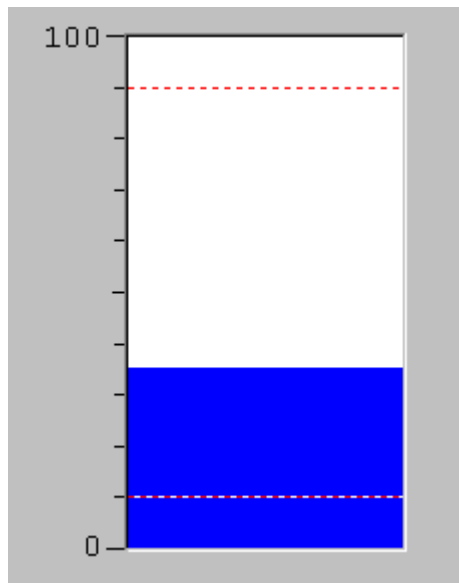
Configure bar graphs by selecting the symbol illustrated.

### Settings

Bar graphs are linked to the PLC by means of a tag. Apart from this tag, which contains the value that is required to be displayed, you can define other tags in order, for example, to visualize when a limit value has been reached or to display or hide the bar field.

You can freely define the direction, scaling, bar and background color and also labeling of the Y axis. In addition, you can show limit value lines to indicate limit values.

The following figure shows a bar that is updated vertically and has limit value lines drawn on it.



*Example referring to a bar graph*

## 5.2.9 State areas

### Purpose

The state area is a display element. Depending on the configuration, you can configure a warning indicator, signal the status of a unit of equipment that cannot be seen from the operating unit (such as a motor), or configure a light indicator whose color changes or flashes as a function of the status concerned.



You create state areas by selecting the symbol shown or by choosing "Insert" > "State Area" from the menu.

### Response

You can configure the response of the state area. Select one of the following types for this:

- One state: A state area can only display one state, such as a warning or tip, for example.
- Two states: The state area has a stable idle condition (Motor OFF, for example). When a state changes (Motor ON in our example), it changes to "ON" state and remains in that state until there is another change of state.

### "One state" type

For a state area with just one state you can set a range of values in which the state area is displayed or hidden. In conjunction with a tag representing the fill level of a tank, for example, you can configure the display of a warning.



## "Two states" type

You use a "two states" type state area to display the status of a unit of equipment on the operating unit.

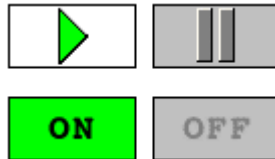
- **Tag connection:** For the "two-state" type state button, you can configure a tag which represents the current state of the state area and can also change it. If you wish, you can define a value for the configured tag that represents the "ON" state (not for BOOL type tags). The table shows which state the state area assumes as a function of the tag:

Status	Value (BOOL tag type)	Value (integer tag)
On	TRUE (1)	Configured value $n$
Off	FALSE (0)	Each of $n$ different values

When linked to an integer tag, the state area shows the "ON" state only if the tag accepts the configured value  $n$ . With any other value, the "OFF" state is displayed.

- **Text/graphic display:** For the two states "ON" and "OFF", you can configure different blocks of text or graphics which visualize the current state of the state area.

The figure shows differently configured state areas.



## 5.2.10 Buttons

### Purpose

A button is a virtual key on the operating unit screen to which you can assign one or more functions. You operate a button with a pointing device, such as a mouse, or by hand in the case of touch-operated devices.



You create buttons on the open screen in the "Screen Objects" toolbar by selecting the button shown.

### Labeling

You can label buttons statically or dynamically with text or graphics.

- **Static labeling:** When configuring the button, enter a text or select a graphic. This type of labeling is static, which means that it is not changed on the operating unit.
- **Dynamic labeling:** When configuring the button, select a text or graphics list and an associated "index tag". The value of the index tag determines which list element is displayed on the button at run time.

### Operation

Pressing, clicking and releasing a button triggers the functions that you set for these events.

If a button is pressed by mistake and the function is not supposed to be triggered, the operator can keep the button pressed and move the mouse pointer, for example, or his finger (for touch panels) away from the button. This process is not interpreted as a `release` event.

---

### Note

Do not operate several function keys or soft keys or buttons simultaneously. Unintended actions might be initiated otherwise.

---

### Invisible buttons

Invisible buttons are transparent buttons that are not displayed upon the operating unit. If you place invisible buttons on graphics, you can operate parts of the graphic - for example, a motor or a valve.



You create invisible buttons on the open screen in the "Screen Objects" toolbar by selecting the button shown.



If you use invisible buttons in your project, you should enable the display of invisible buttons while configuring by means of the illustrated button in the toolbar.



## 5.2.11 State buttons

### Purpose

The state button is a control and display element with two states: "pressed" and "released". State buttons can signal the status of a unit that is not visible from the operating unit (for example, a motor). This means that you can simultaneously change the status of the unit on your operating unit.



State buttons are created on the open screen in the "Screen Objects" toolbar by selecting the illustrated button or by choosing "Insert" > "State Button" from the menu.

### Response

You can configure the response of the state button. Select one of the following types for this:

- **Switch:** The switch has two stable states. When operate, it changes from one state to the other and retains that state until it is operated again.
- **Pushbutton:** The pushbutton has a stable idle condition. When operated, it goes to the "pressed" state and retains this state for the whole time it remains pressed. It then reverts automatically to the "released" idle condition.

### "Switch" type

- **Tag connection:** For the "switch" type state button, you can configure a tag which represents the current state of the switch and can also change it. Functions that you assign to the switch are triggered only by operating the switch and not by any change in the value of the configured tag.

If you wish, you can define a value for the configured tag that represents the "pressed" state (not for BOOL type tags). The table shows which value the tag assumes when the switch is pressed:

Status	Value (BOOL tag type)	Value (integer tag)
Pressed	TRUE (1)	Configured value $n$
Released	FALSE (0)	For each of $n$ different values

When linked to an integer tag, the switch shows the "pressed" state only if the tag accepts the configured value  $n$ . With any other value, the "released" state is displayed.

- Text/graphic display: For the two states "pressed" and "released", you can configure different pieces of text or graphics which visualize the current state of the switch.  
The figure shows the switch for different ways of configuration.



*Example referring to a switch*

### "Pushbutton" type

The "pushbutton" type of state button can be configured in two different ways:

- Pushbutton: This is the standard way. It behaves very similarly to a button. You can configure text or a graphic for pushbutton labeling, as you please. The labeling applies to both states - in other words, "pressed" and "released". An optical checkback signal of the current state is possible only with a configured 3-D display using the different border colors (refer to figure).



- Pushbutton with checkback signal: With this form you can configure different labeling for the two states "pressed" and "released". The figure shows an example of an optical checkback signal of the two states by using different graphics.



## 5.2.12 Using buttons as direct keys

### Purpose

You can also configure buttons as direct keys on touch-operated devices in order to allow high-speed key operations without delays attributable to communication. Shortcuts are, for example, a condition for typewriter mode.

### Constraint

Direct keys cannot be configured on Panel PCs and OPs; permanent keys are assigned to OPs.

### PROFIBUS direct keys

With PROFIBUS direct keys, you set bits on the operating unit directly in the I/O area of the SIMATIC S7. The bit area is set in SIMATIC STEP 7. The bit is set when the direct key is touched and reset when the key is released or when you quit the screen.

#### Requirements

- At the time your project is generated, ProTool is installed and integrated.
- The operating unit is connected to a SIMATIC S7 over the PROFIBUS-DP during operation.
- You have defined the bit area for direct keys in SIMATIC STEP 7 (you will find configuration tips in the "Communication Manual").

Panel	Inputs (bits)	Outputs (bits)
TP 170B	32 (0...31)	-
TP 270 6"	32 (0...31)	-
TP 270 10" MP 270 TOUCH	40 (0...39)	-
MP 370 TOUCH, MP 370 15" TOUCH	40 (0...39)	-

## 5.2.13 Special features with direct keys

### Functionality of DP direct keys

The functionality of the DP direct keys has been modeled on that of the hard-wired hardware keys.

Particular attention was paid to the fast response time.

#### Windows CE based key-operated devices

With Windows CE based key-operated devices you have to bear in mind that key operation is evaluated irrespective of the screen that happens to be shown, such as when the screen saver is active.

#### For Windows-based touch-operated devices

With touch-operated devices you have to remember further that the direct key is triggered only by touch control and not by clicking the mouse.

Password protection is not supported for buttons having the "Direct Key" function. The DP direct key function is triggered irrespective of the password level configured on the button when the button is operated. Do not use password protection for buttons having the "Direct Key" function.

### Using the "Direct Key" function

When you configure a button with the "Direct Key" function, a fixed area of the touch display is reserved for direct key operation.

Graphic elements that are located above the button having the "Direct Key" function in this area at run time visually conceal the button but operation of the touch display at this point will nevertheless result in the "Direct Key" function being triggered.

---

#### Warning

Unintentional actions on the system can be initiated by operating the button and might possibly result in persons being endangered or in material damage.

---

The following points have to be taken into account to avoid this problem:

- When configuring screens, graphic elements must not conceal the button containing the "Direct Key" function.
- Dynamic updating of the position or the display (enable) of a graphic object as a function of process values must not result in the button having the "Direct Key" function being concealed at run time.

It is essential for these requirements to be taken into account by the configuring engineer. Existing configurations must be modified without delay if necessary.

The problem described above does not occur in the following cases if the button containing the "Direct Key" function is concealed:

- Operate message window
- Cancel screen saver

- Enter a value using the on-screen keyboard
- Run an external application such as Internet Explorer
- Activate the wipe screen
- ProTool recipe dialog boxes

---

**Note**

The "Direct Key" function can be used only once per button.

---

**Behavior of DP direct keys in Offline mode**

If the "Mode" function is triggered with the "Offline" parameter on the operating unit, the connection to the configured PLC or PLCs is cleared.

---

**Note**

The configuring engineer has to give due consideration in as far as STEP 7 and ProTool are concerned:

The DP direct keys remain active in this instance. If you operate a button having the "Direct Key" function in "Offline" mode or operate a corresponding button on a key-operated device, the associated bit is set on the PLC.

---

### 5.2.14 Switches

#### Purpose

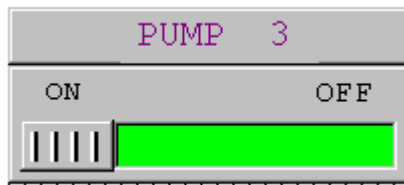
A switch is used to input and output a binary status: it can only be on or off. The "OFF" status corresponds to the value 0 (or FALSE) of the tag linked to the switch. All tag values which are not the same as zero, are interpreted as "ON".



Create a switch by selecting the symbol shown or by choosing "Insert" > "Switch" from the menu.

#### Display:

You can configure the switch in both the horizontal and vertical directions; the "ON" position of the switch toggle is freely selectable. You can label the switch and also visually highlight the current switch status with a background color.



*Switch position marked by a background color*

#### Tag linking

The table shows which value the tag assumes when the switch is pressed:

Status	Value (BOOL tag type)	Value (integer tag)
switched off	FALSE (0)	0
switched on	TRUE (1)	1

When linked to an integer tag, the switch shows the status "switched off" only if the tag has the value 0. With any other value, the switch is "switched on".

### Mouse and keyboard operation

Switches can be operated by the touch panel, the mouse and the keyboard.

- The operator can move the switch directly with the mouse or by touch. When you click in the slide opening, the switch is immediately set to the opposite position and the tag to the corresponding value.
- From the keyboard, the operator can select the switch with TAB (the border will then be shown as dashed) and then toggle it with SPACEBAR / ENTER.

### Triggered events

If the status of the switch is changed by an operator input, this triggers the following events:

- `Switch on`: as soon as the switch changes from "OFF" to "ON".
- `Switch off`: as soon as the switch changes from "ON" to "OFF".
- `Status change`: each time the switch status changes.

## 5.2.15 Slider controls

### Purpose

Use a slider control to enter and display numerical values in analog form. Values are entered by moving the operable slider, for example, with the mouse or the arrow keys, to the desired position. When it is used as a display element, the value to be displayed is derived from the position of the moving slider.



Create a slider control by selecting the symbol illustrated.

### Settings

The slider control is linked to a tag. When you move the operator-controllable slider, the corresponding value is written to the tag. Conversely, when the tag value is changed, so is the position of the moving slider, unless this is just being changed by the operator.

Apart from the tag containing the value to be displayed/entered, you can define other tags in order, for example, to modify dynamically the values for the upper and lower stop at run time.

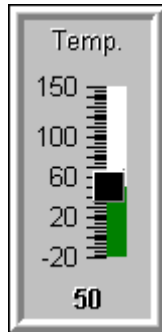
In addition, you can configure, among other things, the following attributes:

- Labeling
- Axis labeling
- Scaling
- Current value (numerical)
- Bar chart



**Display:**

The example in the figure shows a slider for setting and displaying temperature values. The current value (here: 50) is displayed in plain language and also as a bar graph.



*Example referring to a slider*

---

**Note**

To display numeric values in analog form, you can also use the "Analog Display" screen object.

---

---

**Note**

Because scaling is automatic, it could happen that two intermediate scale marks will appear right next to each other (looking like a wide intermediate mark). To undo this effect, make a minor adjustment to the length of the slider control.

You can also completely inhibit the scaling display.

---

**Deviations between slider and tag**

In the cases that follow, the value displayed on the slider might deviate from the actual value of the associated tag:

- The range configured for the slider (minimum and maximum values) do not correspond to the configured limit values for the slider tag.
- An invalid password is entered for a password-protected slider.

## 5.2.16 Analog display

### Purpose

An analog display presents numerical values as a pointer instrument. You can set up as many as three areas on the instrument dial, each with different color backgrounds. This is how to visualize different operating modes, for example, normal operation, warning area and danger area.

---

#### Note

On operating units that run Windows CE, the configured, differently colored scale ranges are not displayed at run time.

You can also configure the display to include a non-return pointer. At run time, the non-return pointer indicates the maximum reached so far by the value to be displayed.



You configure an analog display on the open screen in the "Screen Objects" toolbar by selecting the button shown.

### Settings

The analog display is linked to a tag. When the tag value is changed, the pointer travel of the analog display is also changed. Apart from this tag, which contains the value to be displayed, you can define other tags in order, for example, to modify dynamically the values for upper and lower pointer travel at run time.

In addition, you can configure, among other things, the following attributes:

- Text for the measured quantity (such as temperature) and the unit (such as °C)
- Angle of rotation of the pointer at minimum and maximum values.
- Differently colored scale ranges for Normal, Warning and Danger (are not displayed on Windows CE devices at run time)

---

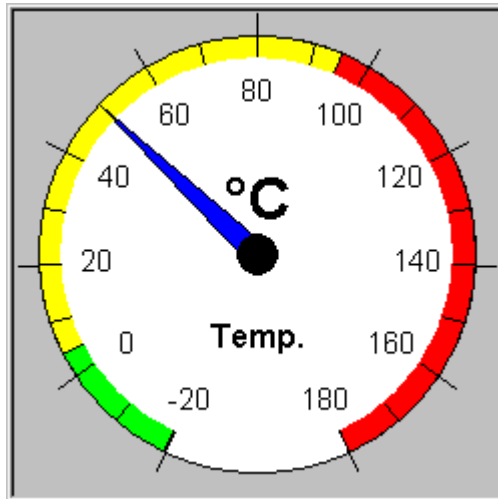
#### Note

The analog display is purely a display element. You cannot use it to enter analog values on the operating unit. Use the "slider control" screen object for this purpose.

---

**Display:**

The example in the figure shows an analog display to present temperatures.



*A typical analog display*

**Background screen and border screen**

You can replace the preset background and border displays with separate screens.

- Background screen: A user-specific background screen covers the dial and the three areas marked in color on the scale. The pointers, pointer axis, scale, scale labeling and text strings for the measured quantity and unit to be displayed, remain visible.
- Border screen: A user-specific border screen replaces the color default background outside the dial.

## 5.2.17 Date/time display

### Purpose

Use the "date/time display" to display calendar dates and times shown on the operating unit and make them modifiable. Date and time displays are language-dependent. It is based on the format specified by your Windows system.

You can also use the "date/time display" for the input and output of tags in date/time format in order to enter, for example, time specifications for the axis of a trend/archive display.

For the "date/time display" you can set a fixed start value by configuring a tag for the "date/time display".



You configure a date/time display on the open screen in the "Screen Objects" toolbar by choosing the button shown.

### Settings

You can perform the following settings, among other things:

- Whether you want to display just the time, just the date or both.
- Whether the displayed data can be modified on the operating unit at run time.
- Whether the date should be shown in short or long form.  
Short form: for example, 24.07.98, 17:09:43  
Long form: for example, Friday, July 24, 1998, 17:09:43

---

### Note

Use four digits for the year in the "Date/time" object. If necessary, change the Regional Options settings in the operating system on your operating unit. Use different separators for date and time in the Regional Settings on the Control Panel. If identical or wrong separators are used, changes of date or time are not applied at run time, or wrong values are set. To display the time, you can use the "digital/analog clock" screen object.

---

### Displaying date and time on the TP 170A

Since the TP 170A does not have a battery-backed system clock, the system time of the PLC has to be displayed in its place. To do this, configure a date/time tag for an input/output field.

## 5.2.18 Digital/analog clock

### Purpose

A digital/analog clock gives you the option to display the system time in numbers (digital) or as a clock with hands (analog).

The digital display also includes the current date. The display format is language dependent. It is based on the format specified by the Windows system of your destination device.



You configure an analog/digital clock on the open screen in the "Screen Objects" toolbar by selecting the button shown.

---

### Note

The digital/analog clock is purely a display element. You cannot set the time or the date on the operating unit. Use the "date/time display" for this purpose.

---

### Display:

The example in the figure shows an analog clock without a face. The clock background has been replaced by a separate screen.



*A typical analog clock*

## 5.2.19 Configuring the date and time for Panels

### xP170B devices or later

These devices can directly access the system time of the HMI device by means of the "Date/Time" screen object. The example "How to configure the date and time for the xP170B or later" will show you how to configure the "Date/Time" screen object. To synchronize the time with the PLC, use the example "How to synchronize the date and time with the PLC".

### TP 170A

Since it is not possible to access the system time of the TP 170A using an input/output field or an output field, the PLC system time has to be displayed in an output field. The examples show you how to configure the date and time on the TP 170A:

- How to display the date and time on the TP 170A
- How to set the date and time on the TP 170A
- How to synchronize the date and time with the PLC

Configuration of the date and time requires programming experience with programmable logic controllers. You will find information about this topic in the documentation for the PLC.

### 5.2.19.1 How to display the date and time on the TP 170A

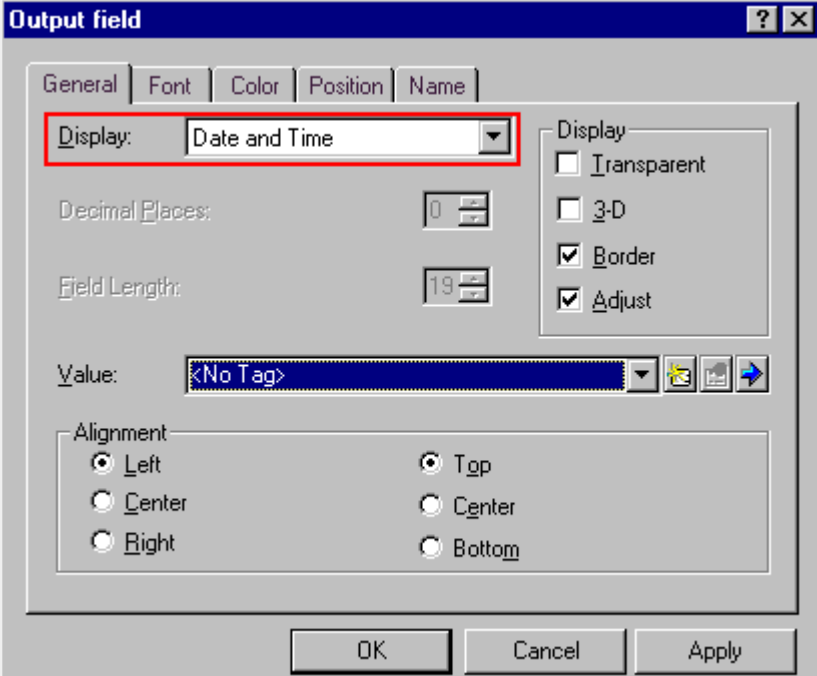
#### Procedure:


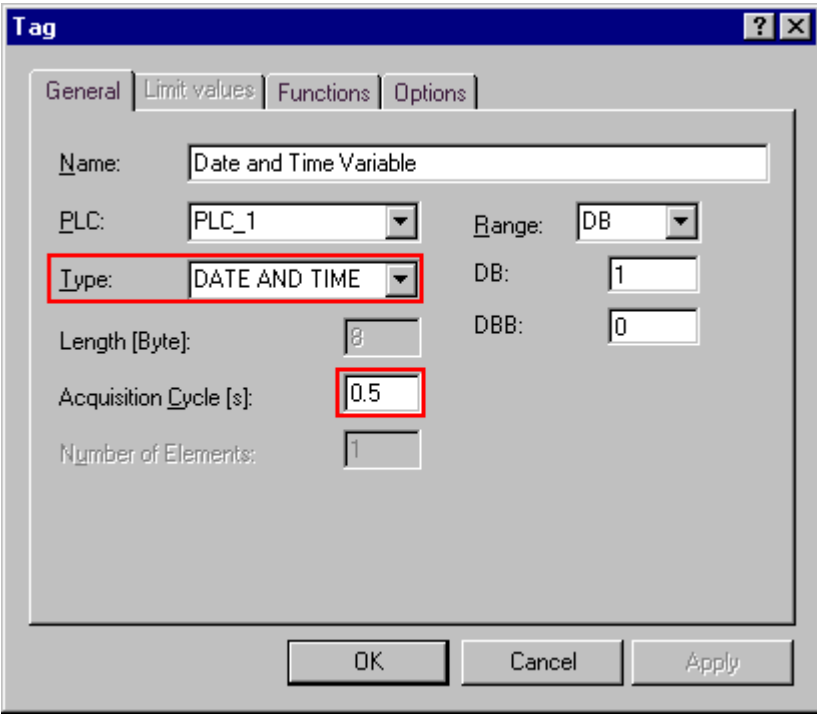
To display the date and time on the TP 170A, configure a date/time tag for an output field. The tag has to be supplied with the system time on the PLC.

#### Note

The example applies to a connection for a S7-300 or S7-400. The differences from a S7-200 connection are pointed out.

The figures show the settings for a connection to a S7-300 or S7-400.

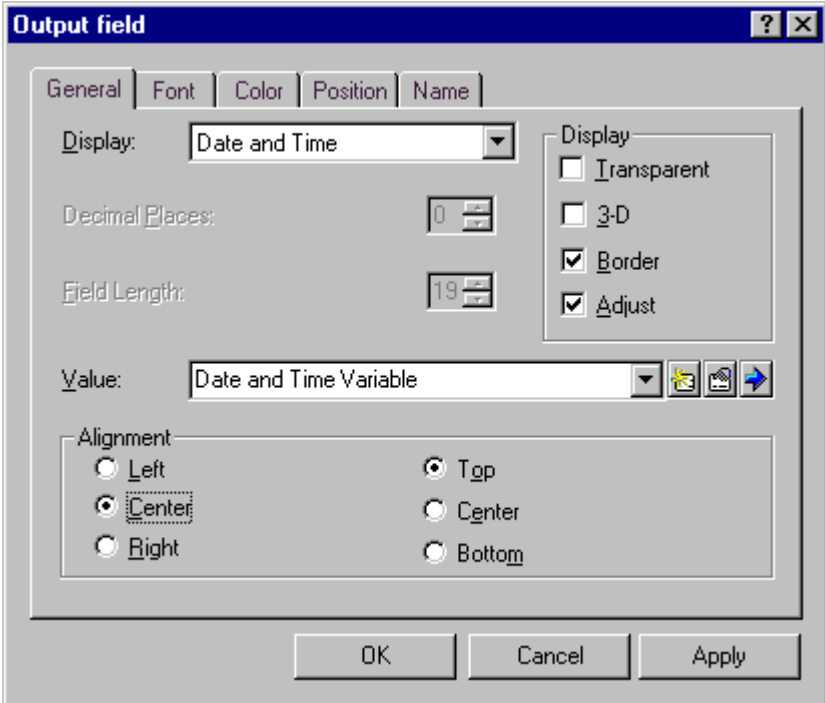
Procedure:	
1	<div data-bbox="483 772 526 814" style="border: 1px solid black; padding: 2px; width: 20px; float: left; margin-right: 5px;">23</div> <p>Insert an output field and perform the following settings:</p> <p>S7-300 or S7-400:            Display: Date and time</p> <p>S7-200: Display: String            Field length: 17</p> 

Procedure:	
2	<p> Create a new tag with the following settings:</p> <p>S7-300 or S7-400:                      Name: Date and Time Tag                      Type: DATE AND TIME                      Acquisition cycle [s]: 0.5                      Area: DB                      DB: 1                      DBB: 0</p> <p>S7-200:                      Name: Date and Time Tag                      Type: STRING                      Length: 17                      Acquisition cycle [s]: 0.5                      Area: V                      VB: 300</p> 



**Procedure:**

3 Apply the settings by clicking OK.



The screenshot shows the 'Output field' dialog box with the following settings:

- Display: Date and Time
- Decimal Places: 0
- Field Length: 19
- Value: Date and Time Variable
- Display options:  Transparent,  3-D,  Border,  Adjust
- Alignment:  Left,  Center,  Right,  Top,  Center,  Bottom

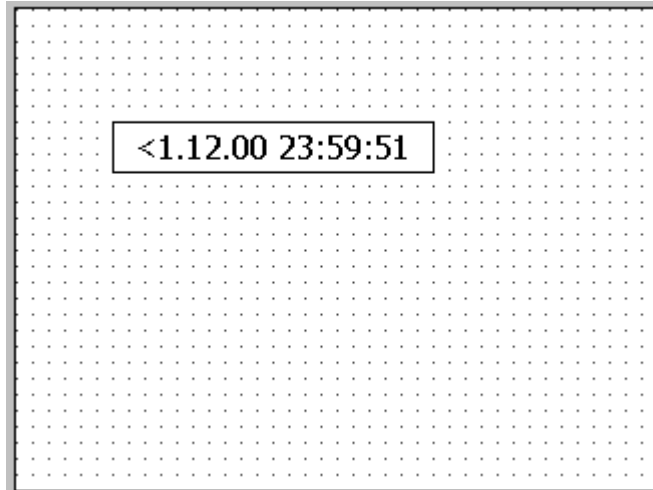
Buttons: OK, Cancel, Apply

<b>Procedure:</b>	
4	<p>S7-300 or S7-400:</p> <p>The text or graphic list of the PLC program in ASCII. It can be imported into the program.</p> <p>126,READ_CLK  126,ReadTime_FC  126,TP_DB  126,WriteTime_FC</p> <p>Define the storage area on the PLC.</p> <p>Data_Block "TP_DB"</p> <p>TITLE = DB with Date and Time tags</p> <p>VERSION: 1.0</p> <p>BEGIN</p> <p>END_DATA_BLOCK</p> <p>Activate FC2 in OB1 (CALL "ReadTime_FC").</p> <p>Activate in PLC program SFC1 "READ_CLK".</p> <p>FUNCTION "ReadTime_FC": VOID</p> <p>TITLE =ReadTime</p> <p>VERSION: 1.0</p> <p>VAR_TEMP</p> <p>END_VAR</p> <p>BEGIN</p> <p>NETWORK</p> <p>TITLE =ReadTime</p> <p>//The PLC system time is read with SFC1</p> <p>END_FUNCTION</p> <p>Note</p> <p>You can activate SFC1 in any program scan cycle. The execution time depends on the CPU being used and can be taken from the CPU documentation. As a rule of the thumb you can use 0.5 ms.</p>

<b>Procedure:</b>	
5	<p>S7-200:</p> <p>In the PLC program, open the "Display Time" subroutine (SBR0).</p> <pre> SUBROUTINE_BLOCK Display_Time:SBR0 BEGIN NETWORK 1 //Time display on TP 170A as 1 STRING tag //VB 300 length 17 //OR //Time display on TP 170A as 2 STRING tags // // TODR VB200 HTA VB202 VB300 2 MOVB 46 VB302 HTA VB201 VB303 2 MOVB 46 VB305 HTA VB200 VB306 2 MOVB 32 VB308 HTA VB203 VB309 2 MOVB 58 VB311 HTA VB204 VB312 2 MOVB 58 VB314 HTA VB205 VB315 2 END_SUBROUTINE_BLOCK </pre> <p>Use the function "Convert hexadecimal number to ASCII string" to convert the time, after it has been read out, to an ASCII string.</p> <p>Note You can activate SBR_0 in any program scan cycle. The recommended setting for the acquisition cycle of the tag "Date and Time Tag" is 0.5 s.</p>

**Result**

The PLC system time is now displayed in the output field.



### 5.2.19.2 How to set the date and time on the TP 170A

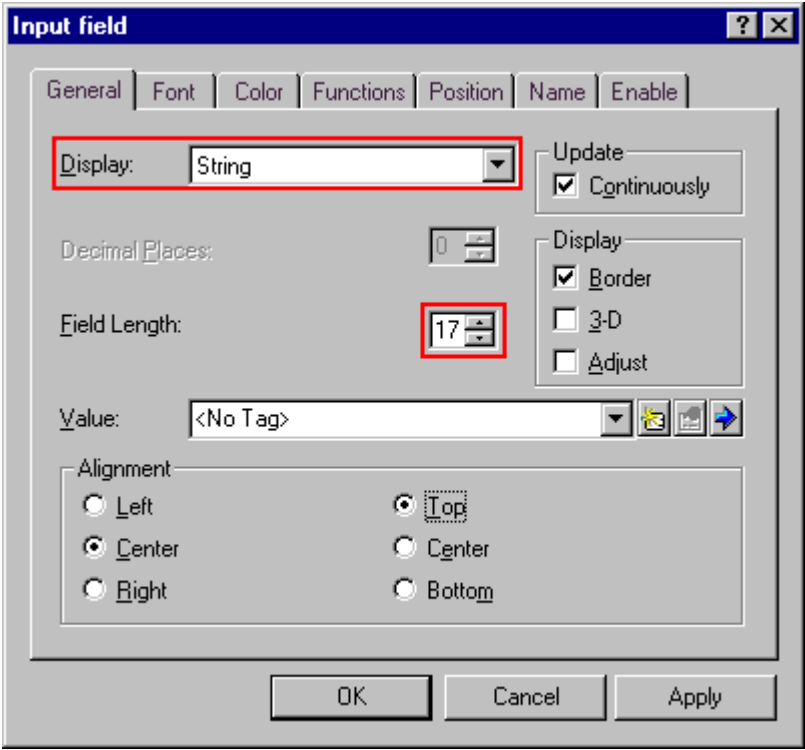
#### Procedure:


If the PLC system time is displayed in the output field on the TP 170A, it should also be possible to set it. To set the date and time, configure an input field and a button for transferring the date and time from the input field to the PLC.

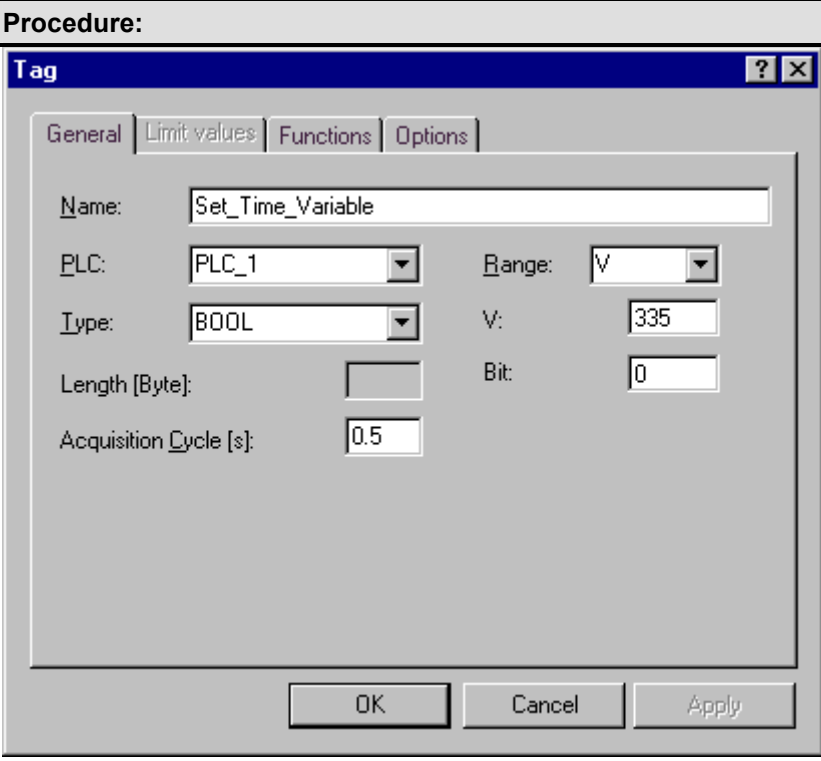
#### Note


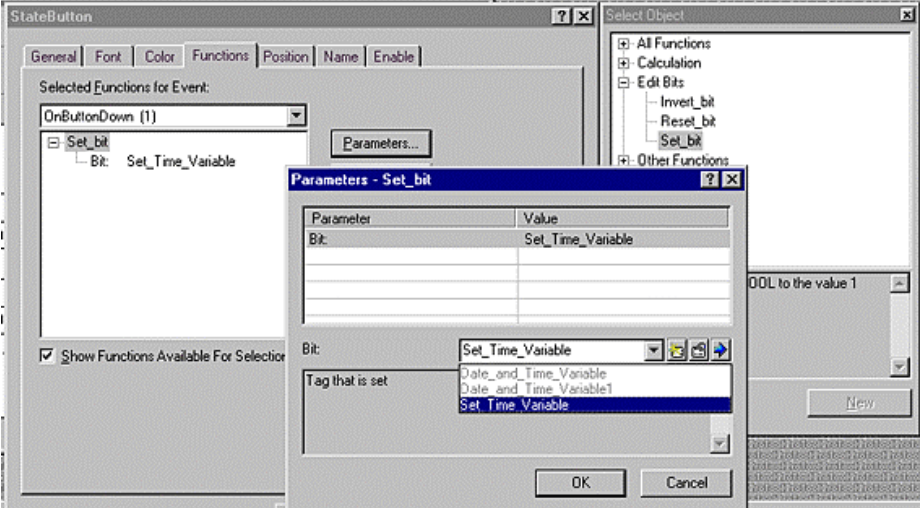
The procedure for configuring an input/output field is the same for a "Date and Time" type output field (S7-300 or S7-400) or a "String" type output field (S7-200). The example applies to a connection for a S7-300 or S7-400. The differences from a S7-200 connection are pointed out.

The figures show the settings for a connection to a S7-200.

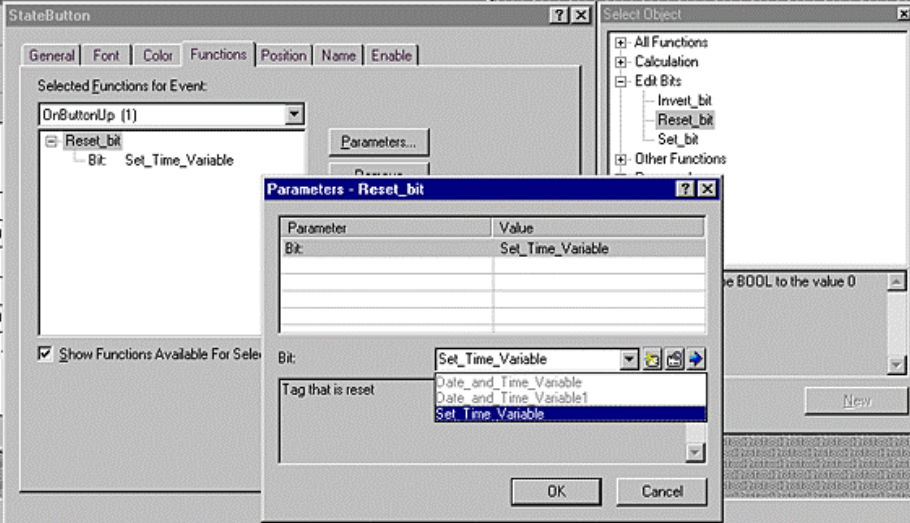
Procedure:	
1	<div data-bbox="479 856 527 907" style="border: 1px solid gray; width: 20px; height: 20px; margin-bottom: 5px;"></div> <p>Insert an input field and perform the following settings:</p> <p>S7-300 or S7-400: Display: Date and time</p> <p>S7-200: Display: String Field length: 17</p> 

Procedure:	
2	<p> Create a new tag with the following settings for the input field:</p> <p>S7-300 or S7-400:            Name: Date_and_Time_Tag1            Type: DATE AND TIME            Acquisition cycle [s]: 0.5            Area: DB            DB: 1            DBB: 8</p> <p>S7-200:            Name: Date_and_Time_Tag1            Type: STRING            Length: 17            Acquisition cycle [s]: 0.5            Area: V            VB: 317</p> <div style="border: 1px solid gray; padding: 10px; margin: 10px 0;"> <p><b>Tag</b> <span style="float: right;">? X</span></p> <p>General   Limit values   Functions   Options</p> <p>Name: <input type="text" value="Date_and_Time_Variable1"/></p> <p>PLC: <input type="text" value="PLC_1"/> Range: <input type="text" value="V"/></p> <p>Type: <span style="border: 2px solid red; padding: 2px;">STRING</span> VB: <input type="text" value="317"/></p> <p>Length [Byte]: <input type="text" value="17"/></p> <p>Acquisition Cycle [s]: <span style="border: 2px solid red; padding: 2px;">0.5</span></p> <p style="text-align: right;"> <input type="button" value="OK"/> <input type="button" value="Cancel"/> <input type="button" value="Apply"/> </p> </div> <p>Apply the settings by clicking OK.</p>

Procedure:	
	
3	<p>Create a tag with the following settings:</p> <p>S7-300 or S7-400:  Name: Set_Time_Tag  Type: BOOL  Acquisition cycle [s]: 0.5  Area: DB  DB: 1  DBX: 16.0</p> <p>S7-200:  Name: Set_Time_Tag  Type: BOOL  Acquisition cycle [s]: 0.5  Area: V  V: 335  Bit: 0</p>

<b>Procedure:</b>	
	<div style="border: 1px solid black; padding: 10px; background-color: #f0f0f0;"> <p>Output:</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 5px auto;">&lt;0000000000000000</div> <p>Input:</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 5px auto;">=0000000000000000</div> <p style="text-align: center;">Format: TT.MM.JJ hh:mm:ss</p> </div> <p>Apply the settings by clicking OK.</p>
4	 <p>Insert a state button with the following settings:</p> <p>Type: Pushbutton Text: Set Time</p>
5	<p>Configure the following functions for the state button:</p> <p>Event "Press (1)": Function <code>Set_Bit</code> with the tag "Set_Time_Tag" as the parameter. When the state button is pressed, the tag is set to "true".</p> <div style="border: 1px solid gray; padding: 5px;">  </div> <p>Event "Release (1)": Function <code>Reset_Bit</code> with the tag "Set_Time_Tag" as the parameter. When the state button is released, the tag is set to "false".</p>



Procedure:	
	
6	Apply the settings by clicking OK in each case.
7	<p>S7-300 or S7-400:</p> <p>The text or graphic list of the PLC program in ASCII. It can be imported into the program.</p> <pre> 126,SET_CLK 126,TP_DB 126,WriteTime_FC </pre> <p>Define the storage area on the PLC.</p> <pre> Data_Block "TP_DB" TITLE = DB with Date and Time tags VERSION: 1.0 BEGIN END_DATA_BLOCK </pre> <p>Activate FC1 in OB1 (CALL "WriteTime_FC").</p> <p>Activate SFC0 "SET_CLK" in PLC program for "true".</p> <pre> FUNCTION "WriteTime_FC": VOID TITLE = SetTime VERSION: 1.0 VAR_TEMP END_VAR BEGIN NETWORK TITLE = SetTime //The PLC system time is set with SFC0 END_FUNCTION </pre>

8	<pre>S7-200 Activate SBR1 "Enter_Time" in the PLC program for "true". SUBROUTINE_BLOCK Enter_Time:SBR1 BEGIN NETWORK 1 //Time entry on TP 170A as 1 STRING tag //VB 317 length 17 //OR //Time entry on TP 170A as 2 STRING tags // // ATH VB317 VB209 2 //Convert day to STRING ATH VB320 VB208 2 //Convert month to STRING ATH VB323 VB207 2 //Convert year to STRING ATH VB326 VB210 2 //Convert hour to STRING ATH VB329 VB211 2 //Convert minute to STRING ATH VB332 VB212 2 //Convert second to STRING TODW VB207 //Read Date/Time END_SUBROUTINE_BLOCK</pre>
---	--

---

**Note**

Alternatively, you can update the system time immediately after your input: Configure the function `Set_Bit` for the "Value Change" event of the "Date and Time Tag1" tag. If the "Value Change" event occurs, the "SetTimeVariable" bit is set. After SFC0 (S7-300 or S7-400) or SBR\_1 (S7-200) has been initiated, the "SetTimeVariable" must be reset, since system functionality will otherwise be impaired.

---

**Result**

Click "Set Time" to apply the date and time from the input field as the new CPU system time.

The screenshot shows a configuration window with a dotted background. It is divided into two sections by dashed lines. The top section is labeled 'Output:' and contains a text box with the hexadecimal value '<0000000000000000'. The bottom section is labeled 'Input:' and contains a text box with the hexadecimal value '=0000000000000000'. Below the input field, the text 'Format: TT.MM.JJ hh:mm:ss' is displayed. At the bottom right of the window is a button labeled 'Set Time'.

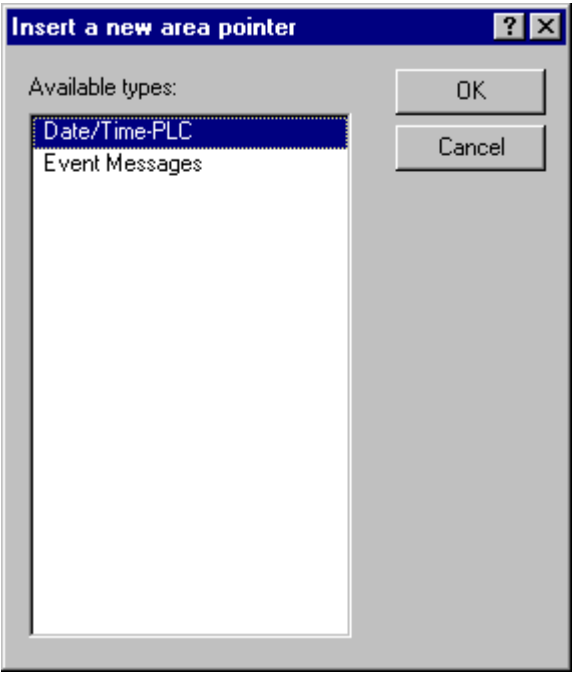
### 5.2.19.3 How to synchronize the date and time with the PLC

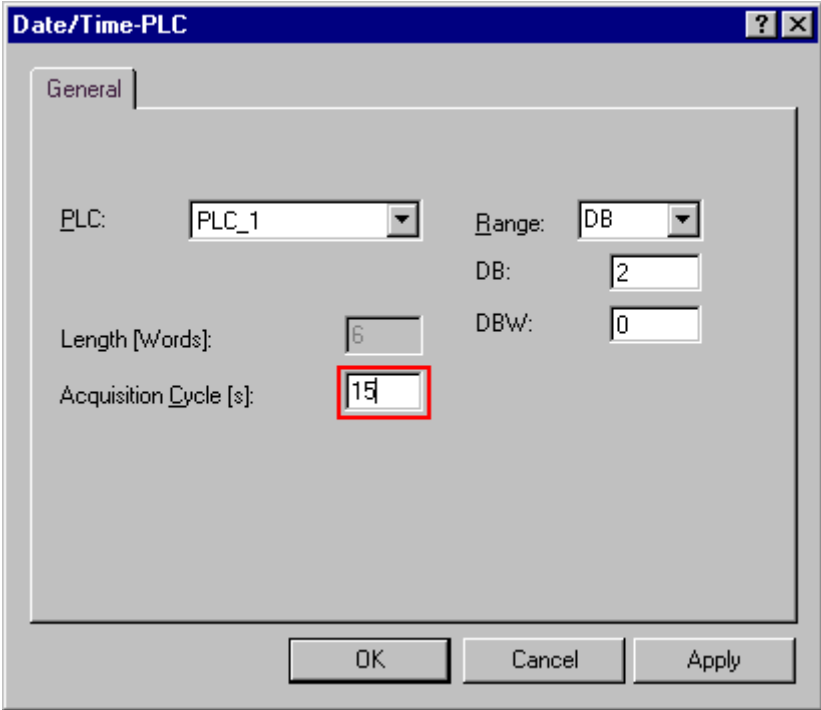
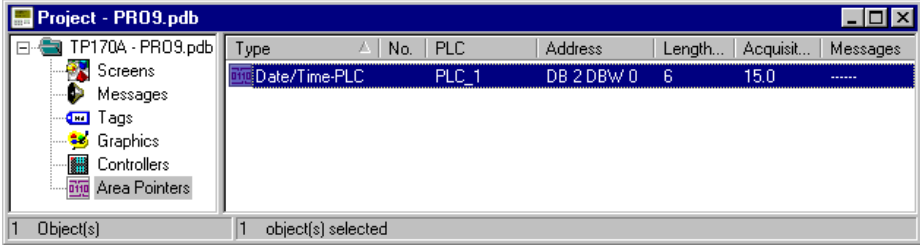
**Procedure:**

With the TP 170A, synchronization of the PLC system time is only necessary if you have configured "Single message view". "Single message view" is the only screen element of the TP 170A that accesses the system time of the device. This constraint applies solely to the TP 170A, not to xP170B.

**Note**

The example applies to a connection for a S7-300 or S7-400. The differences from a S7-200 connection are pointed out.  
The figures show the settings for a connection to a S7-300 or S7-400.

<b>Procedure:</b>	
1	<p>Double-click "Area pointer" in the project window. The "Paste New Area Pointer" dialog box appears.</p> 
2	<p>Select the area pointer PLC Date/Time and click OK to parameterize the area pointer. The "PLC Date/Time" dialog box appears.</p>

Procedure:	
3	<p>Perform the following parameter settings in the "PLC Date/Time" dialog box:</p> <p>S7-300 or S7-400:                      Acquisition cycle [s]: 15                      Area: DB                      DB: 2                      DBW: 0</p> <p>S7-200:                      Acquisition cycle [s]: 15                      Area: V                      VB: 400</p>  <p>Note:                      The area pointer requires 6 DBW. The first four DBWs are reserved for writing the PLC system time in the SIMATIC STEP 7 Date and Time format. The remaining two DBWs are reserved. The recommended acquisition cycle is 15 s.</p>
4	<p>Apply the settings by clicking OK in each case.</p> 

<b>Procedure:</b>	
5	<p>S7-300 or S7-400:</p> <p>The text or graphic list of the PLC program in ASCII. It can be imported into the program.</p> <p>126,BZ_DB            126,CYCL_EXC            126,READ_CLK            126,ReadTime_FC            126,ReadTimeBZ            126,SET_CLK            126,TP_DB            126,WriteTime_FC</p> <p>Define the storage area on the PLC.</p> <p>Data_Block "TP_DB"            TITLE = DB with Date and Time tags            VERSION: 1.0            BEGIN            END_DATA_BLOCK            DATA_BLOCK "BZ_DB"            TITLE = DB for area pointer            VERSION: 1.0            BEGIN            END_DATA_BLOCK</p>
6	<p>S7-300 or S7-400:</p> <p>Initiate FC3 in OB1 (CALL "ReadTimeBZ").</p> <p>Enter the CPU system time in the memory area of the area pointer by initiating SFC.</p> <p>FUNCTION "ReadTimeBZ" : VOID            TITLE = enter system time in BZ            VERSION: 1.0            VAR_TEMP            END_VAR            BEGIN            NETWORK            TITLE =            END_FUNCTION</p>

<b>Procedure:</b>	
7	<p>S7-200</p> <p>Enter the CPU system time in the memory area of the area pointer by initiating SBR2 "Set_Time".</p> <pre> SUBROUTINE_BLOCK Set_Time:SBR2 BEGIN NETWORK 1 // //Set TP 170A clock (for messages) //on the TP 170A -&gt; Date/Time area pointer = VB400, length 6 data words TODR VB214 //read Date/Time BMB VB214 VB400 6 //and load area pointer on TP 170A MOVB 0 VB406 //byte n+6 &amp; n+7 -&gt; MOVB MOVB 1 VB407 //Byte n+6 &amp; n+7 -&gt; reserved END_SUBROUTINE_BLOCK </pre>

The system time of the operating unit will be synchronized at run time with the system time of the PLC without the user having to trigger it.

### 5.2.19.4 How to configure the date and time for xP170B

**Procedure:**

The "Date/Time" screen object is available for xP170B devices and higher. With it, you can directly access the system time of the HMI device.

<b>Procedure:</b>																					
1	<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <div style="border: 1px solid gray; width: 30px; height: 20px; display: flex; align-items: center; justify-content: center; margin-bottom: 5px;">2:37</div> <p>Insert the "Date/Time" screen object and perform the following settings:</p> <p>Time: Yes  Date: Yes  Editing allowed: Yes  System time: Yes</p> </div> <div style="border: 1px solid gray; padding: 10px;"> <div style="border: 1px solid gray; padding: 5px;"> <div style="background-color: #4a7ebb; color: white; padding: 2px; display: flex; justify-content: space-between; align-items: center;"> <span>Date/Time</span> <span style="font-size: 0.8em;">? X</span> </div> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 5px;"> <thead> <tr> <th style="width: 25%;">Position</th> <th style="width: 25%;">Name</th> <th style="width: 25%;">Help text</th> <th style="width: 25%;">Enable</th> </tr> </thead> <tbody> <tr> <td>General</td> <td>Font</td> <td>Color</td> <td>Attributes</td> </tr> <tr> <td colspan="2">                     Display  <input checked="" type="checkbox"/> Time  <input checked="" type="checkbox"/> Date  <input type="checkbox"/> Long Date Format                 </td> <td colspan="2">                     Field Type  <input checked="" type="checkbox"/> Allow Edit                 </td> </tr> <tr> <td colspan="2">                     Alignment  <input checked="" type="radio"/> Left      <input checked="" type="radio"/> Top  <input type="radio"/> Center      <input type="radio"/> Center  <input type="radio"/> Right      <input type="radio"/> Bottom                 </td> <td colspan="2">                     Display  <input type="checkbox"/> Transparent  <input checked="" type="checkbox"/> Border  <input type="checkbox"/> 3-D  <input checked="" type="checkbox"/> Adjust                 </td> </tr> <tr> <td colspan="4">                     Data Source  <input checked="" type="radio"/> System Time  <input type="radio"/> Tag      &lt;No Tag&gt;                 </td> </tr> </tbody> </table> <div style="text-align: right; margin-top: 10px;"> <input type="button" value="OK"/> <input type="button" value="Cancel"/> <input type="button" value="Apply"/> </div> </div> </div>	Position	Name	Help text	Enable	General	Font	Color	Attributes	Display <input checked="" type="checkbox"/> Time <input checked="" type="checkbox"/> Date <input type="checkbox"/> Long Date Format		Field Type <input checked="" type="checkbox"/> Allow Edit		Alignment <input checked="" type="radio"/> Left <input checked="" type="radio"/> Top <input type="radio"/> Center <input type="radio"/> Center <input type="radio"/> Right <input type="radio"/> Bottom		Display <input type="checkbox"/> Transparent <input checked="" type="checkbox"/> Border <input type="checkbox"/> 3-D <input checked="" type="checkbox"/> Adjust		Data Source <input checked="" type="radio"/> System Time <input type="radio"/> Tag      <No Tag>			
Position	Name	Help text	Enable																		
General	Font	Color	Attributes																		
Display <input checked="" type="checkbox"/> Time <input checked="" type="checkbox"/> Date <input type="checkbox"/> Long Date Format		Field Type <input checked="" type="checkbox"/> Allow Edit																			
Alignment <input checked="" type="radio"/> Left <input checked="" type="radio"/> Top <input type="radio"/> Center <input type="radio"/> Center <input type="radio"/> Right <input type="radio"/> Bottom		Display <input type="checkbox"/> Transparent <input checked="" type="checkbox"/> Border <input type="checkbox"/> 3-D <input checked="" type="checkbox"/> Adjust																			
Data Source <input checked="" type="radio"/> System Time <input type="radio"/> Tag      <No Tag>																					
2	Apply the settings by clicking OK.																				



**Result**

The system time of the operating unit is then displayed in the Date/Time field. The "Editing allowed" setting supports direct write access to the system time of the operating unit. You can set the time by entering a new time in the Date/Time field.

The system time of the XP170B can be synchronized with the system time of the PLC using the "PLC Date/Time" area pointer. PLC job 41 can be used to synchronize the system time of the PLC with the system time of the XP170B.

## 5.3 Using tags

### Purpose

Since tags are the most important means of communication between the operating unit and the PLC and for exchanging data, a detailed explanation is presented here of what tags are and what types of tag are used in ProTool.

### Definition

A tag has a symbolic name and a defined data type. The value of the tag changes while the PLC program is being executed.

Tags with a PLC link are referred to as global tags. Tags without a PLC link are known as local tags.

- **Global tags**  
A tag with a PLC link occupies a defined memory address on the PLC, to which read and write access is possible from both the operating unit and the PLC.
- **Local tags**  
Local tags are not connected to the PLC. They are available only on the operating unit. You create local tags, for example, so that the operator can enter limit values on the operating unit.

### Tag types

ProTool recognizes the following tag types (but these are not available on every PLC):

data type	Bit System	Range of Values
BYTE (OP 7/17 only)	8 bit	0 through 255
INT	16 bit	- 32768 through 32767
UINT	16 bit	0 through 65535
LONG	32 bit	- 2147483648 through 2147483647
ULONG	32 bit	0 through 4294967295
FLOAT	32 bit	Upper limit: $\pm 3.402823 \text{ e}+38$ Lower limit: $\pm 1.175495 \text{ e}-38$
DOUBLE (OP7/17 only)	64 bit	Approx. 12 valid digits
BOOL		true (1), false (0)
STRING (1)		1 to 128 bytes
DATE	64 bit	Date/time value

data type	Bit System	Range of Values
ARRAY tags		This tag type combines a given number of tags of the same type to form a total, which can be handled as a whole.

(1): The StringChar data type is featured in addition to the STRING data type. The StringChar data type is made available for the migration of graphic display or text-based displays to Windows-based systems. Use the STRING data type in new projects.

---

#### Note

If you configure tags for input and output fields, you should always configure separate tags for input and output. If you do not, there might be contentions when tags are modified simultaneously by the operator and the PLC.

---

#### Online tags for S5

data type	Bit System
KF	16 bit, fixed-point
KH	16 bit, hexadecimal
KM	16 bit, binary
KY	8 bit, 8 bit (2 x 8 bit, comma separated)
KC	– (ASCII character)
KT	– (S5 timer)
KZ	– (S5 counter)
BCD4:	16 bit, BCD
DF	32 bit, fixed-point
DH	32 bit, hexadecimal
KG	Floating point

#### Online tags for S7

data type	Bit System
CHAR	8 bit
BYTE	8 bit
INT	16 bit, fixed-point
WORD	16 bit
DINT	32 bit, fixed-point
DWORD	16 bit
REAL	32 bits floating point
BOOL	1 bit
STRING	ASCII character

data type	Bit System
TIMER	Timer (S5#Time)
COUNTER	Counter

---

**Note**

The details "fixed-point", "hexadecimal", etc., refer to the representation of the CPU. They have no bearing on the display options on the OP.

---

**Tooltip**

If the mouse moves above a tag name in the project window, a tooltip is displayed. It shows you the following information about the tag:

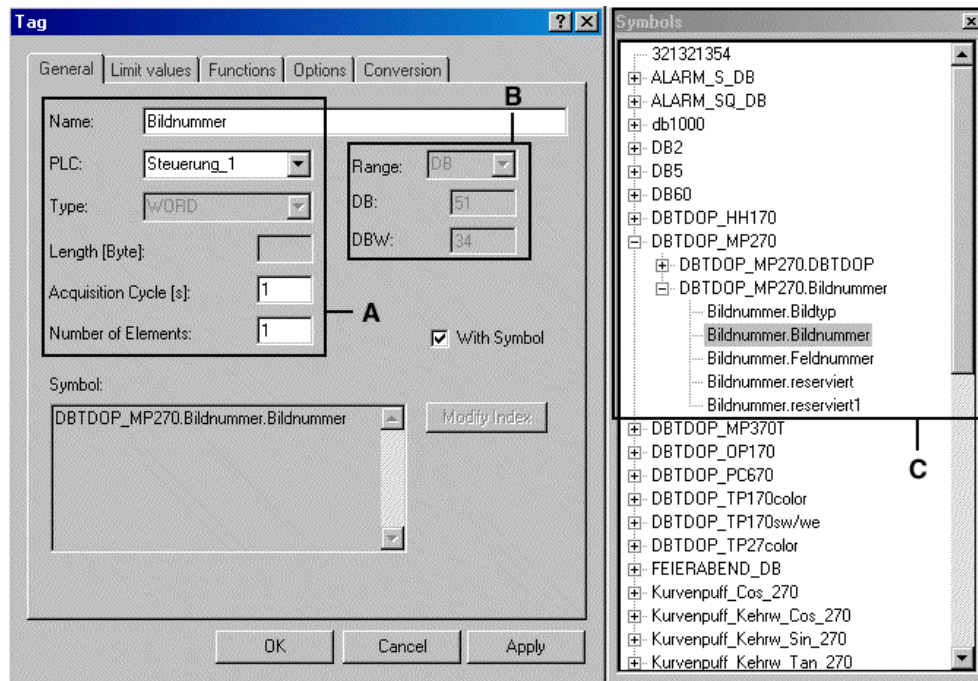
- Type
- PLC
- Address
- Symbolic link (if integrated)
- Full tag name

### 5.3.1 Properties of tags

#### Tag definition (Section A)

To define a tag, set the properties of the tag in the "Tag" dialog box.

The figure shows an example of the "Tag" dialog box for SIMATIC S7 integrated in SIMATIC STEP 7.



"Tag" dialog box for the SIMATIC S7

The available data types and data formats depend on the PLC you selected. You select it at "Type" or "Format" in the "Tag" dialog box.

#### Updating tags (Section A)

The acquisition cycle determines the time interval in which the value of a tag is updated on the operating unit.

- If 0 is entered for the acquisition cycle, the tag is updated only when a screen is opened and then not updated any more until the screen is opened again.
- If a value > 0 is entered for the acquisition cycle of a tag, the tag is updated periodically in the specified time and also when the screen is opened.

The acquisition cycle is a multiple of the "standard clock pulse", which you configure for every PLC in the "PLC" dialog box. You globally modify the acquisition cycles for all the tags of a project by setting the standard clock pulse.

### **Array tags**

An array tag represents a number of tags of the same type with consecutive memory addresses. If you wish to define an array tag, enter the number of elements in the array in the "Tag" dialog box at "Number of elements".

The system default setting is 1, in other words, the tag is not defined as an array.

Application example:

You can use array tags for history trends, for instance.

### **Address (Section B)**

The address determines the memory location of a global tag on the PLC. The address therefore depends on the PLC you are using.

The display of the address depends on the PLC you selected. This section of the "Tag" dialog box adjusts dynamically to the programmable address areas.

### **ProTool integrated in SIMATIC STEP 7 (Section C)**

If you have installed ProTool integrated in SIMATIC STEP 7, you can access the "STEP 7" symbol table directly in the "Tag" dialog box. The symbol table is displayed in the "Symbols" dialog box when you select the "With symbol" check box.

### **Accessing tags by means of OPC, OLE or simulator**

If tags are accessed by means of OPC, Visual Basic (OLE Automation) or the simulator, which can also be used in functions as parameters or in scripts, it might occur in isolated cases that changes of screen no longer work or some screen elements can no longer be displayed.

In such a case, set the tags concerned to "Read continuously", or configure an additional tag with the same address.

## 5.3.2 Array tags

### Definition

Array tags consist of a definable number of elements, in which tag values of the same type can be stored. The number of elements specifies the memory addresses that are sequentially assigned in the memory.

### Usage in ProTool

You can explicitly address the contents of an element using an Index tag. The use of array tags in ProTool makes sense if you read in values sequentially or if you have to configure several tags of the same type:

- History trends  
To specifically access the values of the history trend, configure an array tag. You can have all acquired values of the history trend read out by gradual incrementing of the index tag.
- Recipes  
If you have a large number of tags of the same type, you can configure an array tag with the corresponding number of elements. In this way you can cut your configuration time because you only have to configure one tag. Furthermore, performance is improved in Runtime when process data are being transferred.

### Special features and constraints

Even if you only have the value of an array element displayed - ProTool treats an array tag as **one** tag and therefore always reads all the contents of the array element. If an array tag is connected to the PLC, all contents will be consequently transferred upon a change as a general rule. You therefore cannot write to the same array tag simultaneously from the operating unit and the PLC.

---

#### Note

To avoid unnecessary system loading, you should use array tags only where values are not subject to frequent change.

---

---

#### Note

If you have configured large arrays (such as more than 50 word elements) that are read cyclically, trend plots and tag archives are influenced unfavorably.

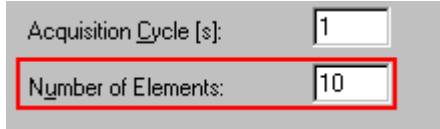
---

The following constraints apply to array tags:

- They can consist only of one-dimensional array elements.
- They cannot be String type tags.

**Configuring array tags**

Configure array tags by entering the number of elements (at least two) in the "Tag" dialog box:

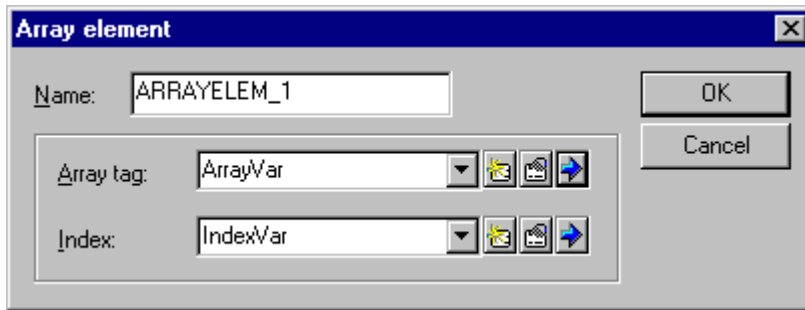


**Array element**

An array element in ProTool consists of an array tag with an associated index. The individual elements can be specifically addressed by means of the index. The index can be either a constant or a tag. An array element is always attached to an object (for example, an input field).

**Configuring an array element**

Configure an array element by opening the Properties dialog box for the desired object. In the dialog box, select "Array element" and then "New". In the "Array Element" dialog box, select the array tag and the associated index and assign a name to the array element:





### 5.3.3 Tag settings

#### Setting a start value

You can set a "start value" for the tag at "Options". When the project is downloaded, the tag is assigned the start value. The start value is displayed only on the operating unit and is not stored on the PLC.

#### Example

If tags are used for scaling trends and bar graphs, the initial value may be the start value of the scaling.

---

#### Note

It is not always possible to specify a start value for STRING type tags.

---

#### Setting limit values

An upper and a lower limit value can be configured at "Limit values" for tags.

If the tag value is outside the defined range, in other words, it is higher or lower than the limit value concerned - this has the following effect on the input fields, for example. If the operator enters a value outside the configured limit values, the input is rejected and the original value is retained.

#### Configuring tags with functions

You can assign functions to tags in input/output fields - for example, the "Select Screen Permanent" function. The function is triggered as soon as the value of the tag changes thus in our example the screen is selected.

### 5.3.4 Updating tags

#### Principle

You can update the tags in your project in different cycles. So as not to place an unnecessary burden on system performance, the update time must be set in an optimum manner.

The following applies to calculation of the update time:

**Update time = acquisition cycle plus transfer time plus processing time.**

You achieve optimum update times by taking the following points into account while you are configuring:

- Create the different data areas as small as possible and as large as necessary.
- Define contiguous data areas where they belong. The actual update time improves if you create one large area instead of several small ones.
- If you select too small an acquisition cycle, it has an adverse degrades overall performance. Set the acquisition cycle in line with the update rate of the process values. Example: The temperature variation of a furnace, for example, is considerably less dynamic than the variation in speed of an electric drive.
- If necessary, dispense with the cyclic download of user data areas (data entry cycle 0) to improve update times. Instead, use PLC jobs to download user data areas spontaneously.
- Place all message or screen tags in a data area.
- For the operating unit to be able reliably to detect changes on the PLC, the changes must be queued during the actual acquisition cycle.

#### Options: Read continuously

On the "General" tab, you can choose at "Acquisition cycle" how often the value of a tag is transferred between the operating unit and the PLC and updated.

If you select the option "Read Continuously",

---

#### Note

Use the "Read Continuously" option only for tags that you want to have updated continuously. Otherwise you will unnecessarily degrade the performance of your system.

---

### Acquisition cycle

The acquisition cycle determines the time interval in which the value of a tag is updated on the operating unit.

- If 0 is entered for the acquisition cycle, the tag is updated only when a screen is opened and then not updated any more until the screen is opened again.
- If a value > 0 is entered for the acquisition cycle of a tag, the tag is updated periodically in the specified time and also when the screen is opened.

The acquisition cycle is a multiple of the "standard clock pulse", which you configure for every PLC in the "PLC" dialog box. You globally modify the acquisition cycles for all the tags of a project by setting the standard clock pulse.

System default setting: Acquisition cycle 1 s

### Standard clock pulse

Set the standard clock pulse in the "PLC" dialog box. The standard clock pulse of the operating unit is the basic factor for the update rate, which you can set by means of the acquisition cycle of the tag. You globally modify the acquisition cycle for all the tags of a project by modifying the standard clock pulse.

System default setting: Standard clock pulse 500 ms

### Example

If you increase the standard clock pulse to 1000 ms, the acquisition cycle for all tags is increased to 2 s.

In order not to overtax communication between the PLC and the operating unit, the times you set for the standard clock pulse should not be too short. This would mean that other processes, such as the updating of trends or the execution of PLC jobs, would take considerably longer.

### 5.3.5 Address multiplexing

#### Requirements

Address multiplexing is possible only when a SIMATIC WinAC, SIMATIC S7-300, SIMATIC S7-400 or SIMOTION is being used as a PLC.


#### Definition

With address multiplexing, the address parameters of a tag are modified as a function of the value of the multiplex tag.

#### Purpose of address multiplexing

With address multiplexing, you can reference a number of memory addresses in the address area of the PLC with a single tag. You can read or write to the addresses without having to define a tag for every single address, that is, it is a very efficient method of processing data.

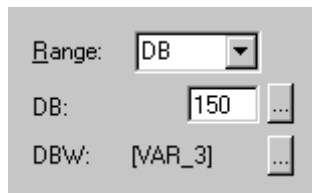
#### Configuring a multiplex tag

 You define the multiplex tag by pressing the button shown in the "Tag" dialog box next to the address.

Pay attention to the following when configuring multiplex tags:

- The multiplex tag must not be multiplexed.
- The "type" of multiplex tag must match the type of address parameter which the multiplex tag is defining.

The following figure shows the result of the address multiplexing for tags:



*Address multiplexing with tags*

---

#### Note

Address multiplexing is not possible with ProTool tags that are used in scripts, because the tag value is not read directly from the PLC after the address is changed.

---

## 5.3.6 Multiplexing tags

### Definition

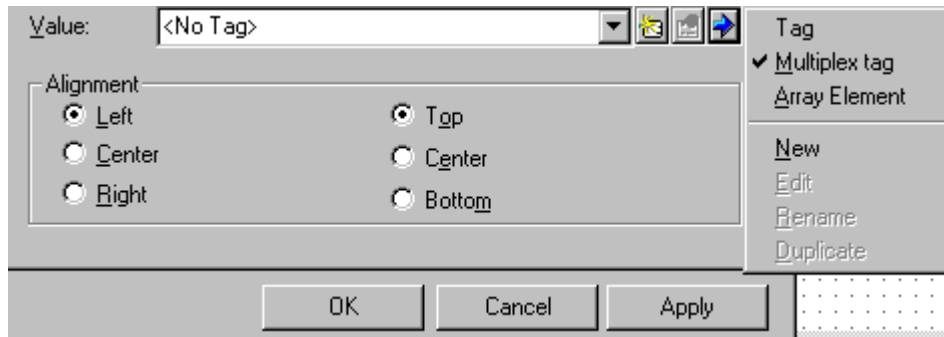
With tag multiplexing, tags are modified as a function of the value of an index tag.

### Purpose of tag multiplexing

With tag multiplexing, you can access a number of tags, which you saved in a multiplex tag, by means of an index tag. To visualize machines of identical types, you only have to configure one screen, thus saving memory on the destination device and also configuration time. You can implement switching between the different views by means of a selection list, for example, for which you configure an index tag.

### Configuring a multiplex tag

You define a multiplex tag by selecting the "Multiplex tag" entry in the Properties dialog box of the screen object you want to configure:



All configured multiplex tags are displayed in the selection list. You can select an existing multiplex tag or create a new one by clicking "New" on the menu.

In the "Multiplex tag" dialog box, create a list of tags and select the associated index tag.

---

### Note

You can also create a multiplex tag by double-clicking "Multiplex tag" in the project window. All configured multiplex tags are displayed in the project window.

---

### 5.3.7 Archiving tags

#### Principle

Windows-based systems offer you the opportunity to archive process data (in other words, store them permanently and analyze them). You set the trigger condition and the range of values for archiving a tag in the "Tag" dialog box on the "Archive" tab.

You will find more information at "Archiving process data" and in the example called "Establishing an archive for tags".

## 5.3.8 Functions to change tags

### Converting tags linearly

In the "Tag" dialog box on the "Scaling" tab, you can configure the value of a tag to be converted linearly between the PLC and the operating unit.

The tag values are converted in accordance with the following equation:  $y = a*x + b$ .

The scaling is applied logically between the PLC value and the tag in other words, the scaled value is already contained in the tag.

If you do not configure scaling, the tag value on the PLC corresponds to the tag value displayed on the operating unit.

### Functions to change tags

Here you are given an overview of the various functions available to you in ProTool for editing tag values.

Basically, the contents of tags can be modified as follows:

- By setting individual tag bits  
Functions for bit operations are set if you want to switch between two states, for example, to turn a switch on or off, or to open or close a valve.
- By modifying the value of the complete tag  
Value changes of the entire tag are used if you have to switch between more than two states (for example, between several languages) or process freely selectable values (for example, to enter a volume).
- By linearly converting the value of a tag  
Linear calculations are used to scale values or to render them relative to an alternative quantity.

### 5.3.9 Functions to set tags bit by bit

#### Usage

A typical requirement for an operating unit interface is to turn a switch on or off, or to open or close a valve. When configuring the interface, this type of process can be reduced to setting binary states: the switch can either be set to "ON" or "OFF", the valve is either open or closed.

The classic approach is to control such tasks using BOOL tags, where the single bit either has the value FALSE (0) or TRUE (1).

With PLCs which do not use BOOL tags (for example, SIMATIC S5) or where only limited memory is available, you can also edit tags bit by bit to control binary states (switch ON/OFF, valve OPEN/CLOSED).

When using functions for editing tags bit by bit, you must take the following points into consideration:

- As, after the change is made, ProTool writes the entire tag to the PLC and does not check whether the value in the PLC has changed in the meantime, the PLC only has read access to the tag.
- Depending on the PLC, the bits of a tag will either be counted from right to left or from left to right. As you usually know to which PLC the tag is linked, ProTool automatically scales your information to the requirements of the PLC.

However, if you use functions to edit tags bit by bit in VBScripts, the PLC to which the tag is connected is not known from the outset. So in VBScripts, the bits are always counted from right to left, starting with 0.

#### Functions

You can use the following functions to set the individual bits of a tag:

Call	Function	Application examples
Set Bit Reset Bit Set/Reset Bit	Changes the value of a tag to TRUE or FALSE	can only be used for BOOL tags
Set Bit in Tag Reset Bit in Tag Set/Reset Bit in Tag	Changes a bit from 0 to 1 or vice versa; the entire tag is always written to the PLC	Usage for PLCs, where there are no tags of the BOOL type, or to save memory in the PLC; used, for example, to operate switches
Set Bit on Pressing Key	The bit remains set until the key is released	Operating the traversing keys

You will find an example of using these functions at the topic called "Using bit operations for operator control".



### 5.3.10 Functions to change the value of a tag

#### Usage

Value changes of the entire tag are used for the configuration if you have to switch between more than two settings (for example, between several languages) or process freely selectable values (for example, to enter a volume).

#### Functions

You can use the following functions to change the entire value of a tag:

Call	Function	Application examples
Set Value	A specific value is assigned to a tag	Set the start value for a process
Decrease Value Increase Value	The value of a tag is reduced or increased by a specific amount	Gradual up and down adjustment

You will find an example of using these functions in the topic called "Configuring stepping switches".

### 5.3.11 Functions for the linear scaling of tags

#### Usage

With many projects, it is simpler for the operator to enter the values of a tag in a different unit to that used by the PLC. In this case, you can scale the entered tag values automatically, before they are transferred to the PLC. Set this automatic scaling as a tag "property" on the "Scaling" tab.

If you want to access the scaled value in your project (for example, to output it), you need a second tag, to access the scaled value in the PLC.

#### Functions

With the following functions, you can perform linear scaling for the value of a tag and store it in a second tag or forward it to the PLC:

Call	Function
"Tag" dialog box, "Scale" tab	For a tag with the PLC connection: "Variable_1", linear scaling is automatic before it is transferred to the PLC. A corresponding reverse transformation is performed on values from the PLC, before the value in "Variable_1" is changed on the operating unit.
Scale Linearly 1 Scale Linearly 2	"Variable_1" without a PLC connection undergoes linear scaling to "Variable_2".

You will find an example of using the "Scaling" tab in the topic called "Automatically converting entered values for the PLC".



### 5.3.12 Examples of configuring tags

Using examples, we will show you here how you configure a multiplex tag and how a tag archive is structured.

#### 5.3.12.1 How to configure a multiplex tag (address multiplexing)

##### Procedure:

To multiplex an address, create the following tag, for example:

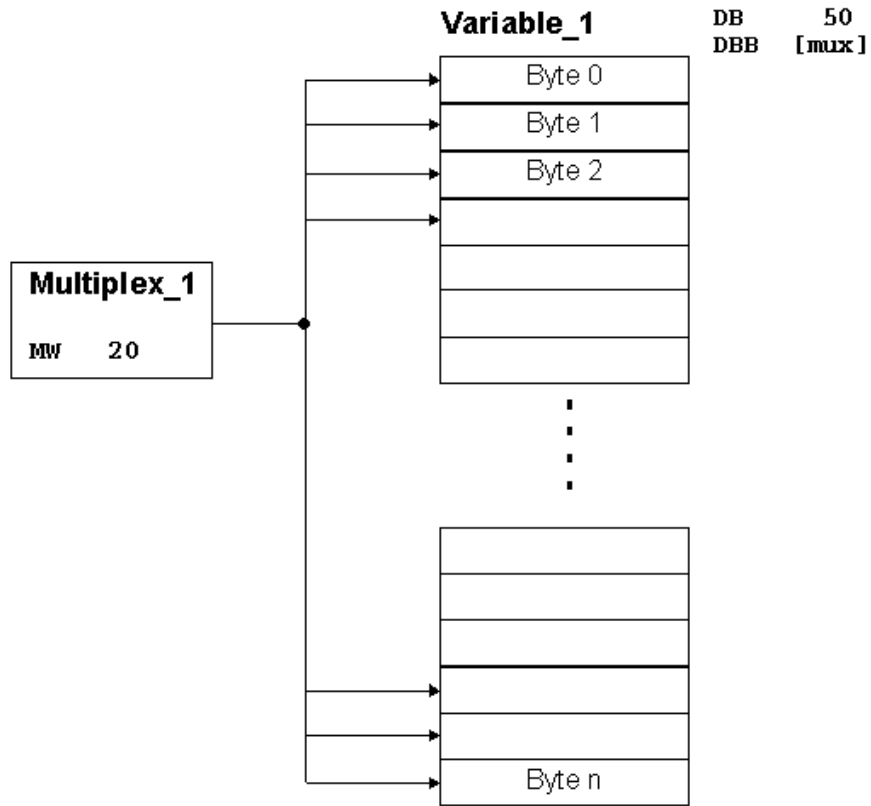
Procedure:	
1	Select "Insert" > "Object" to configure a new tag for any object.
2	In the "Tag" dialog box, enter <code>Variable_1</code> as the name of the new tag.
3	On the "Address tab", select <code>INT</code> at Type and enter the address as: <code>DB 50</code> <code>DBB 0</code>
4	 Click this button to open the "Address Multiplexing" dialog box.
5	Activate address multiplexing.
6	Enter <code>Multiplex_1</code> as the name of the new multiplex tag.
7	 Click this button to define the multiplex tag.
8	On the address tab, select <code>BYTE</code> at Type and enter the address: <code>F W 20</code>
9	The multiplex tag can assume values between 0 and 255, in other words you can reference 256 different addresses with the tag <code>Variable_1</code> . Apply the settings by clicking OK.

##### Note

If you want to make use of the maximum range of values of the multiplex tag, `DB 50` in the STEP 7 program has to be created with at least 128 data words.

**Result**

The name of the multiplex tag is displayed as the address of the tag Variable\_1:  
 DB 50  
 DBB [MULTIPLY\_1]



*Example: configuring a multiplex tag*

### 5.3.12.2 Structure of an archive for tags

#### Structure

Here is an example of a file for tags:

```
"VarName";"TimeString";"VarValue";"Validity";"Time_ms"
"Var_107";"01.04.98 11:02:52";66,00;1;35886460322,81
"Var_108";"01.04.98 11:02:55";60,00;1;35886460358,73
"Var_109";"01.04.98 11:02:57";59,00;1;35886460381,22
```

The following is entered in the various columns:

VarName	Name of the ProTool tag
TimeString	Time stamp as a string, that is, in a readable date format
VarValue	Value of the tag
Validity	Validity: 1 = value is valid 0 = error has occurred
Time_ms	Time stamp as decimal value <sup>*)</sup> Converted: required only for displaying the tag values as a trend.

<sup>\*)</sup> Note on converting values in the `Time_ms` column:

If you wish to process a value in another program, it must be divided by 1 000 000.

Example: 36343476928: 1 000 000 = 36343.476928

A date from 31.12.1899 onward is derived by calculation from the value before the decimal point.

The time is derived from the value following the decimal point: 0.476928. Multiply by 24 to give the hours; multiply any remainder by 60 to give the minutes; again multiply any remainder by 60 to give the seconds; any further remainder multiplied by 1000 gives milliseconds.

Result: 07. 02.1999 11:26:46

### 5.3.13 Details on configuring tags

At "Details" you will find more information on the subject of tags.

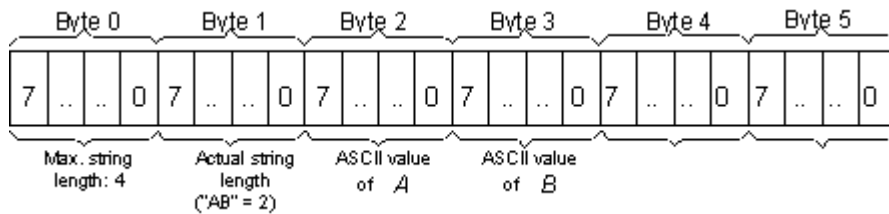
#### 5.3.13.1 Saving STRING tags

##### ProTool integrated in SIMATIC STEP 7

The STRING data type contains a length identifier in the first few bytes in addition to the test information. This data type is compatible with STEP 7 and is used when you are working with ProTool integrated in SIMATIC STEP 7.

If you have integrated ProTool into SIMATIC STEP 7 and have used STRING type tags, these tags are saved and updated by ProTool in a similar manner to SIMATIC STEP 7.

The following example shows the order of the bytes when specifying the STRING[4] data type with the initial value of 'AB'.



*Example: STRING[4] type tag in ProTool*

ProTool/Pro interprets the hexadecimal value 00H in strings as an ID for the end of the string. The following characters in the string are not displayed.

The StringChar data type is featured in addition to the STRING data type. It was included specifically for the migration of graphics-oriented projects. In contrast to the STRING data type, the StringChar data type contains only text information. It is an array of n characters, where n denotes the length of the string.

---

**Note**

Use the STRING data type in new projects.

---



---

**Note**

The maximum sensible length is determined by the width of the display. It differs from device to device.

---

### 5.3.13.2 Decimal places with tags

#### Decimal places with floating point tags

With Windows-based systems, the possible number of decimal places with the decimal notation is always "0 to 16".

If you wish to display tags with decimal places in an input/output field, the tags must have a suitable format. The format differs as a function of the PLC you selected, for example:

PLC driver	Format
SIMATIC S5	KG
SIMATIC S7 SIMATIC WinAC SIMATIC 500/505	REAL
No PLC	FLOAT

#### Decimal places with integer tags

If you configure decimal places for integer tags, you can also enter values with decimal places in input/output fields, for example, and display them. If the actual number of places exceeds the configured number of places, the places that follow are truncated without being rounded up or down.

The table shows the interrelationship between the configured number of places, the value displayed or entered on the operating unit and the actual value on the PLC:

Configured decimal places	Value entered or displayed on the operating unit	Value on the PLC
0	2275	2275
1	227,5	2275
2	22,75	2275
3	2,275	2275

**Special features when entering limit values for integer tags**

Limit values for tags are entered without a point. The constant for the limit value is interpreted differently, depending on whether or not you specify decimal places for the tag.

The number of specified decimal places determines how many places of the limit value are interpreted as decimal places.

The table shows an example.

Configured decimal places	Specified limit value	Actual limit value in ProTool
0	3871	3871
1	3871	387,1
2	3871	38,71
3	3871	3,871

---

**Note**

Limit values for tags are allowed only in DECIMAL.

---



### 5.3.13.3 Address and type of a tag

#### Dependence of tags on the PLC

The address and the data type (the data format) of a tag depends on the PLC.

---

**Note**

You will find information about the various PLCs at "Start" > "SIMATIC" > "ProTool" > "ProTool Information System" in the topic "Process communication".

---

#### Available PLCs

SIMATIC S5 - AS511

SIMATIC S5 - DP

SIMATIC S7-300 or 400

SIMATIC WinAC

SIMATIC S7-200

SIMATIC 500/505

SIMATIC 500/505-DP

SIMOTION

OPC (Client/Server)

Allen-Bradley DF1

Allen-Bradley DH485

LG GLOFA-GM

MITSUBISHI FX

MITSUBISHI Protocol 4

Modicon Modbus

Telemecanique Uni-Telway

GE Fanuc SNP/SNPX

Omron Host-Link

## 5.4 Creating text or graphic lists

### Purpose

Text or graphics are often more meaningful than abstract values. For example, the pieces of text `full` and `empty` or two graphic symbols present the state of a tank more lucidly than the corresponding numerical values. Which is why ProTool gives you the opportunity to configure text or graphic lists.

These text or graphic lists are lists in which you assign an element from the list to each tag value.

### Usage

With text or graphic lists, for example, you can label buttons dynamically, display text or graphics in output fields and select texts for input in input fields. To a large extent, this eliminates misinterpretations in the display and operator errors during input.

You will find further information on using text or graphic lists on a screen in the topics dealing with input fields, output fields and buttons.

### Text list

A text list assigns text to every value of a tag. At run time, the tag value determines which text is selected from the list and displayed, for example, in an input/output field on the operating unit.



To create a new text list, select the button illustrated on the toolbar.

---

#### Note

The configurable range of values in a text list ranges from 0 to 32767 for a maximum of 3500 different blocks of text.

---

### Graphic list

A graphic list assigns a graphic to every value of a tag. At run time, the tag value determines which graphic is selected from the list and displayed, for example, in an output field on the operating unit.



To create a new graphics list, select the button illustrated on the toolbar.

## 5.5 Creating graphics and text

### Using graphics

You can configure static and dynamic graphics in ProTool:

- **Static graphics**  
Static graphics are static display elements that are not connected to the PLC. They cannot be modified at run time on the operating unit. Use static graphics, for example, for displaying your system or as explanatory symbols for dynamic display elements and controls.
- **Dynamic graphics**  
Dynamic graphics are display elements that are connected to the PLC. The position of the graphic on the operating unit can change at run time. Use dynamic graphics to display the fill level of a tank, for example.

### Creating graphics

ProTool features an option of embedding external graphic editors via the OLE interface. In this way you can create graphics with your usual applications without having to become familiar with a new graphics program.

Once created, a graphic is embedded in ProTool either as a bitmap or as a vector graphic, depending on the graphics program you are using.



You configure a static graphic on the open screen in the "Screen Objects" toolbar by selecting the button shown.



You configure a dynamic graphic on the open screen in the "Screen Objects" toolbar by selecting the button shown.

---

#### Note

To make a graphic "operable", configure an invisible button which you can place over the graphic.

---

### Possible graphics programs

You can use the following program versions to create vector or pixel graphics.

- MS Paint, all versions
- MS EXCEL, 2000 and XP
- MS PowerPoint, 2000 and XP
- Paint Shop Pro 7.0
- Micrografx Designer 9.0
- Corel Draw 10
- Corel Photo Paint 9.0  
We recommend you not to create graphics in Corel Photo Paint, since Corel Photo Paint uses a disproportionately high amount of configuration memory.
- AutoSketch 7.0 Build 54

Some problems have become known with earlier versions in the meantime.

If you insert an Excel spreadsheet as an OLE object in your configuration and have enabled the "Gridlines" option in Excel, the gridlines will not appear correctly on the operating unit so that you have to disable the "Gridlines" option. This effect does not occur in Excel 2000.

### Using Excel and Word objects

If you use Excel or Word objects in a ProTool screen, you should perform subsequent modifications in the original file and then insert the modified object back into your configuration.

### Using AutoSketch

If you wish to create a new AutoSketch object, create a graphics field in your ProTool configuration. In the "Graphic Object" dialog box, click "New" and select the "New Object" check box. After you have twice confirmed the message that the drawing does not contain any objects that might be copied to the Clipboard, AutoSketch is started as an OLE server.

If you wish to integrate an existing AutoSketch object, start AutoSketch and open an existing graphic. Select the Graphic and copy it to the Clipboard. Insert the material on the Clipboard in your ProTool configuration by choosing "Edit > Paste" from the menu.

### Optimum color presentation on the operating unit

This setting is only available when you create projects for devices with Windows CE, because these operating units only support 256 colors.

If the depth of color of the configuration computer is set to a value greater than 256 colors, you can choose between the following settings when inserting graphics in your project, in order to achieve optimum color presentation on the operating unit:

- "Pattern with error scatter" (system default setting): The colors of the screen are mixed to create different types of shading in this manner.
- "Similar color": The individual screen colors are replaced by a new color from the system colors, which comes closest to the original color of the destination device.

To obtain optimum results, use the "Similar color" setting in vector graphics and the system default setting "Pattern with error scatter" for color photographs".

### Smooth display

If the "Smooth" setting is enabled, an anti-alias filter for replacing missing pixels due to interpolation is used when increasing or decreasing the size of a graphic. This setting results in softer transitions in photo-realistic displays. In the case of drawings, the display quality is enhanced when zooming out of graphics in particular, since the complete screen information is retained.

### Color presentation on the operating unit

With pixel graphics, if you notice differences between the colors displayed on the operating unit and those configured in ProTool, proceed as follows:

On the configuration computer, choose on the Start menu "Settings" > "Control Panel" > "Display" and on the "Settings" tab, select at "Colors" the setting "True Color".

Retain this setting for configuring and compiling the project.

## 5.5.1 Making optimum use of graphics

### Optimizing graphic applications

It may take a relatively long time to load and save projects, especially if you use a large number of Designer, Corel Photo Paint or Corel Draw graphics.

- **Save As**  
Save at "Save As ..." from time to time. This optimizes data storage and makes the \*.pdb file smaller.
- **Graphics editor**  
Avoid large numbers of Designer, Corel Photo Paint and Corel Draw graphics. Program sections of these editors are used when graphics are processed. Some of these editors are very slow. The best times are achieved with Paint.
- **Reduce the color depth**  
The memory requirements for graphics increases in proportion to the number of colors you use. Therefore use only the absolutely essential color availability in your graphics.

### Efficient creation of graphics

To create graphics as efficiently as possible, take the following recommendations into consideration:

- Use pixel graphics primarily in those cases where the size of the window is fixed, as with icons for soft keys or buttons, for example.
- You should use vector graphics for graphics whose size has to be subsequently modified.
- Always create your graphics with the same aspect ratio as the clicked and dragged field in ProTool. In doing so, it should be borne in mind that a graphic having a size of, say, 10 cm × 20 cm cannot be displayed faithfully from a pixel point of view in a field of 1 cm × 2 cm. Information is lost in such a case.

### Saving large graphics

If a large number of graphic elements are used in a configuration (for example, as a background image or in graphics lists), there may be storage problems in the following cases. The reason is that the working memory is too small. The following cases can occur:

- While you are configuring, you reach the limit of the working memory. The following message appears: "Could not find the server application program, the source file or the element..."
- When you are compiling, you reach the limit of the working memory. The following message appears: "Insufficient memory."
- You open a sizable configuration on a computer that does not have sufficient memory. The following message appears: "Could not open document."

If one of these cases occurs, check, and if necessary, modify the size of the virtual memory. Increase the size of the physical random access memory (RAM).

## 5.5.2 Static text

### Using static text

Static text is text that is not linked to the PLC. It cannot be modified at run time on the operating unit. Use static text, for example, to label controls and output fields. The relative importance of different text strings on a screen can be illustrated by means of different fonts and properties (size, color, flashing etc.).

---

#### **Note on operating units with Windows CE**

Avoid tabulators in static text, since they are replaced by other characters.

---

You must configure static text separately for any language available on the operating unit.



Configure static text by selecting the symbol illustrated.

### 5.5.3 Vector graphic elements

#### Purpose

Vector graphic elements are vector graphics of simple geometrical basic shapes - for example, lines, circles and rectangles. By using graphic elements, you can draw simple vector graphics directly in ProTool without having to use an external graphic editor. Use vector graphics, for example, for displaying your system or as explanatory graphics for dynamic display and controls.

Vector graphic elements can be dynamically modified - in other words, their position can be dynamically changed at run time.

#### Selecting vector graphic elements

You can select the following drawing tools for creating simple vector graphics from the "Screen Elements" toolbar:



Line

You use this tool to draw different types of line. You can set the color, width, type and ends of a line. This means, for example, that you can draw continuous or dashed lines with squared, rounded or arrow-shaped ends.



Polyline

You use this tool to draw open polylines and polygons. You can set the color, width and type of a line. With polygons, you can also specify the fill-in color.



Ellipse/circle

You use this tool to draw ellipses and circles. You can set the color, width and type of a line and the fill-in color.



Rectangle/square

You use this tool to draw rectangles and squares. You can set the color, width and type of a line and the fill-in color.



Rounded-edge rectangle and square

You use this tool to draw rectangles and squares having rounded edges. You can set the color, width and type of a line, the fill-in color and radius of the rounded edges.



## 5.6 Creating trends

### Usage

- Real-time trend With a realtime trend, only one trend value at a time is read from the PLC for each time unit (clock pulse) and added to the trend displayed on the operating unit. Real-time trends are suitable for displaying slow, continuous patterns.
- Pattern trend With a pattern trend, all trend values are read simultaneously from the PLC and displayed as a trend on the operating unit by setting a trigger bit. Pattern trends are suitable for displaying rapid changes if the trend variation, seen in its entirety, (profile) is more interesting than individual values.

### Trigger:

The trigger type defines how trend data is read in:

- Clock trigger (time triggered)  
In the case of a clock trigger, the data that has to be acquired in the trend is read from the PLC in a fixed, adjustable time slot. To do this, select "Read Continuously" on the "Options" tab when configuring the tag. If the configured number of samples has been reached, the oldest value is overwritten by every new one.  
By means of the specified time slot and the number of samples you want to have displayed, you set the period of time that has to be covered by the trend data. This type of trigger is suitable for "real-time trends".
- Bit trigger (event triggered)  
In the case of a bit trigger, data to be acquired on a trend is read in on an event-triggered basis. The event is triggered by the PLC as a result of setting a defined bit. There are two possibilities of reading in trend data:  
Single-value acquisition:  
Only one value is read from the PLC every time a bit is set. Single-value acquisition is suitable for displaying "real-time trends".  
Buffered data acquisition:  
Buffered data are read out from the PLC as an entire block every time a bit is set. Buffered data acquisition is suitable for displaying "history trends".

### Communication areas for bit-triggered trends

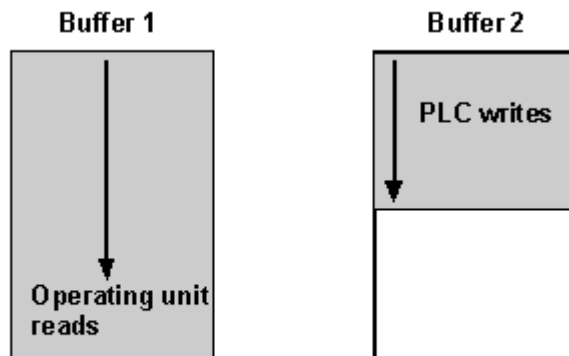
For you to be able to configure bit-triggered trends, you have to set communication areas at "Area Pointers" in the project window and create them on the PLC. The operating unit and the PLC communicate with each other via these areas:

- "Trend request area": This area can be used by the PLC to evaluate which trend is currently being displayed on the operating unit.
- "Trend transfer area 1": This area is used to trigger the trends.
- "Trend transfer area 2": This area is required only for trends that you configure with a switch buffer.

A specific trigger bit is assigned to every trend in all bit areas. If, for example, you have assigned trigger bit 4 to a trend, that trend will be identified by bit 4 in all bit areas.

### Switch buffer

A switch buffer is a second buffer which you can create for a pattern trend. While the operating unit is reading the trend values from buffer 1, the PLC can already write the new values to buffer 2. When the operating unit is reading buffer 2, the PLC is writing to buffer 1. The switch buffer prevents the PLC from overwriting values while the operating unit is reading the trend.



**Trend buffer is full, bit is set in trend transfer area 1**

*Switch buffer: reading and writing trend values simultaneously*

### Display archive data as a trend

Data from an archive are displayed as a "history trend".

An application for trends of this kind are, for example, a turn-on operation or a temperature pattern upon heating a furnace.

## 5.7 Configuring schedulers

### Purpose

A scheduler defines a periodically recurring or unique time at which a particular function is to be executed.

When a scheduler has run its course, the configured function is triggered. Typical applications are, for example, print and archive functions that have to be executed regularly at a particular time slot.

The following table shows the available scheduler types:

Scheduler type	Required time entries
at x minute intervals	
hourly	Minute
daily	Hour, minute
weekly	Day of the week, hour, minute
monthly	Day, hour, minute
annually	Month, day, hour, minute
once only	Year, month, day, hour, minute

### Changing schedulers at run time

You can change the scheduler intervals on the operating unit for the following scheduler types at run time:

- daily
- annually
- once only

For this, configure the scheduler to a time tag and the time tag to a date/time field. By making an entry in the date/time field, the operator can reset the time tag and thus set the scheduler.

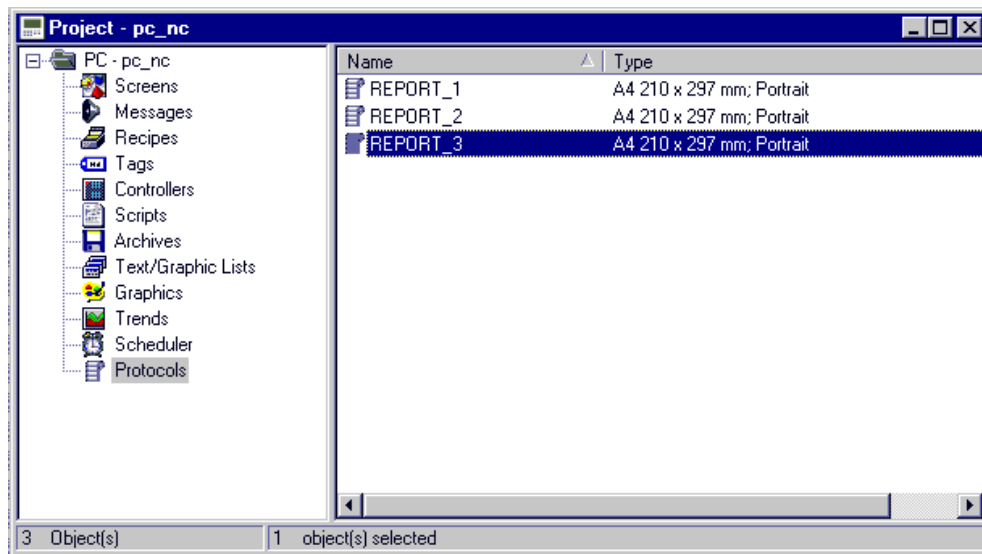
## 5.8 Creating reports

### Usage

On visualization systems, reports are used to document process data and executed production cycles. A report, therefore, documents the operations carried out during a working shift.

With ProTool, you can perform the following, for example:

- Configure a report that is output at the end of a shift and in which, for example, downtimes are documented.
- Configure a report that can be used in product or quality testing (ISO 9000) as documentation.















### *Examples of reports in the project window*

For a report, define the contents, layout and the event at which you want printing of the report to be triggered. You can configure up to 10 different reports. Every report can be up to 10 pages long.

## Report contents

Set the contents of the report in the report editor. You can include the following objects in your report:

Object	Name	Description
	Graphics	With the graphic field, you can use graphics from external graphics programs in your report.
	Text	You can use the text field to include static text in the report. Various formats are possible: You can present text in various fonts, for example, enter text on more than one line, and align it vertically or horizontally.
	Output field	Using the output field you can display process values. Different display formats - for example, decimal and binary - are possible. Output fields can be formatted (such as 3D effects, colors, etc.).
	Symbolic output field	A symbolic output field shows text or a graphic, instead of a numerical value. Example: Instead of the values 0 and 1, you can output the texts <code>Motor OFF</code> and <code>Motor ON</code> in the report.
	Page number	The corresponding number of pages is output at the point you paste this symbol in a report. If you have put a cover sheet at the front of the report, a page number will not appear on it.
	Date/Time	You can incorporate the system time in your report using Date/Time.
	Viewing a graphic	With the graphic view, you can dynamically position graphics from other graphics programs in your project. The current status of the graphic is then mapped to the report.
	Graphic list	The graphics list displays a graphic from a list to make the current process status more comprehensible. Example: Instead of the values 0 and 1, you can display a graphic for a closed valve and a graphic for an open valve.  

Object	Name	Description
	Message Printout	You use the illustrated symbol to read out the contents of the message buffer or a message archive to a report.  Message Printout is always the last object on a page – you cannot insert any more objects after it on the page. The messages are always read out in full - if the page, on which the message printout was configured, is not sufficient, follow-on pages are printed automatically.
	Print Recipe	You use the illustrated symbol to output the contents of data records for selected recipes in a report.
	Vector graphics	A vector graphics elements is, for example, a line, a circle or a rectangle. You will find an overview of objects for vector graphics in "Vector graphic elements".

## Layout

To define the layout of your report, select the report in the project window and choose "Edit" > "Properties" from the menu. You can configure cover sheets, headers and footers, as well as page numbers.

## Trigger event

The printed version of the report can be initiated automatically or manually at run time.

To print a report automatically at the end of a shift, for example, the `Print_Report` function is configured for the scheduler.

To initiate manual printing of the report, configure the `Print_Report` function for a function key, for example.

## Output language

By controlling the language, you can create reports in different languages. The procedure is the same as for screens in the screen editor. At run time, the report is then printed in the language set on the operating unit.

**Special situations when printing**

For the report to be printed correctly, the printer that is connected at run time must support the paper format and report layout. Reports are printed in Graphics mode. Owing to the volumes of data involved, a network or USB printer should be used.

---

**Note**

When used on Windows CE devices, make sure which printers the operating unit is able to support.

---

**Printing tag values in reports**

The value of a tag included in a report is read and issued only at the precise moment the report is printed. In the case of multipage reports there may be a considerable time difference between the first and last page. It is therefore possible for one and the same tag to appear with a different value on, say, the first and last pages.

## 5.8.1 Printing messages in a report

### Purpose

The contents of the message buffer or a message archive will be printed in a report.

---

### Note

Message Printout is always the last object to appear on a page you cannot insert any more objects after it on the page. Messages are always printed in their entirety - if the page, on which the message printout was configured, is not sufficient, follow-on pages are printed automatically.

---



To output the contents of the message buffer or of a message archive in a report, paste the illustrated "Message Printout" symbol in the report editor.

### Properties of Message Printout

You set the contents and appearance of a report on the tabs of the "Message Printout" dialog box.

- "Messages" Here you set whether you want to log current messages or messages from an archive, select the type of message that you want to print and set the number of lines per message.
- "Filter" Here you configure the tags for the time stamp that set the range of messages (from and to) you want to include in the report.
- "Columns" Here you set the columns you want to have output. The default value is the output of all columns apart from the diagnostic capability of a message and the of the PLC from which the message originates. Further, you set the sort order and display of the columns.
- "Font", "Color", "Position" Here you define the appearance of the message printout .
- "Name" You can use this name to identify the message printout.



## 5.8.2 Printing recipes in a report

### Purpose

The data records of one or more recipes are read out to a report.

---

### Note

A Recipe Printout is always the last object to appear on a page you cannot insert any more objects after it on the page. The Recipe Printout is always read out in its entirety if the page, on which the Recipe Printout was configured, is not sufficient, follow-on pages are printed automatically.

---

### Note

There are no PLC-specific formats in a Recipe Printout, just standardized tags types such as REAL, INT, etc.



To read out recipes to a report, paste the illustrated "Recipe Printout" symbol to the report editor.

### Properties of Recipe Printout

You set the contents and appearance of a report on the tabs of the "Recipe Printout" dialog box.

- "Recipes" Here you set whether you want data records to be read out from a recipe or a range of recipes, or whether you want all recipes to read out.
- "Data records" Here you set whether you want just one data record or a range of data records to be read out, or whether you want all data records of the selected recipe(s) to be read out.
- "Columns" Here you set the appearance of the printout. For this purpose you can select data, whether you want to have explanatory titles and borders to be displayed and whether selected data should be read out by rows or by columns.
- "Font", "Color", "Position" Here you define the appearance of the recipe printout .
- "Name" You can use this name to identify the recipe printout.

### 5.8.3 Page numbers in reports

#### Purpose

This object is used to output page numbers in a report for example, in the header or footer. During printing, the number of the report page concerned is inserted at the position where you pasted "Page Number". If you have put a cover sheet at the front of the report, a page number will not appear on it.



To configure page numbers for a report, select the illustrated symbol in the report editor.

Set how the page number is formatted on the tabs of the "Properties" dialog box.

## 5.8.4 Example of creating reports





The example shows you how to create a report with a custom layout, automatic insertion of the date and contents of the message buffer and to have it printed automatically at certain times of the day.

### 5.8.4.1 How to create a report

#### Objective

Suppose you want to create a report containing a title underscored with a bar, the date and time of output, and the contents of the System Messages buffer. You want the report to be issued at 6 a.m. each day and every 8 hours thereafter.

#### Procedure:

Procedure:	
1	Double-click "Reports" in the project window. A new report opens. You can leave the default settings for the paper format and layout as they are.
2	 Click on this symbol and use the mouse to open a "text field" on the current page.
3	Type the title "Message buffer from" in the box.
4	 Click on this symbol and open up next to the text box a further box in which to display "date and time".
5	Confirm the "Date/Time" dialog box by clicking OK.
6	 Click on this symbol in order to put a line in the title and draw the line to the required length below the title. The "Line" dialog box opens.
7	Go to the "Colors" tab, and set "black" as the foreground color and confirm by clicking OK.
8	 Click this symbol to insert the "message view" for the message buffer and drag the space required for outputting the message buffer below the line you inserted.  The "Message Printout" dialog box opens. At first, all the check boxes on the "Messages" tabs are selected.

9	<p>On the "Messages" tab, deselect all messages except the "system messages" type; on the "Columns" tab, deselect the elements not required for the output line (for example, "Diagnostic capability") and confirm by clicking "OK".</p> <p>You cannot paste anything else in a report page once you have pasted a message view to it. If you want other objects to follow, you must insert a new page.</p> <p>When you exit the report editor, the report is saved as REPORT_1. You can change this setting by choosing "Tools" &gt; "Default Settings" or assign a name each time, with a maximum of 12 alphanumeric characters.</p>
10	<p>To output the report three times a day to a printer, configure a scheduler for each time you want to have the report printed: for example, at 6 a.m., 2 p.m. and 10 p.m.</p>
11	<p>Configure the <code>Print_Report</code> function for each scheduler and enter the name of the report as a parameter.</p> <p>As a result, every eight hours the operating unit will automatically send the printer a report with the contents you have selected.</p>

## 5.9 Messages

To display process and operating states or to acquire and log process data on the operating unit that you obtain on the process from the PLC, configure "messages".

Messages are normally initiated by the PLC and can be displayed on the operating unit.

### 5.9.1 Displaying operating and process states

#### Displaying operating and process states in messages

Configure messages to display process and operating states or to acquire and log process data on the operating unit that you obtain on the process from the PLC.

Messages are initiated by the operating unit or PLC and can be displayed on the operating unit.

#### Message classes in ProTool

##### Event messages

Display routine operating and process states and processes.

##### Alarm messages

Display critical or hazardous operating and process states and require operating personnel to react by issuing an acknowledgement or by processing.

##### System messages

Display states and faults of the operating unit, the PLC or the communication between them. They are issued by the operating unit (HMI system messages) or by the PLC (system messages).

#### Configuring SIMATIC S7-300/400 messages

##### S7 system messages

Display error states on the PLC or during communication with the PLC. They are initiated by the PLC and can be displayed on the operating unit.

##### ALARM\_S messages

Are configured in SIMATIC STEP 7. Should an error occur, the operating unit receives a message containing the message number, and from this number the corresponding message text is determined and read out.

##### Diagnostic events

Are entered in the S7 CPU diagnostic buffer and can be displayed on the operating unit. Their purpose is to provide above all information on quicker and easier retrieval and troubleshooting.

## Displaying messages from SIMOTION CPUs

### ALARM\_S messages

Are configured in SIMOTION SCOUT. Should an error occur, the operating unit receives a message containing the message number, and from this number the corresponding message text is determined and read out.

### Technological alarms

Display error states. They are initiated by the technological objects and can be displayed on the operating unit.

### Diagnostic events

Are entered in the S7 CPU diagnostic buffer and can be displayed on the operating unit. Their purpose is to provide above all information on quicker and easier retrieval and troubleshooting.

## For what purpose do you use messages?

The messaging system is responsible for the following tasks:

- Signaling events or states that can occur on the system or in the process:  
A state is reported immediately following its occurrence.  
The messages are displayed for the operator as a function of their significance (priority).
- Support in eliminating the cause of error conditions:  
Messages provide in-depth information on the causes of errors (diagnostics).  
The operator can then influence the process on the basis of the message. To make this easier for the operator, you can include the "Edit" button in a message view and configure a function for this button (for example, selecting a specific screen).
- Reporting: The message events are output to a printer.
- Archiving: The message events are stored for further processing and evaluation.

## 5.9.2 The components of a message

### Message structure

A message consists of:

- Message number
- Message text
- Message tag
- Help text

The following messages are displayed on the operating unit but cannot be edited in ProTool:

- The message cannot be modified in system messages, standard diagnostic results (S7 system messages) and technological alarms (SIMOTION).
- Message numbers of user defined diagnostic results can be freely defined in SIMATIC STEP 7 within certain limits.
- The message number is assigned in SIMATIC STEP 7/SIMOTION SCOUT for Alarm\_S messages.

### Message number

The message number is used to reference a message. In ProTool, you have a free choice of message number (within the range 0 to 2000) and message text.

The message number is structured in a specific structure. It consists of a number, which is a CPU-wide unique number, and specification of the message type. The message type is not displayed on the device. You will only see the number on the screen.

---

### Note

It is possible for two identical numbers to be displayed for different messages. The causes of this could be as follows:

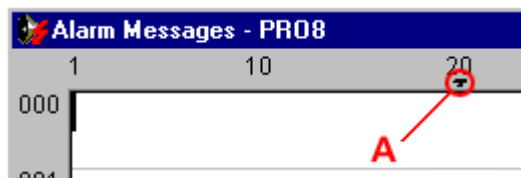
- The displayed number was assigned for several message types.
- The message number was assigned on different CPUs for different messages.

On RMOS devices there is no CPU identification of messages.

---

### Message text

Message text contains the description of a message. The length of the message text depends on the operating unit. The number of characters per line is marked by this character (A) at the top border of the window during configuration.



By choosing "Edit" > "Style" from the menu, you can select the following styles, for example, for the message text, depending on the operating unit: flashing, underscore, italic.

---

**Note**

You can also configure operator instructions as a message.

---

**Message tags**

A message can contain output fields with tags. They are also referred to as "message tags".

By choosing "Edit" > "Style" from the menu, you can select from the following styles for output fields, depending on the operating unit: flashing, underscore, italic.



Insert an output field by selecting the symbol illustrated.

---

**Note**

The values of message tags in the message buffer are updated upon all events (active, cleared, and acknowledged).

---

**Help text**

Help text containing further details on a message can be configured for every message. Help text is displayed for the operator in a separate window on the operating unit by pressing the "HELP" key.



Enter Help text by selecting the symbol illustrated or by using the menu command "View " > "Help Text".

**Other message components**

When configuring message views for certain devices, other message components can be displayed - for example, details of the error location (on the PLC).



## 5.9.3 Message properties

### Configurable properties

You can configure the following parameters for messages:

- Acknowledgement group
- Print
- Functions

The window for configuring the properties opens automatically when a message class is opened from the project window.

#### Acknowledgement group

Alarm messages can be assigned to acknowledgement groups. Configure the assignment to an acknowledgement group for every single message. If the operator acknowledges an alarm message in an acknowledgement group, all the alarm messages in that group are acknowledged simultaneously ("group acknowledgement").

#### Print

If you select "Print", the message event (active, cleared, acknowledged) is logged automatically on the printer if you selected "Print ON", at "System" > "Messages" > "Settings".

Configure printing in the "Attributes" dialog box for every single message.

If you archive messages, you can print the messages stored in an archive.

#### Functions

You can configure one or more functions for the events Active, Cleared, Acknowledged and Edit Message.

- The "Edit Message" event can only be triggered by the operator (by pressing the "Process" button).
- The "message acknowledged" event can be triggered by the system or by the operator (by pressing the ACK button or the ACK key).

---

#### Note

To enable the operator to trigger the functions configured for acknowledgement or process, the ACK and "Edit" buttons must be configured in the message view, or the ACK key must be available on the operating unit.

Alternatively, you can select the functions

`Message_View_Acknowledge_Message` and

`Message_View_Process_Message` for buttons or keys in your project.

---

## 5.9.4 Acknowledging messages

### Principle

An alarm message can be acknowledged either by the operator on the operating unit or by the PLC program. By acknowledging an alarm message, the operator confirms that he has taken note of the message.

### Single acknowledgment and group acknowledgment

When you configure messages, you can set whether the operator as to acknowledge each message singly or whether the acknowledgement will apply to a group of messages. The use of acknowledgement groups is practical in particular for messages having the same cause for example, the message of the initial error and consequential errors.

### Assigning acknowledgement groups

In the Attributes dialog box of the message editor, you can combine messages to form acknowledgement groups by assigning them the same number at "Acknowledgement".

You can allocate the messages to one of up to 16 acknowledgement groups.

The value 0 or a blank "Acknowledgement" field signifies that an acknowledgement applies only to that alarm message (single acknowledgment).

### Acknowledgement hierarchy of message views

If several message views are displayed simultaneously on the operating unit and the operator acknowledges queued alarm messages, the acknowledgement affects the different message views in the hierarchical order:

1. Message view with focus (dotted rectangle)
2. Alarm message window
3. Message line
4. Message view in the fixed screen
5. Message view in the main screen

## 5.9.5 Settings for message classes

### Configurable properties

You can configure the following parameters for message classes:

- Acknowledgement
- Printout
- Archiving
- Filter

#### Acknowledgement

Messages classified as "alarm messages" have to be acknowledged. Alarm messages are displayed until they are acknowledged.

- Individual acknowledgement: When you acknowledge a message, only that particular message is acknowledged.
- "Group acknowledgement": When you acknowledge one message belonging to an acknowledgement group, all the queued messages of that group are acknowledged as well.

#### Printout

Printout is enabled for all messages if you have selected the "Print" attribute in the "Message Settings" dialog box ("System " > "Messages" > "Settings"). The messages are printed on a printer attached to the operating unit.

#### Archiving

To archive message events, create archives. Message events are then written to the message buffer and also to an archive. The message buffer is volatile; an archive containing messages is permanent. Archiving is configured separately for each message class.

You configure archiving in the "Archives" dialog box. Open the dialog box from the "Message Settings" window.

#### Filter

You can set filter criteria for displaying messages:

- In the "Message Settings" dialog box ("System" > "Messages" > "Settings") Select at "Alarm messages" whether the first or the last Active event should be displayed first.
- In the "Message View" Properties Dialog Box Set on the "General" tab what you want to have displayed:
  - Messages
  - Message events
  - Diagnostic buffer
  - Message archive

## 5.9.6 System messages

### What are system messages?

System messages advise you of operating unit and PLC operating modes. System messages can range from information to serious errors.

A distinction is made between system messages and HMI system messages.

- **System messages** Are issued by the PLC and cannot be configured in ProTool.
- **HMI system messages** Are issued by the HMI device (such as "Data record deleted" or "Disk full").

A system message consists of a "message number" and "message text". The message text may contain internal system tags that provide further details on the cause of an error message.

---

#### Note

The operator will find a numerical list of HMI system messages, their causes and remedial action in an appendix to the equipment manuals and the "Communication Manual for Windows-Based Systems".

---

### Displaying system messages on the operating unit

System messages have to be transferred by the system to the operating unit.

Choose "System" > "Messages" > "System Messages" from the menu to set which

- SIMATIC OP HMI system messages from SIMATIC operating units (cannot be deselected)
- SIMATIC S7 (minimal): S7 PLC system messages in abbreviated form. The message numbers of the S7 system messages (with time stamp) are read out. The operator can refer to the cause of the error in his S7 manual by referring to the number.
- SIMATIC S7 (default): S7 PLC system messages in long form. In addition to the minimal display, the message text of the S7 system message is output.
- SIMOTION Messages from SIMOTION PLCs. These messages are displayed in a message view for diagnostics events, that is, not in a message window.

You determine which system messages you want to have output to the message window and how long they will be displayed on the operating unit.

S7 components and DP standard slaves can provide diagnostic information when a system error occurs. With its "Report System Error" function, SIMATIC STEP 7 features a user-friendly option to display this information in the form of messages. These system error messages can also be displayed on an operating unit by means of the ALARM\_S message procedure.

---

**Note**

The 0 setting means the display is static. The message window is not closed until the operator presses ESC.

---

**Displaying HMI system messages with message view**

You can output HMI system messages and SIMATIC/SIMOTION diagnostic messages to the operating unit using the "Message View" screen object.

**Displaying HMI system messages on the TP 170A**

Bear the following in mind for displaying HMI system messages on the TP 170A:

- If necessary, increase the default setting for the duration of display with regard to system messages at "System" > "Messages" > "System Messages".
- On screens in which HMI system messages are important, configure a "single message view".

## 5.9.7 Logging messages

### Enabling and disabling message logging

To log message events on the printer, select the "Print" check box in the "Attributes Message" dialog box in the message editor. With the check box selected, every message is logged on every change of state (active, cleared, and acknowledged).

By choosing "System" > "Messages"> "Settings" from the menu, you can select the corresponding check box to activate message logging for the project.

### Report

You also have the option of collectively reading out the messages from the message buffer or a message archive at certain times (by configuring a scheduler) or linked to certain events (such as key operation). To do this, you configure a report and include the "Message Printout" object in the report.

## 5.9.8 Displaying messages on the operating unit

### Types of message view

You can view messages in global message windows on the operating unit and configure a message view in a screen, for example.

### Message indicator

Set display of the message indicator on the operating unit by choosing "System" > "Screen/Keys" from the menu. The message indicator indicates that alarm messages are waiting or have to be acknowledged. The message indicator can be used with touch devices.

As soon as an alarm message is waiting, an alarm message window and the message indicator are displayed.

The message indicator may assume two states:

- "Flashing": as soon as at least one unacknowledged alarm message is queued.
- "Static": when all queued alarm messages have been acknowledged but at least one of them has not yet been cleared.

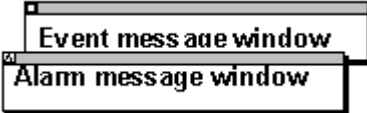
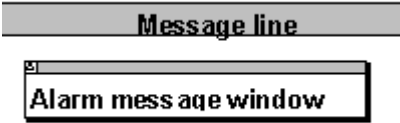
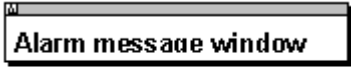

Not until all acknowledged messages have departed is the message indicator hidden. In this way it is not possible for queued alarm messages to be forgotten.

### Message window and message line

Messages are displayed in message windows or on the message line. You can still operate the operating unit when messages are queued and displayed.

- The "message line" permanently resides on the operating unit and only ever contains one message, the current one. If a separate message window has been configured for alarm messages, the message line only contains the current event message.
- Alarm messages and event messages are displayed in separate "message windows". A message window always contains all the queued messages of the particular message class. The alarm message window opens automatically whenever an alarm message arrives. For the event message window, you have to configure a control element to help you open or close the window.

You can configure the display of messages on the operating unit by choosing "System" > "Screen/Keys" from the menu.

<p>Window/window Event messages and alarm messages are displayed in separate message windows. The alarm message window opens automatically whenever an alarm message arrives; enable and disable the event message window by choosing the function <code>Display_Event_Message_Window</code>.</p>	
<p>Window/line An alarm message is displayed in the message window, whereas an event message is displayed on the message line. The message line is visible at all times. The alarm message window opens automatically whenever an alarm message arrives.</p>	
<p>Window/OFF An alarm message is displayed in the message window. Event messages are not displayed.</p>	
<p>Line/line The message line is visible at all times. Any alarm messages or event messages that are issued are displayed on the message line according to their priority.</p>	

**Message view**

Choose "Insert" > "Display Message" from the menu to configure a message view on a screen: You can set what you want to have displayed in the message view:

- Messages
- Message events
- S7 diagnostic buffer
- Contents of a message archive

You can customize the appearance and behavior of the message display to suit your project.

**Colored messages**

You can assign different colors to different types of messages in order immediately to make the message type clear to the operator. This means, for example, that you can configure alarm messages and event messages in different colors by choosing "System" > "Messages" > "Colors" from the menu.



**Sort criterion for alarm messages**

Should several unacknowledged alarm messages having the same priority be queued on the operating unit, you can select, by choosing "System" > "Messages" > "Settings" from the menu, whether the oldest (first) or the most recent (last) alarm message will be displayed first.

## 5.9.9 Message view

### Purpose

In ProTool, the default is for event messages, alarm messages and the operating unit HMI system messages to be displayed in special process windows.

As an alternative, the "message view" provides a display in tabular form of

- Messages
  - Queued messages:  
Messages are queued messages until the "cleared" message event occurs, even if the message has previously been acknowledged
  - Messages To Be Acknowledged:  
Messages are messages awaiting acknowledgement queued messages until the "acknowledged" message event occurs, even if the message has previously been cleared Since system and event messages are not acknowledged, they are not displayed among the "messages to be acknowledged".
- Message events
- Contents of the diagnostic buffer of a SIMATIC 300, SIMATIC 400 or SIMOTION-CPU: (not possible with WinAC and OPC) Since there is no "cleared" message event for diagnostic events and system messages, they are treated as queued messages as long as they are set at "System" > "Messages" > "System Messages".
- The contents of a selected message archive

You can combine the display of "messages" and "message events" from the following message classes:

- Alarm messages
- Event messages
- HMI system messages
- Diagnostic events



Configure the message view by selecting the symbol illustrated.

---

### Note

You can enable the message view while you are configuring by using SHIFT and double-click or from the "Edit" option of the shortcut menu.  
In active mode you can drag and drop a column to the required position and set the column width during the configuration phase.

---

### Operating a message view

You can display the following buttons for operating a message view on the operating unit:

- ACK: to acknowledge alarm messages
- Help: to display Help text
- Edit: to trigger functions

The different columns can be hidden or shown, whatever you want, and edited like a table on the operating unit - for example:

- Arrange column order
- Set column width
- Sorting columns in ascending and descending order (possible only with the columns "Date" and "Time").

---

#### Note

When you change the system language to "English", modify the time output simultaneously to the 24 hour format, since the standard width of the time field in the message view is geared to the 24 hour format.

---

## 5.9.10 Configuring message processing

### Usage

You can configure messages to give the operator the opportunity to process the active message. To do this, configure a suitable function for the "edit message" message event, for example, "Select Screen Permanent".

When the message has arrived, the operator can then press the "Process" button, perform the necessary steps on the screen he has opened and exit the screen.

---

### Note

The screen in which the message arrived, is not automatically redisplayed to the operator. After changing screens with the "Process" button, he can only operate the current screen.

Note that a message which you get rid off by clicking is acknowledged automatically.

---

### Steps to take to configure message processing

To make it possible to edit messages, take the following steps:

1. Open the screen in question. In the "Message View" dialog box, on the "Display" tab, configure the "Process" button by selecting the "Button for processing" check box.
2. Open "Alarm messages", for example, in the project window. Highlight the message in question and in the "Attributes" dialog box, click the "Functions" button.
3. In the "Messages" dialog box, for the "Edit Message" event, configure the required function - for example, selecting a screen in which the operator can make a meaningful response to the message.

## 5.9.11 Contents of message buffers

### Definition

A message buffer is a memory area in the main memory of the operating unit in which message events are stored in chronological order. The message buffer is a FIFO buffer of a given size and is not configured.

The size of the message buffer depends on the device.

---

### Note

When the operating unit is powered down, the contents of the message buffer are deleted. To keep your messages available for a longer period of time, you have to archive them.

---

### Displaying the message buffer

The message buffer contains all the message events of alarm messages or event messages. You must configure a control in order to display the alarm message buffer and the event message buffer.

To structure the message buffer more clearly and at the same time reduce the load on the system, you can configure controls which allow the user to delete all or part of the message buffer.

### Data storage in message buffer

Every message event is stored with the following information:

- Message number
- Time stamp consisting of time and date
- Identification of the event ("K" for Active, "G" for Cleared, "Q" for Acknowledged)
- Message text
- Value of the message tag at the time of arrival or departure

**Example**

Below is an example of output from the message buffer to the operating unit:

No	Time	Date	Status	Text
0010	11:34:26	27.03.98	K	Boiler pressure too high: 12.7 bar
0029	11:35:18	27.03.98	K	Oil supply shut off.
0010	11:38:09	27.03.98	(K)Q	Boiler pressure too high: 12.7 bar
0029	11:40:47	27.03.98	(K)Q	Oil supply shut off.
0010	11:58:08	27.03.98	(KQ)G	Boiler pressure too high: 7.9 bar

**Note**

If the new status of a message in the message buffer is not known (for example, when the CPU is overloaded or being updated after connection malfunctions), only the message status prior to the update is shown in the message buffer, such as (GKQ).

## 5.9.12 Communication areas for messages

### Communication area required for operating unit/PLC communication

For communication between the operating unit and the PLC to function properly, establish an assignment to the following communication areas in the project window of your ProTool project at "Area Pointers": "Event messages" and/or "Alarm messages".

These areas are imperative if you configure event messages and alarm messages.

They must be chosen at least large enough for a bit to be available for every configured message. If the communication area is not made large enough, a warning will be issued during compilation of the project.

If you wish, you can create the following communication areas:

- PLC Acknowledgement
- OP Acknowledgement

### Optional communication areas: Acknowledgement areas for messages

If the PLC is required to send an acknowledgement automatically, create the following acknowledgement areas in the project window at "Area Pointers".

- "PLC Acknowledgement"  
The acknowledgement bit is set by the PLC program and thus causes acknowledgement of the corresponding alarm message to be displayed on the operating unit.  
The acknowledgement area "PLC acknowledgement"
  - must be contiguous with the associated alarm message area,
  - must have the same acquisition cycle as the alarm message area,
  - must not be longer than the associated alarm message area.For SIMOTION and OPC PLCs, this acknowledgement area is set in the associated alarm message area.
- "OP Acknowledgement"  
The operator acknowledges an alarm message on the operating unit and thus sets the acknowledgement bit of this alarm message on the PLC. When he does so, the entire acknowledgement area is transferred to the PLC. The "OP Acknowledgement" acknowledgement area must not be longer than the associated alarm message area.

### 5.9.13 Message procedure

#### Message procedures in ProTool

The message procedure identified the transfer path of messages and thus also the parameters of the messages.

ProTool supports two message procedures:

- The message bit procedure This is the default message procedure in ProTool. The operating unit detects the messages by setting a corresponding bit on the PLC.
- The ALARM\_S message number procedure This is the procedure by which "system fault message" (SFM) type messages are transferred from the PLC to the operating unit.



### 5.9.13.1 Message bit procedure

#### Principle

With the message bit procedure, the operating unit detects the arrival, departure or acknowledgement of a message by setting a bit in the address area, the bit being assigned by choosing "System" > "Area Pointers" from the menu to the event/alarm message area.

The setting, resetting and acknowledging of the bit is known as a "message event".

- |                 |   |
|-----------------|---|
| Active (K)      | The message bit has been set and the message is waiting to be displayed.  |
| Cleared (G)     | The message bit has been reset because the cause that gave rise to the message no longer exists.                                    |
| Acknowledge (Q) | With alarm messages only:<br>The operator (or the PLC) acknowledges the message to confirm that note has been taken of the message. |

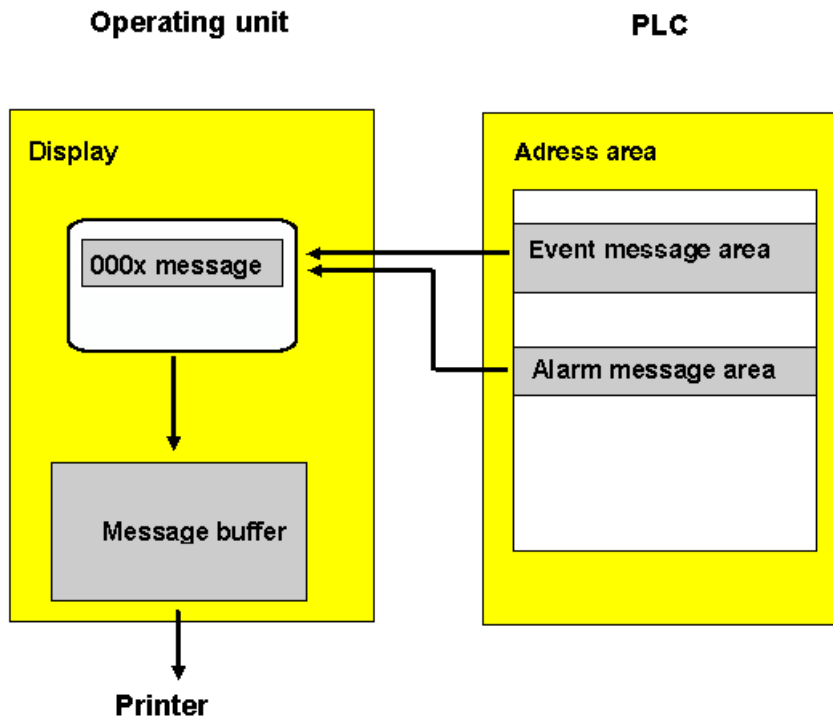
The message events have been acquired by the operating unit and time stamped.

The operating unit automatically enters all message events in the message buffer. In this way the message events can be viewed at a later time again.

#### Communication areas for messages

The display of messages on the operating unit is initiated by the PLC by a bit being set on it in a defined communication area.

You can see which communication areas for messages have to be created on the PLC in the following figure:



*Displaying messages*

### 5.9.13.2 Message number procedure ALARM\_S

#### Overview

In this section you will learn how ALARM\_S differs from the message bit procedure and how the system behaves in the event of critical bottlenecks.

#### Requirements

The display of ALARM\_S messages on the operating unit can be configured only if you have integrated ProTool in SIMATIC STEP 7 and the PLC you are using is a SIMATIC S7-300 or a SIMATIC S7-400 or a SIMOTION.

---

#### Note

For a SIMATIC PLC, it is far simpler to configure ALARM\_S messages for process diagnostics if you have installed S7-PDIAG.

---

#### What is ALARM\_S?

ALARM\_S is a message number procedure. The message numbers are assigned by the system when you are configuring in SIMATIC STEP 7 or SIMOTION SCOUT. The numbers are used as the basis for the specific assignment of the correct message text.

When a fault arises, the operating unit receives a message containing the message number. On the basis of the number, the corresponding message text is identified and displayed.

The PLC stores not only the status of the message (arrived, departed, acknowledged) but also the time. This information is retained after the messages have been sent, too, so individual network components (for example, operating units) can log on later and update.

#### Types of ALARM\_S messages

The messages you can select at Alarm\_S differ as follows:

- alarm messages
- event messages
- messages of "Report System Error" (SFM) type

You can configure in SIMATIC STEP 7 or SIMOTION SCOUT whether an ALARM\_S message needs to be acknowledged or not.

## Configuring ALARM\_S messages

When configuring messages, you can either enter simple unformatted text or make use of the facility for device-specific message configuration. Depending on the display unit in question, you can then specify such effects as flashing text and define additional information.

ALARM\_S messages are not configured in ProTool, but:

- For SIMATIC S7-300/400-CPU in SIMATIC STEP 7:

Configuration using S7-PDIAG

Configuration in S7-GRAPH or S7-HiGraph

Call these using SFC17, SFC18

---

### Note

Diagnostic events generated in the S7 CPU by SFC52 are not supported by ProTool/Pro RT and are not displayed on the operating unit.

---

- For SIMOTION-CPU with SIMOTION SCOUT

The advantage of this is that the messages are configured just once and are then available throughout the project for all operating units and can be displayed.

You can find detailed reference information about the procedure in the online Help in SIMATIC STEP 7, SIMOTION SCOUT and in the relevant option packages.

## Advantages of ALARM\_S

As compared with the message bit procedure, ALARM\_S has the following advantages:

- ALARM\_S is an active message procedure. When a message is issued, the CPU actively notifies all networked units. The operating unit is relieved of the task of continually polling the message area.
- The process data always precisely reflect the situation at the time of the message. This cannot be guaranteed with the message bit procedure.
- The time stamp precisely indicates when an event occurred even if the operating unit is not connected until a later time.

## Messages of "Report System Error" (SFM) type

S7 components and DP standard slaves can provide diagnostic information when a system error occurs. With its "Report System Error" function, SIMATIC STEP 7 features a user-friendly option to display this information in the form of messages. These system error messages can also be displayed on an operating unit by means of the ALARM\_S message procedure.

If the operating units you are using only have a small amount of application memory, you can inhibit the transfer of "Report System Error" messages.

**Display classes**

You can assign different display classes to individual messages in SIMATIC STEP 7. When configuring in ProTool, you then can then choose a specific selection of display classes for an operating unit. In that way you can distribute the messages selectively between different display units.

**Priorities**

You can assign the messages different priorities when configuring ALARM\_S messages as well.

---

**Note**

Make sure that you assign messages that can bring about consequential errors a higher priority than the consequential errors themselves.

---

### 5.9.13.3 Display classes for ALARM\_S messages

#### What are display classes?

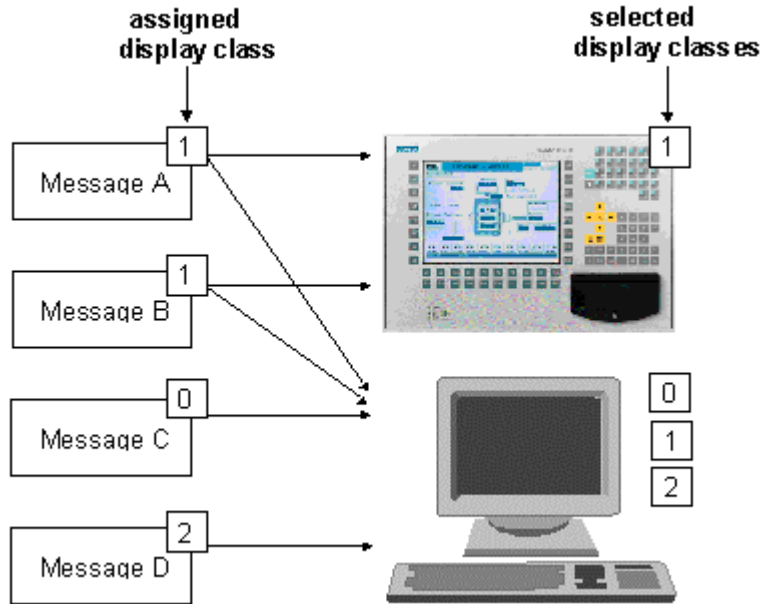
An S7 CPU always issues ALARM\_S messages to all stations that are logged in. However, it may be that you do not want to display all messages on a particular operating unit in order to avoid a torrent of messages, for example. In that case, specific messages could be displayed just on a control desk, for example.

In order to be able to control the display of messages so selectively, in SIMATIC STEP 7/SIMOTION SCOUT each message can be assigned what is termed a "display class". In all, there are 16 display classes (display classes 0 to 15).

#### Example

Messages that are to appear on the machine might be assigned display class 1, those that are to appear on the control desk, display class 2.

The various operating units then only analyze those messages that belong to specific display classes. Any other ALARM\_S messages are immediately discarded.



*Display classes determine which groups of messages are to be displayed on the operating unit*

### **Specifying display classes**

The display class to which an ALARM\_S message belongs is determined when the message is configured in SIMATIC STEP 7/SIMOTION SCOUT. For a precise description of the procedure, please consult the information on message configuration in your SIMATIC STEP 7 or SIMOTION SCOUT documentation.

If you do not specify a display class in SIMATIC STEP 7, the message concerned is automatically assigned to display class 0.

### **Selecting display classes**

In ProTool you can specify which display classes are to be displayed on the operating unit being configured.

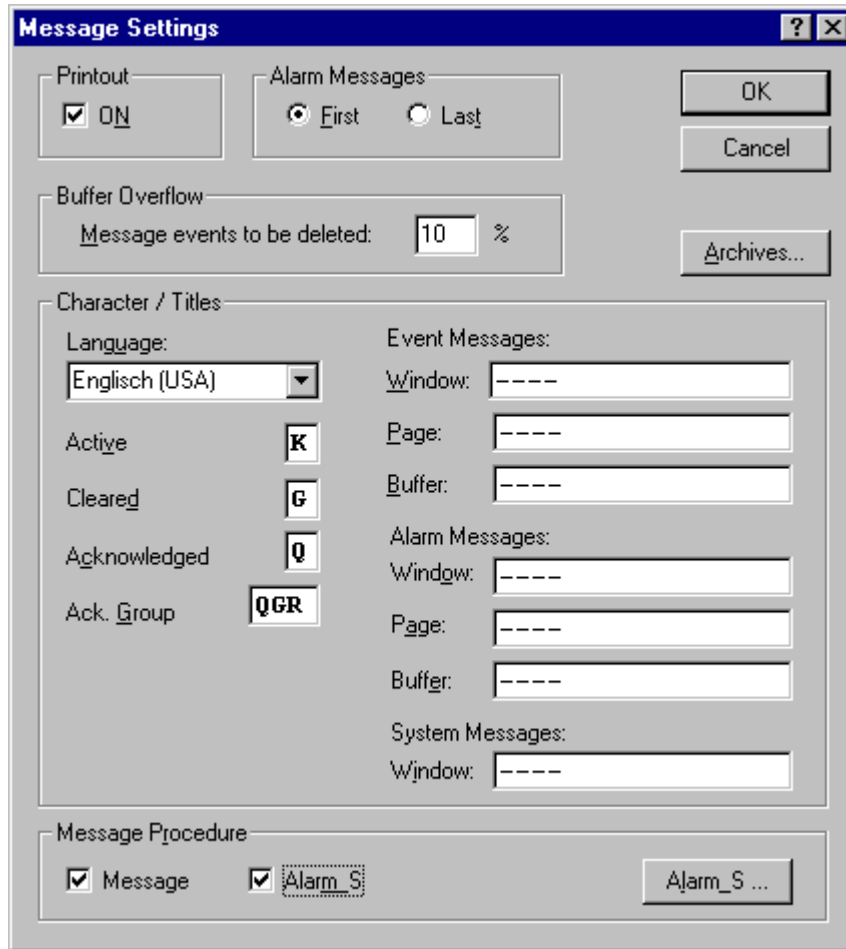
You make your selection by choosing "System" > "Messages" > "Settings" from the menu. You will find a detailed description in "Setting the message procedure and selecting the display classes".

If you do not make a selection, all display classes (display classes 0 to 15) are displayed as the default.

### 5.9.13.4 Setting the message procedure and selecting the display classes

#### Principle

To specify the message procedure(s) you are intending to use, go to the "Message Settings" dialog box. To access this dialog box, choose "System" > "Messages" > "Settings" from the menu.



Select the required message procedures from the lower part of the dialog box. Once you have selected "ALARM\_S", you can use the "Alarm\_S" button to set the messages that will be displayed on the operating unit.



If you want to do without "Report System Error" (SFM) type messages, deselect the "SFM" check box. Messages of this type are not transferred and do not take up memory on the operating unit.

---

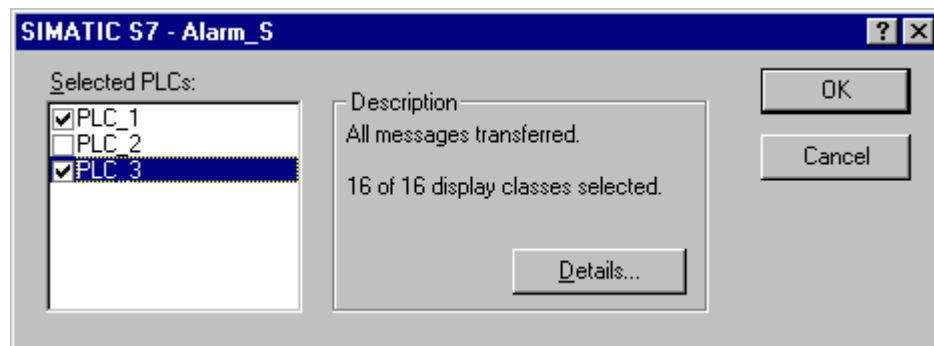
### Note

If ALARM\_S is the only message system you selected in ProTool for a project, the entries for event messages and alarm messages will no longer be offered in the ProTool project window.

---

## Selecting display classes

When you click the "Alarm\_S" button, the "SIMATIC S7 ALARM\_S" dialog box opens.



*"SIMATIC S7 ALARM\_S" dialog box*

In this dialog box you can set, individually for each PLC on your system, the messages to be displayed on the operating unit. By default, all messages are sent.

---

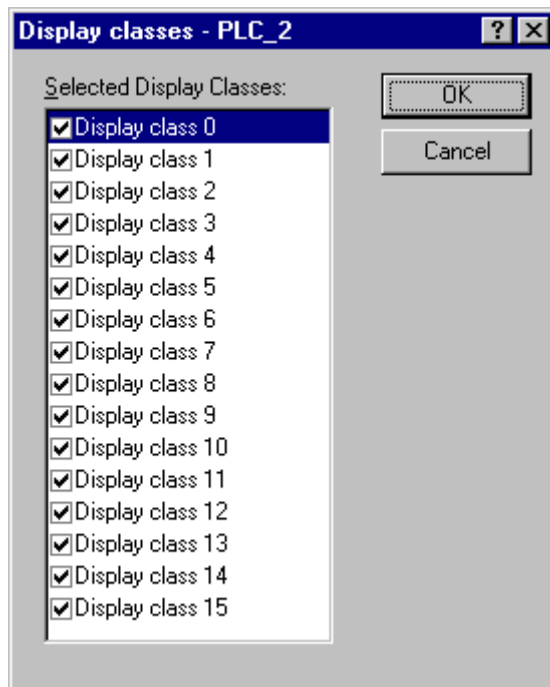
### Note

By carefully targeting your selection to specific display classes, you can restrict the display on the configured operating unit to very specific messages. Other messages can then be displayed on another device, such as a control desk.

---

The left-hand column of the dialog box shows you a list of all the PLCs on your system. For each PLC you can cause the operating unit to display no messages, a selection of messages or all messages.

When you click the "Details" button, the "Display Classes" dialog box opens.



*"Display Classes" dialog box*

Display classes will have been defined when the PLC was being programmed in SIMATIC STEP 7. At the same time each message will have been assigned to one or other of these display classes.

You can now select individual display classes and define which messages will be displayed on the operating unit.

---

**Note**

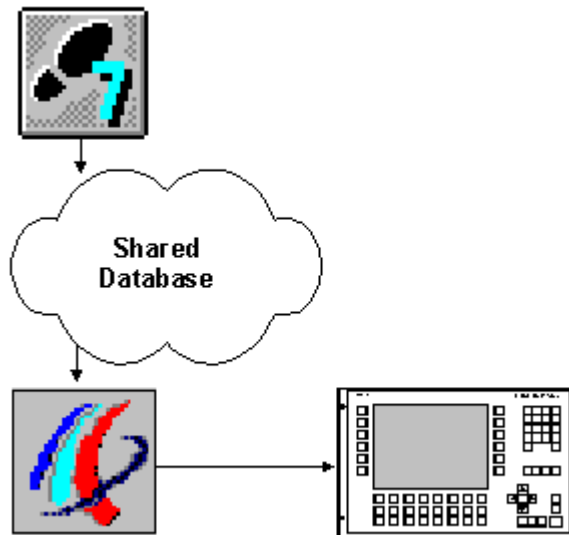
Messages of the "Report System Error" type are always displayed on the operating unit irrespective of the number of selected display classes when the "SFM" check box is enabled in the "Message Settings" dialog box.

---

### 5.9.13.5 Incorporating ALARM\_S messages

#### Configuring in SIMATIC STEP 7

With message configuration in SIMATIC STEP 7, the stored blocks of text and attributes are placed in a database shared by ProTool. During the process of compiling the project, ProTool automatically imports the necessary data and subsequently downloads it to the operating unit.



#### *Configuring and downloading ALARM\_S messages*

It is therefore important that the shared database is always up to date during the compilation process and that data synchronization is active.

### 5.9.13.6 Updating the operating unit

#### Principle

The S7 CPU only stores information about pending messages, which means that once all events (arrival, departure, acknowledgement) have occurred, the message is deleted from the S7 CPU.

A network operating unit which lost its connection to the S7 CPU can log on to the S7 CPU again and get an update on currently pending messages.

When obtaining an update, therefore, the operating unit processes any missing events automatically if a message is not known to the S7 CPU but the arrival and acknowledgement events are not entered on the operating unit.

The events are entered and labeled in the message buffer on the operating unit.

#### Example 1: Standard situation

Entries for a message in the message buffer:

No	Time	Date	Status	Text
3029	14:28:08	27.03.99	K	Boiler 13: Temperature 190 degrees Notify shift supervisor.
3029	14:31:12	27.03.99	(K)Q	Boiler 13: Temperature ### degrees Notify shift supervisor.
3029	14:40:37	27.03.99	(KQ)G	Boiler 13: Temperature 120 degrees Notify shift supervisor.

If a message contains process values, then those values that were available when the message event arrived or departed are stored in the message buffer. In the case of the "acknowledged" message status, the operating unit does not acquire any current process values. The characters ### stand for the value.

**Example 2: Lost connection situation**

The operating unit logs back onto the S7 CPU following a lost connection. By that time the message is no longer stored in the S7 CPU. The following is displayed on the operating unit.

No	Time	Date	Status	Text
3029	14:28:08	27.03.99	K	Boiler 13: Temperature 190 degrees Notify shift supervisor.
3029	14:52:11	27.03.99	(KQG)	Boiler 13: Temperature ### degrees Notify shift supervisor.

As the message is no longer stored in the S7 CPU, the operating unit will not be able to read its process values. These are replaced by ### characters.

**Note**

After an interrupt, the new status of a message in the message buffer is not known. Only the message status before the update is shown in the message buffer in this instance, for example (KQG). These message states cannot be displayed in the archive and are displayed as ().

### 5.9.13.7 Resource consumption of messages

#### ALARM\_S messages

ALARM\_S messages are configured in SIMATIC STEP 7/SIMOTION SCOUT. The data is then stored in a shared database, imported during the process of compiling the ProTool project and finally downloaded to the operating unit. That means that ALARM\_S messages use up resources on the operating unit.

#### Economizing on storage space

There are basically two ways in which you can economize on storage space on the operating unit:

- The following applies to all message procedures: the shorter the message text, the less storage space will be required.
- To economize on storage space on the operating unit of Windows-based systems, you can also deselect the SFM" check box in the "Message Settings" dialog box (choose "System" > "Messages" > "Settings" from the menu). Messages of this type will then not be transferred to the operating unit.

You will find more information in the topic called "System limits". This section contains information about the memory requirement of projects, on the basis of which you can assess whether or not the memory of your operating unit is large enough for the intended project.

---

#### Note

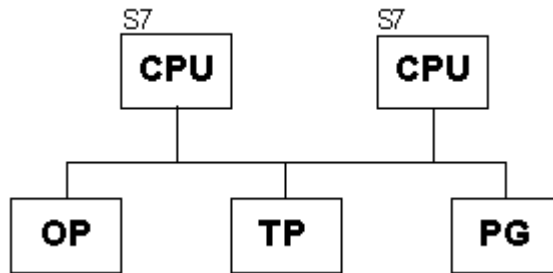
In SIMATIC STEP 7 there are restrictions regarding the number and size of tags that can be used within a message. For more precise details, please consult your SIMATIC STEP 7 documentation.

---

### 5.9.13.8 Communication sequence in the event of ALARM\_S messages

#### Logging On For ALARM\_S

More than one station (for example, more than one OP, PU, etc.) on a network can log on for ALARM\_S messages. Each station that wants to display ALARM\_S messages logs on to the CPU for ALARM\_S.



*MPI network*

The number of nodes that can log on to a CPU for ALARM\_S depends on the CPU. You will find further information on this in the equipment manual for the S7 CPU.

#### When a message event occurs

When a message-triggering event occurs, the CPU actively sends telegrams to that effect to all stations that have logged on. The message number identifies the corresponding message text that has previously been downloaded to the operating unit.

This means that the CPU does not have to be actively scanned for messages by regular polling as was the case with the message bit procedure. The operating unit and network are completely relieved of that burden on the system.

#### Time stamp

With the ALARM\_S procedure, the time stamp is not issued by the operating unit but by the CPU. The messages are stored in chronological order on the basis of their time stamp in the message buffer on the operating unit – even if they originate from different CPUs.

#### Information stored

The CPU stores not only the time of the message but also the status (active, cleared, acknowledged) and any process parameters. That information is retained until a message has been completely processed - in other words, until it has arrived, departed and been acknowledged. On the operating unit, the information in the message buffer is kept even longer.

### Restarting the S7 CPU

When S7 CPUs are restarted, it is possible - though dependent on the hardware configuration of the CPUs - that the CPU will delete all queued ALARM\_S messages.

---

#### Note

Earlier versions of S7-300 CPUs cannot report this restart to the operating units concerned. As a consequence, messages are shown as being queued on the operating unit even though the CPU has deleted them already.

---



### 5.9.13.9 Acknowledging ALARM\_S messages

#### ALARM\_S messages requiring acknowledgement

When a message is acknowledged by a station on the network, the CPU is immediately notified. The CPU then distributes the acknowledgement to all connected stations. It is only to this message that the operating unit responds - in other words, only at this point does it enter the acknowledgement in the message buffer.

---

**Note**

The CPU issues a time stamp for the acknowledgement event but does not store it. Later, it is no longer possible to determine if and when a message has been acknowledged.

---

#### ALARM\_S messages not requiring acknowledgement

Certain ALARM\_S messages are configured as alarm messages but do not require explicit acknowledgement by the user; the CPU acknowledges such a message automatically when it arrives.

#### Message overload on the operating unit

In the event of a communication bottleneck in the CPU (multiple arrival and departure of an ALARM\_S message could not be acquired), a message might be displayed as *not acknowledged* on the operating unit even though it has been acknowledged in the CPU already.

The problem occurs only if simultaneously with a message

- A communication bottleneck occurs in the CPU
- The message in the CPU arrives
- The message is acknowledged by the operating unit

The operator must then acknowledge the messages several times on the operating unit.

### 5.9.13.10 Printing ALARM\_S messages

#### Principle

You specify which messages are to be output to a printer in the usual way in ProTool by choosing "System" > "Messages" > "Settings" from the menu.

#### Message Logging

If you do not perform device-specific message settings in SIMATIC STEP 7, all message events will always be logged directly on the attached printer.

If you configure device-specific messages, you can specify separately for each message whether it is to be logged or not.

Messages are always printed in the order in which they are sent by the CPUs. This is also the case with every synchronization.

---

#### Note

If there are several CPUs in a network, this means that messages need not always be received in chronological order by the operating unit.

---

### 5.9.14 Examples of configuring messages

These examples show you:

- How to configure an alarm message.
- How to create message areas for different devices.
- How system messages might appear, depending on the operating unit.

---

**Note**

The examples essentially describe how to configure bit messages. The configuration of ALARM\_S messages is described in the SIMATIC STEP 7 and SIMOTION SCOUT documentation.

---

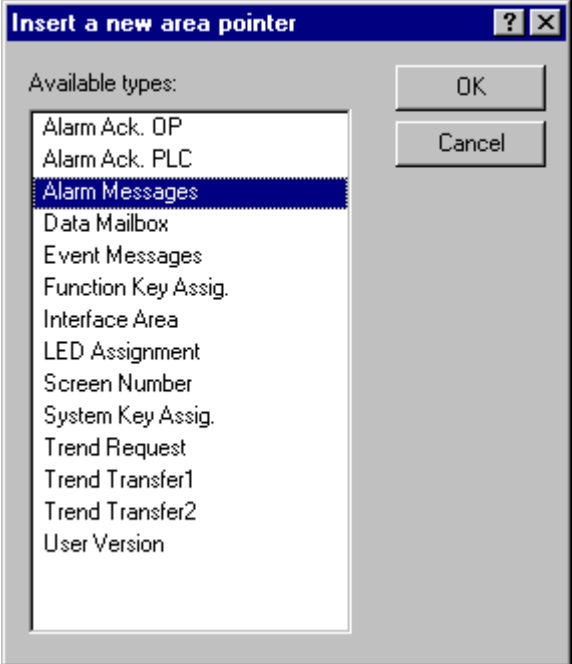
### 5.9.14.1 How to configure alarm messages

#### Objective

You want to set the alarm message area and configure an alarm message with a SIMATIC 300 or SIMATIC 400 PLC.

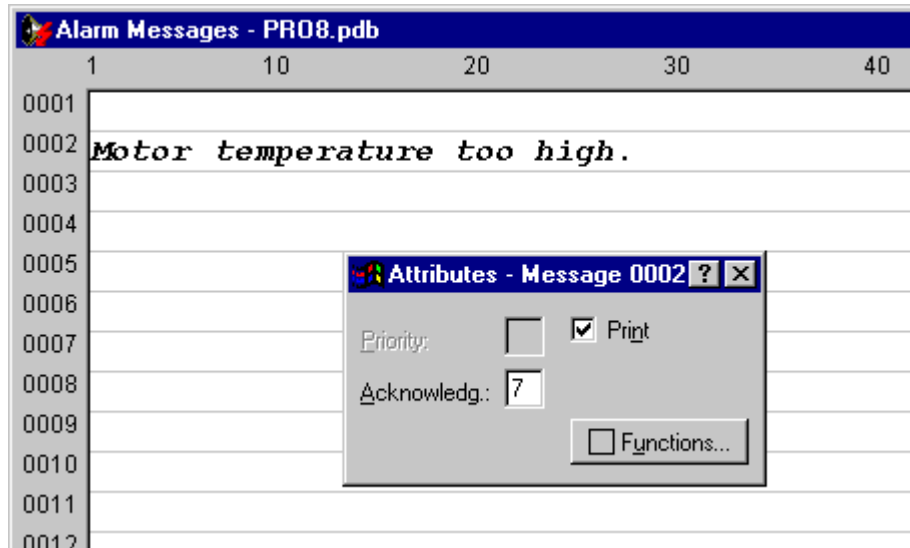
In this example, you will configure the alarm message area first and then an alarm message.

#### Procedure:

Procedure:	
1	<p>Left double-click in the project window on "Area pointers" to create the alarm message area. The "Area Pointers" dialog box is opened.</p> 
2	Here you select the "Alarm Messages" area pointer. Click OK.
3	<p>The "Alarm Messages" dialog box is opened. Enter the following:</p> <p>DB: 10            DW: 2            Length: 2            PLC: PLC_1</p> <p>This means you have created an alarm message area for 32 alarm messages.</p>
4	Confirm all settings by clicking OK.
5	Double-click the "Alarm Messages" window to open it. Position the cursor at message No. 2.

Procedure:	
6	Enter the following message: <i>Motor temperature too high.</i>
7	Perform the following settings in the "Attributes" window: Acknowledgement: 7 Print: <input checked="" type="checkbox"/>

The figure shows the configured alarm message:



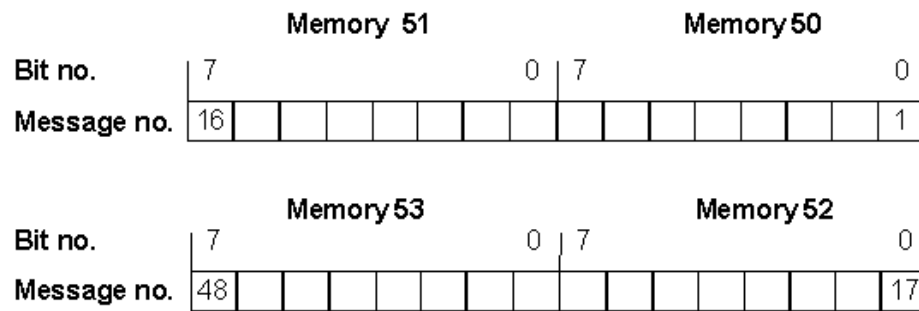
### 5.9.14.2 Message areas for SIMATIC S5

#### Memory on a SIMATIC S5

The message area is defined in memory words - for example:

FW 50

Length 2

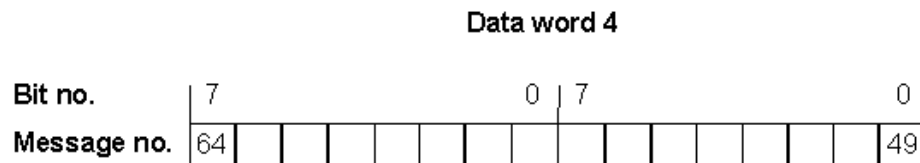
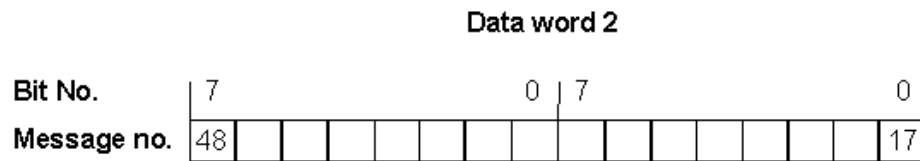
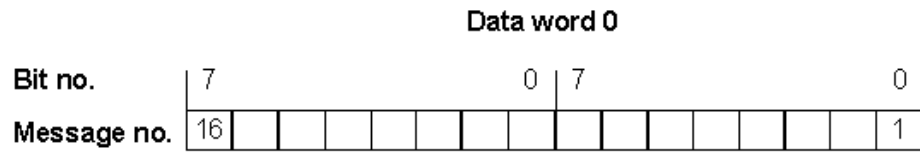


### 5.9.14.3 Message areas for SIMATIC S7

#### Data words on a SIMATIC S7

The message area is defined in a data block - for example:

DB 70  
DBW 0  
Length 4



#### 5.9.14.4 Examples of system messages

##### SIMATIC OP

The HMI system messages of the operating unit are read out, for example:

13001 Not enough memory available on hard disk

##### SIMATIC S7 (Minimal)

The message numbers of the S7 system messages are output. The operator can refer to the cause of

MESSAGE S4303 07.03.98 15:16:34

##### SIMATIC S7 (Standard)

In addition to the minimal display, the message text of the S7 system message is output. For this, the operator can select the corresponding message with the help of the arrow key.

MESSAGE S4303 07.03.98 15:16:34

If the RIGHT arrow key is operated:

PLC\_1:  
STOP by operating Stop switch

##### SIMATIC diagnostic buffer

Message text of diagnostic events are read out with the appropriately configured message view.

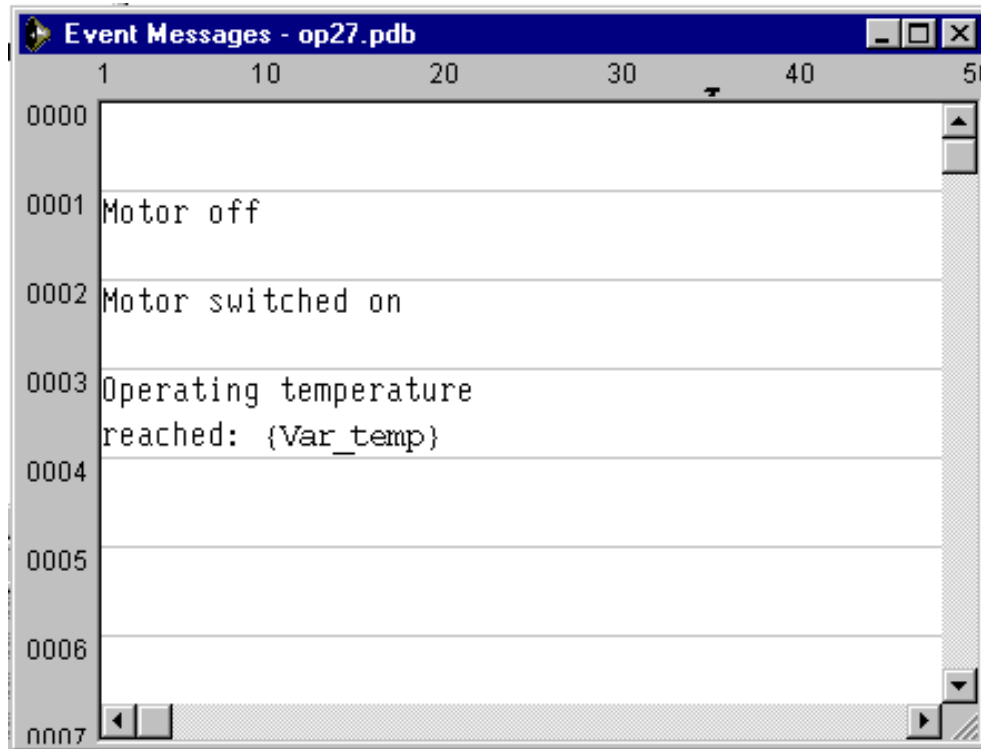
No	Time	Date	To ...	Text
12354	09:33:47	12.06.98	K	Module
12484	09:34:18	12.06.98	K	Distributed I/O: Station
16387	09:36:24	12.06.98	K	STOP by operating Stop switch
20493	09:36:25	12.06.98	K	New startup information in operating state STOP
4225	09:36:28	12.06.98	K	Manual restart request



### 5.9.14.5 Event messages export file

#### Structure

Below you can see how the export file of the illustrated event messages is structured:



SIMATIC ProTool/Pro B5.0.0.20 d 16.06.98 11:31:20

Legend:

:N: = Number

:T: = Text

:I: = Help information

:A: = Attributes

:V: = Tag

:L: = Text lists

Pieces of text may be distributed over several ':T:' lines,

Help information may be distributed over several ':I:' lines.

In message text (:T:), the following control characters are possible:

'\n' = New line

'\b' = Flash (Start/End)

'\u' = Underscore (Start/End)

'#' or '&' = Placeholders for message tags

The following identifications are used in the attribute line:

Px = Priority x

Ax = Output (port number) x

R+- = Relay ON (+) or OFF(-)

D+- = Print (+) or do not print (-)

:N: 0001

:T: "Tank1 empty:####"

:A: P1 A0 D+ R-

:V: "Z\_BAUDR" Dd TO

...

:N: 0005

:T: "Mixer full, \n"

:T: "Valves of Tanks 1 to 3 closed"

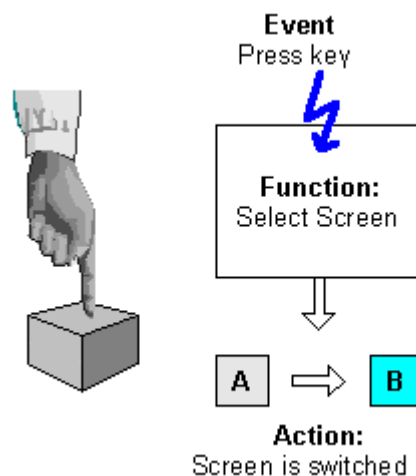
:A: P1 A0 D- R-

## 5.10 Using functions

### Basic principle

In ProTool you can link "events" (such as "Key pressed") with predefined functions. If the event occurs during operation, the function executes a specific action on the operating unit or the PLC.

For example, the "Select Screen Permanent" function on the operating unit opens a particular project screen.



### Triggering a function

### Areas of application

In general, you can use functions to:

- Set up the project for specific processes (for example, to switch from one screen to another)
- Control the process (for example, to set a bit in order to switch on a motor with it)
- Use features of the operating unit (for example, to display or print out the message buffer)
- Perform system settings online on the operating unit (for example, change modes)

In most cases you can configure the behavior of the functions precisely by means of parameters. If you wish to initiate several actions, you can also combine different functions.

**Configuration**

You configure functions on the "Functions" tab in the "Properties" dialog box of the object in question.

In addition, you can define global entry points (for example, for change of screens, buffer overflow and change of password) for many operating units by choosing "System" > "Functions" from the menu.

## 5.10.1 Events for triggering functions

### Necessity

The execution of a function is always linked to a specific event. The function is only triggered when this event occurs.

The events that can be linked to a function depend on the type of the function. Many functions are only effective with certain specific events.

### Examples of events

Examples of events that can trigger functions are the events "press key" and "release key". In the case of the former, the function is executed the moment a particular function key is pressed; in the latter case, it is executed the moment the function key is released again.

### Object-linked functions

Functions and events are generally linked to a specific object. The following objects can be linked to functions, for example:

- Function keys, soft keys and buttons
- Screen objects
- Screens
- Tags

Functions with tags are triggered only if the tag is displayed on the screen (default) or "Read Continuously" is selected on the "Options" tab in the Properties dialog box for the tag.

A function which you have configured for an internal tag (without PLC connection) whose value is modified by another function is not executed. Functions are executed only as a result of changes of values triggered by the PLC.

---

### Note

Depending on the operating unit, it might not be possible to configure functions for all these objects.

---

**Global functions**

You can also link some functions to global, object-independent events - for example:

- Changing screen
- Buffer overflow
- Change of password

The configurable global events depend on the operating unit:

You will find a detailed description of all the permissible events in online Help under the keyword "Configurable events".

## 5.10.2 Function parameters

### Necessity

Many functions can only execute one particular action. These functions are unique on account of their names alone. The `Events - Delete_Buffer` function empties the buffer for event messages, for example.



#### *Function without input parameters*

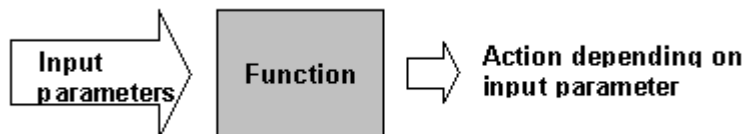
However, the situation is not always this clear. Many functions can work in several ways.

### Input parameters

Imagine that you want to open another screen using a key on the operating unit. To do this, you configure the `Select Screen Permanent` function. However, your project will generally have several screens. Which of these screens should ProTool open?

You therefore have to give the function more information. This is done using parameters. For the `Select Screen Permanent` function, you specify as a parameter the name of the screen to be opened, for example.

Another example is the function. `Switch Language`. In this case, the parameter you specify is the language to be set.

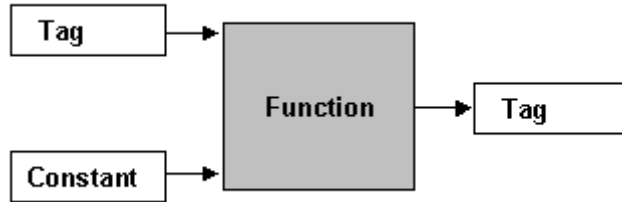


#### *Function with input parameters*

Some functions require a single parameter; others require several.

**General principle**

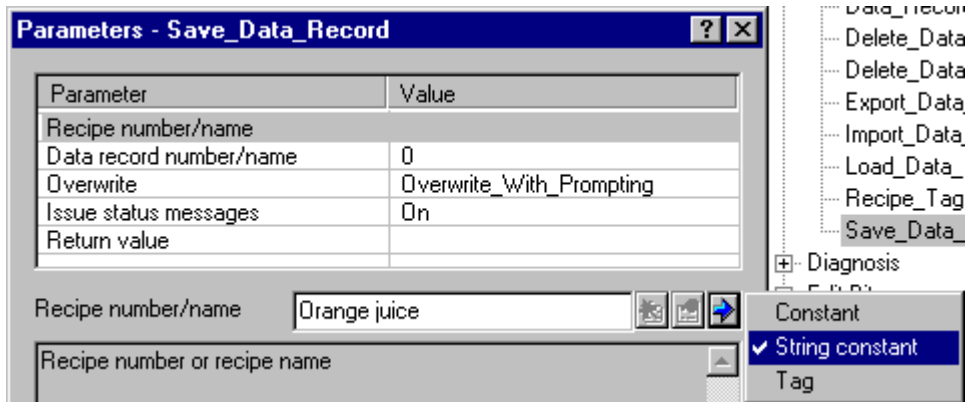
The following figure illustrates the principle of how a function works:



*How a function works*

An input parameter is specified for the function. This can be either constant or read from a tag. The tag may be local or it may have a connection to the PLC. If there is a connection to the PLC, the value is set by the process. The result of the function is written to a tag, which itself can either be local or have a connection to the PLC.

Since an input parameter can contain letters in addition to figures (for example, in recipe names), this has to be reported to the function. This is done by selecting a "string constant":

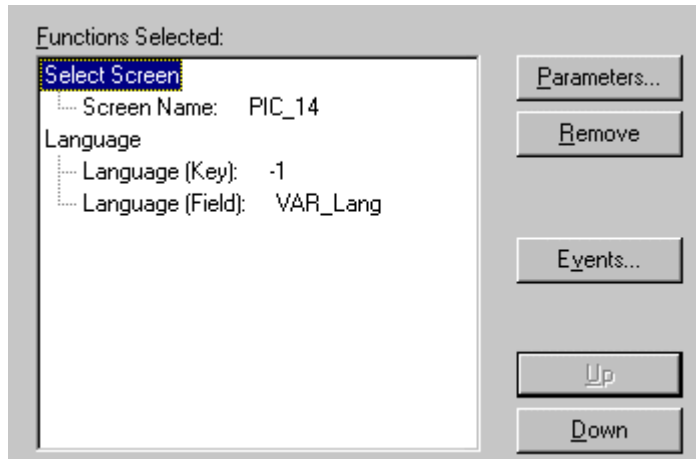




### 5.10.3 Combining multiple functions

#### Multiple functions with one object

You can also assign an event several functions. The functions are then initiated one after the other. You set the order in which this occurs in the "Functions" dialog box during configuration.



*Multiple functions with one object*

#### Sequence

You can change the order of the functions by using the "Up" and "Down" buttons. The list of functions is processed from the top down.

---

#### Note

On account of the differing runtimes of the functions, the start of the next function in the list is not linked to the completion of the function preceding it in the list. Thus, it can happen that several functions are being executed concurrently and the result of a previous function is not yet available.

---

## 5.10.4 User-defined functions

### Special tasks

If you want to execute an action or calculation for which there is no predefined function in ProTool, you can use VisualBasic Script to define your own functions and then integrate them and use them just like ProTool functions.

You will find additional information on VisualBasic Script in "Creating VBScripts".

### Function blocks

You can also use scripts to create readily reusable function blocks that you can call several times in your project without having to reconfigure all the functions each time.

## 5.10.5 Special features with conversion functions

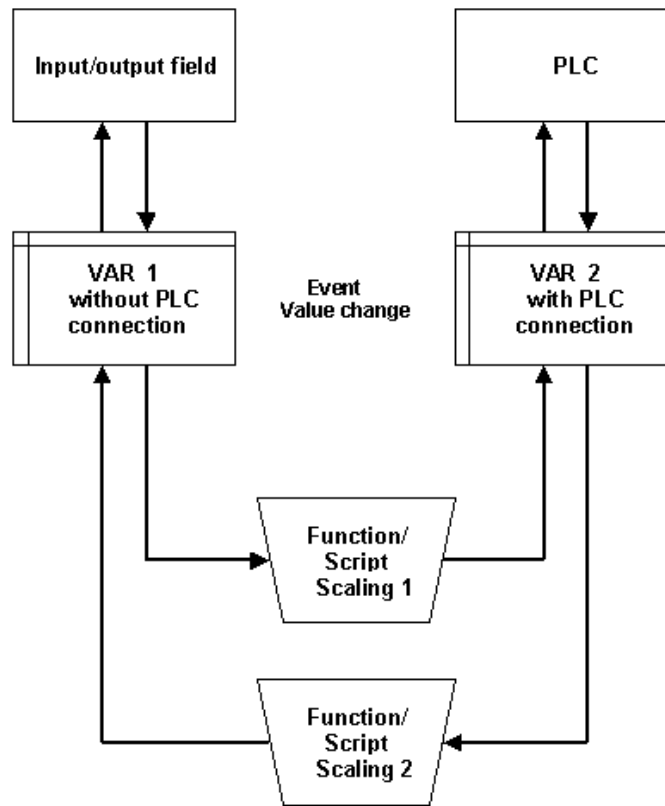
### Conversion possibilities

In Windows systems you have two alternatives for converting values between the PLC and the operating unit:

- For one tag, you can configure a simple linear conversion between different scale ranges in the "Tag" dialog box on the "Conversion" tab.
- An example of this would be conversion between two units of measurement, such as degrees centigrade and degrees Fahrenheit.
- For all other kinds of linear conversion with several tags, use the `Scaling_Linear_1` and `Scaling_Linear_2` functions or scripts.

### Peculiarities with functions and scripts

Conversions that are performed automatically are configured for the "Value Change" event of a tag. Since, in a reciprocal conversion, a distinction must be drawn depending on whether the value of the tag changes when it originates from the PLC or from the operating unit, an auxiliary tag has to be used. The figure below outlines what happens:



*Reciprocal conversion*

When an entry is made in the input/output field, the following takes place:

1. The value of the VAR\_1 tag changes.
2. The function or script for converting from VAR\_1 to VAR\_2 is triggered by the "value change" event.
3. Since VAR\_2 has a connection to the PLC, the changed value is available on the PLC.

When a value changes on the PLC, the calculation of the reverse function takes place:

1. Since VAR\_2 has a connection to the PLC, the value of VAR\_2 changes.
2. The function or script for converting from VAR\_2 to VAR\_1 is triggered by the "value change" event.
3. The changed value of VAR\_1 appears in the input/output field on the operating unit.

---

**Note**

In order to avoid any recursions, functions are not executed at the "value change" event when the value of the tag changes as a result of another function. The change to the VAR\_2 tag thus does not cause an immediate change to VAR\_1, which would result in an endless loop.

---

### 5.10.6 Example of configuring a function

This example shows you how to configure the "Change Mode" function with parameters so that "Online" and "Offline" modes can be set with function keys. In addition, the result of the operation will appear on the operating unit.

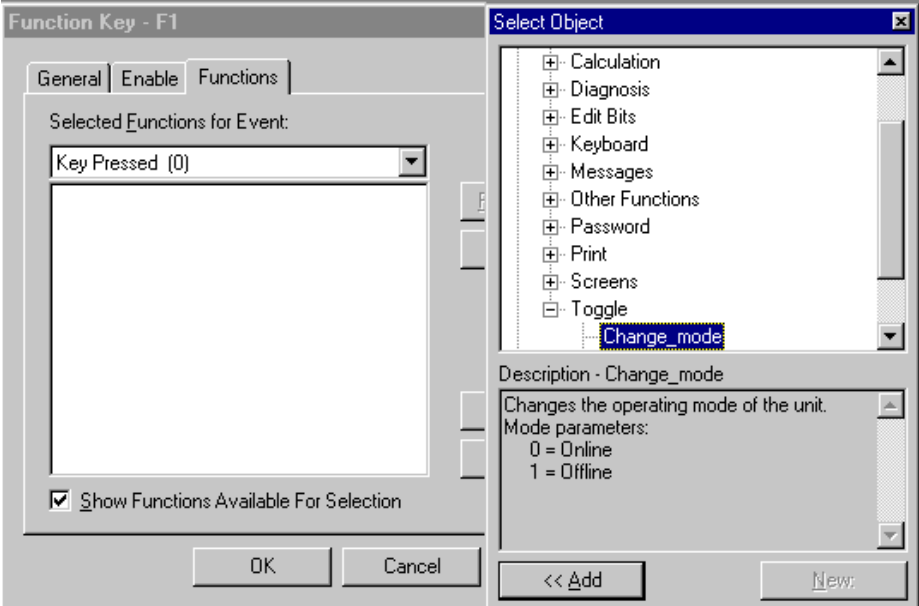
### 5.10.6.1 How to change the operating mode with a current display

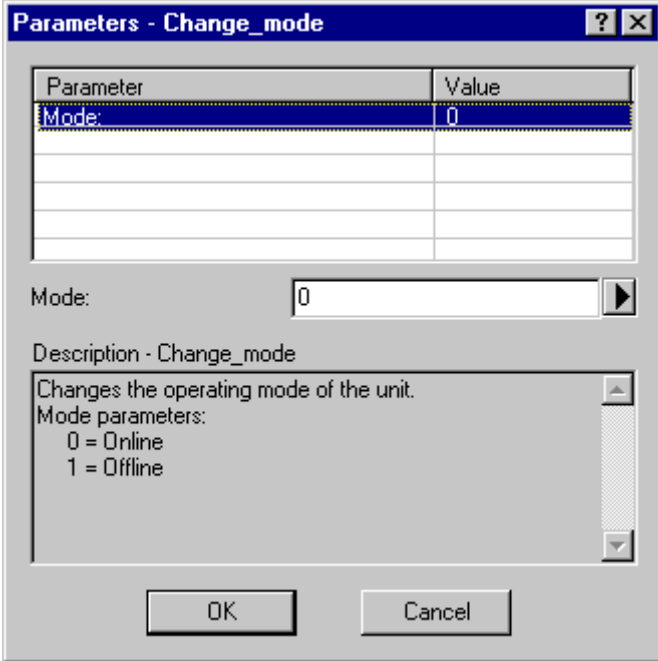
#### Objective

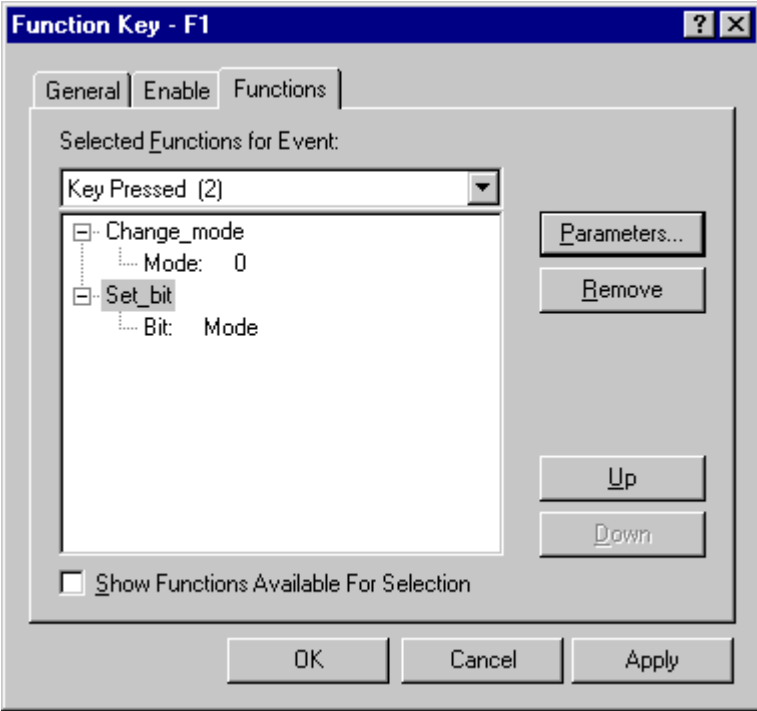
The following example using the `Change Mode` function illustrates how you configure functions with parameters. You also see how the result appears on the operating unit.

You want to set the `Online` and `Offline` operating modes by means of two function keys. You want the text `Online Mode` and `Offline Modes` to appear on the operating unit.

#### Configure the functions

Procedure:	
1	Create and open the screen in which the switchover and display take place.
2	Click the function key that you want to use to enable the <code>Online</code> operating mode. The "Function Key" dialog box appears.
3	Select the "Functions" tab.
4	At "Selected Functions", select the event "Press Key"
5	Select the "Display Selectable Functions" check box to open the "Select Object" window.
6	Select the <code>Change Mode</code> function in the "Toggle" function group. 
7	Click the "Add" button. The "Parameters" dialog box appears.

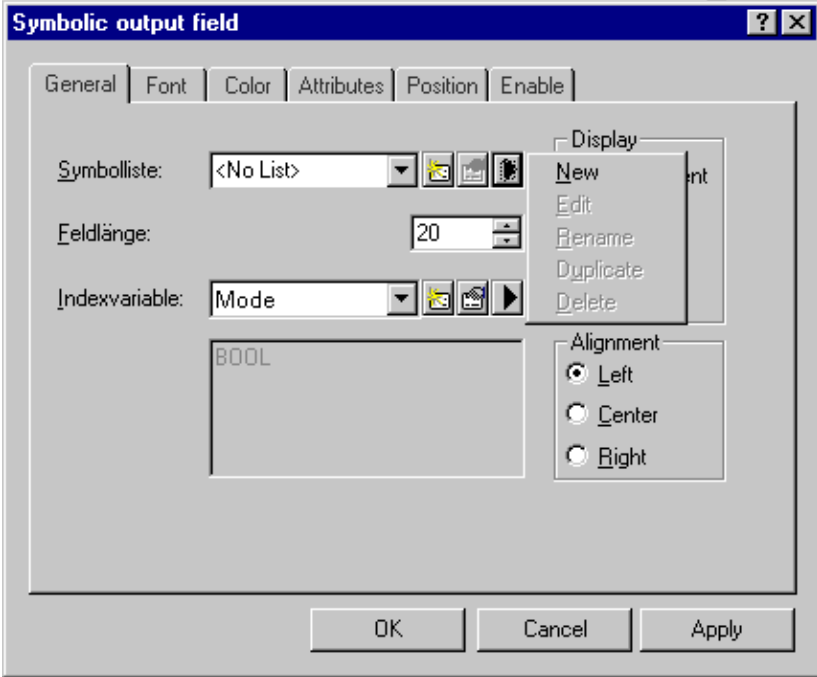
<b>Procedure:</b>	
8	<p>Select the <code>Operating Mode</code> parameter from the list, and enter the value 0 in the input field under the list. This parameter value controls the behavior of the "Change Mode" function, so that the mode changes to Online mode.</p> 
9	<p>Click OK to close the "Parameters" dialog box. The function then appears in the "Selected Functions" list.</p>

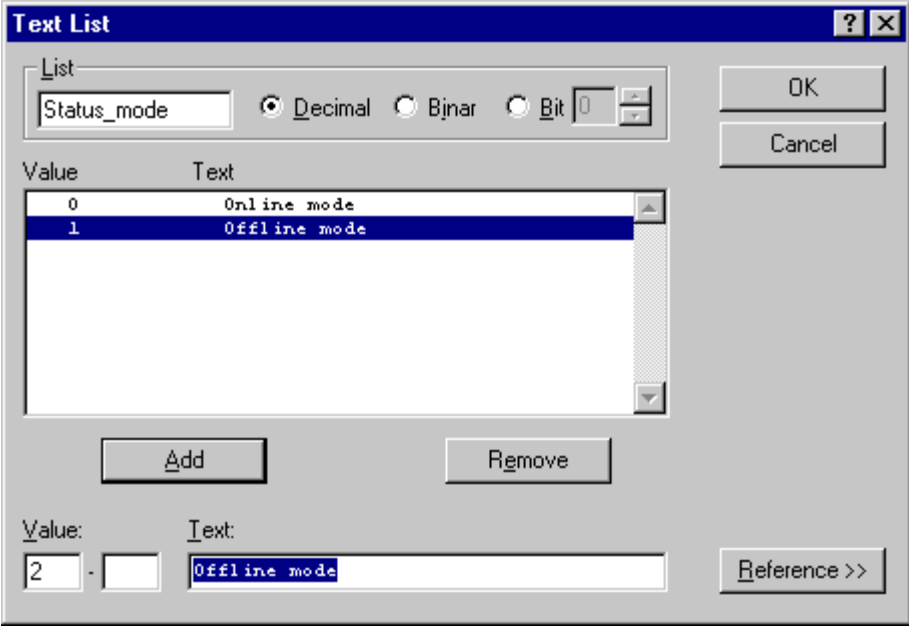
<b>Procedure:</b>	
10	<p>Configure <code>Set_Bit</code> as the second function. Specify a local tag of the Bool type as the parameter. You can create the tag during value assignment by clicking the arrow pointing to the right under the parameter list in the "Parameters" dialog box. Configure the tag with the following data:</p> <p>Name: <code>Operating mode</code>                      PLC: <code>&lt;No PLC&gt;</code>                      Type: <code>BOOL</code></p> <p>Both functions, including the configured parameters, appear in the list in the "Function Key" dialog box.</p>  <p>Close the dialog box by clicking OK.</p>
11	<p>Right-click the function key and select "Copy". This copies the properties of the function key you have just configured.</p>
12	<p>Using the shortcut menu, paste the copied function key properties to the second function key.</p>
13	<p>Click the second function key and change the value to 1 in the Parameters dialog box of the "Operating mode" parameter. Replace the function <code>Set_Bit</code> with the function <code>Reset_Bit</code> with the tag <code>Operating Mode</code>.</p>

In the next step you create an output field whose contents change dynamically depending on the `Operating mode` tag. The user can thus read on the operating unit at any time which operating mode is currently set.



### Configure the display

Procedure:	
1	Insert a symbolic output field in the screen. The "Symbolic Output Field" dialog box appears.
2	On the "General" tab at "Display", choose at "Index tag" the previously created <code>operating mode</code> tag.
3	Click the arrow pointing to the right next to "Text or Graphic List", and choose the "New" menu command from the shortcut menu that appears.  <p>The "Text or Graphic List" dialog box appears.</p>

Procedure:	
4	<p>Call the Text or graphic list <i>status</i> and create the list with the data:</p> <p>Type: Bit 0                      Value: 0 Text: Online mode                      Value: 1 Text: Offline mode</p> 
5	Click OK to accept all the dialog box entries.

**Result**

At run time, the operator can use the two configured function keys to set the operating mode and read the current operating mode on the display.

## 5.11 Create archives

Windows-based systems offer you the opportunity to archive process data (in other words, store them permanently and analyze them).

The following types of process data can be archived:

- **Tags**In the "Tags" dialog box you define a trigger condition and the range of values for archiving the tag.
- **Messages** By choosing "System" > "Messages" > "Settings" you can define which messages will be archived.
- **Trends**Use the "Data Source" tab in the "Trend" dialog box to set the archive from which the tags for displaying the trend will be read.

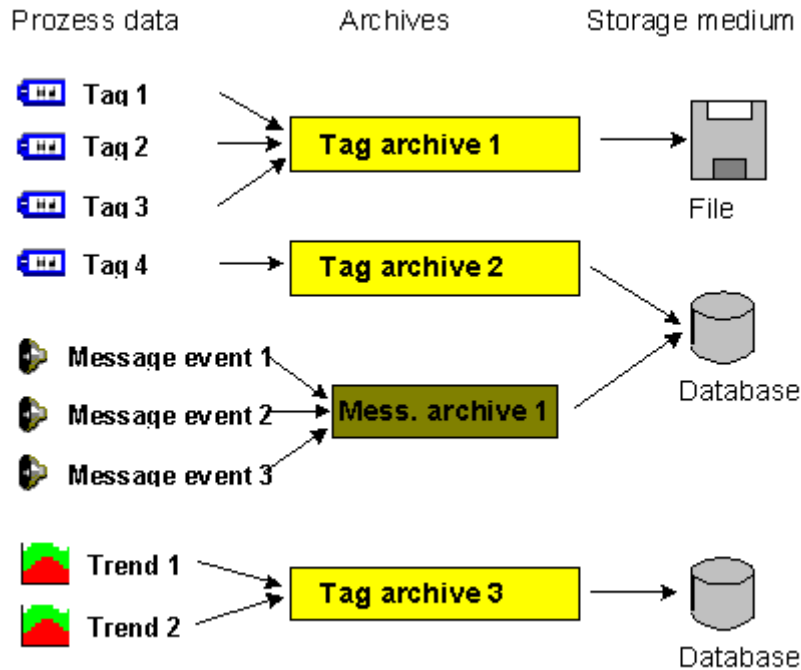
### 5.11.1 Archiving process data

#### Archive properties

The properties of an archive, such as its storage location, etc., are defined in the "Archive" dialog box. The "Archive" dialog box appears when a new archive is created in the project window or an existing archive is opened.

#### Archiving model

The following diagram shows the archiving model:



*Archiving model*

#### Storage media

The data can be archived on different storage media:

- File
- Database (not for Multi Panel)

### Storage locations for archives

Depending on the hardware configuration of the operating unit, you can configure the following storage locations in ProTool CS:

- \Storage Card\... (PC-Card)
- \Storage Card2\... (CF-Card)
- Network path (when a network is created on an operating unit)

---

#### Note

When creating archives on the CF card, make sure it is inserted. If not, the data will be stored in the RAM file system of Windows CE and are lost on power down.

---

### Storing data to file

Each archive is stored as a separate file. You must specify a path at "Storage location".

If you are creating an archive but you either enter nothing at "Path Name" on the "Settings" tab or you specify only a drive, such as C:\, an archive will not be created.

---

#### Note

To the extent possible, avoid using spaces in the path name. If you cannot avoid them, place path names containing spaces in quotation marks.

---

When you archive process data to a floppy disk, archiving is terminated without checkback when the floppy disk is full.

The data are stored in the file in compliance with the Microsoft CSV standard, that is in a text file that has entries containing freely definable separators. You will find the "list separator" set on your system on the "Start" menu at "Settings" > "Control Panel" > "Regional Settings" on the tab labeled "Numbers".

All data are stored as ANSI characters. CSV files can be read in and processed by other programs for the purpose of evaluating the data.

---

#### Note

In the case of Windows CE devices, the internal FLASH memory cannot be specified as a storage location.

---

---

**Note**

The following applies to Windows CE devices:

In ProTool 5.2 and higher, the regional options such as format of the date and time and decimal and list separators are "German" by default.

Modify this setting or replace the separators in the file, as necessary, if you are using archives or exported recipe data records in ProTool 5.2 or higher that you created in ProTool 5.1. The reason for this is that the operating unit can read archives and exported recipe data records correctly only if the set decimal and list separators are identical at the time of reading and writing.

When you change the system language to "English", modify the time output simultaneously to the 24 hour format, since the standard width of the time field in the message view is geared to the 24 hour format.

---

**Data storage in database (except Windows CE devices)**

The database is addressed by its "data source name". Select the database you want to use in ProTool from the "Start" menu at "Settings" > "Control Panel" > "ODBC Data Sources".

This means that you then have the entire functionality of the database at your disposal for further processing and evaluating your archive data. Basically, any database featuring ODBC interface can be used.

The following databases have been approved:

- MS Data Engine 97 and 2000
- MS Access 97 and 2000
- MS SQL Server 7.0 and 2000

To enable archive data to be saved to a database, a feature known as a data source name (DSN) must be specified during configuring instead of a directory name. The DSN is used to provide a precise reference to the database and its location.

The data source that establishes the connection to the database must be created on the same computer as the Runtime software is stored. The DSN configured there is then specified in ProTool during creation of the archive.

**Archive response in Runtime**

If the archives are activated after a cold restart (power OFF/ON) of the Panel before the network connection has been established, an initialization error is reported.

Therefore, configure in your ProTool project an operating option whereby you can manually open the archives in Runtime, such as a button for the function "Open\_Archives".

## 5.11.2 Settings for an archive

### Settings

When creating an archive, you can choose the response for a full archive in the Properties dialog box on the "Settings" tab:

- FIFO buffer
- Creating sequence archives

### Short-term archive

A "short-term archive" is a FIFO buffer, which means that if the buffer has a capacity of 100, for example, only the last 100 values are archived. Older values are overwritten.

---

#### Note

On performance grounds it will always be possible to read at least 80% of the values from the buffer. Therefore if you want to have a certain number of values available (say 100 values), you should be sure to create a FIFO archive which is larger (say 125 values).

---

### Sequence archive

A "sequence archive" is filled to the defined size. The data medium or archive then has to be changed in order to continue archiving.

You can select one of the following options for sequence archives:

- "Automatically Create Sequence Archive" A new archive is created automatically. The archives receive the specified archive name with an additional number (1 ... n). You can configure the number of archives. Once the last archive has been filled, the first archive is filled again.
- "Output System Message When" If the archive (a floppy disk, for example) is full, a system message is issued. You can specify the level at which the message is to be output as a percentage.
- "Trigger Function" If the archive is full, functions are triggered which allow special sequential archive handling.

### 5.11.3 Displaying archive data in a specified time window

#### Principle

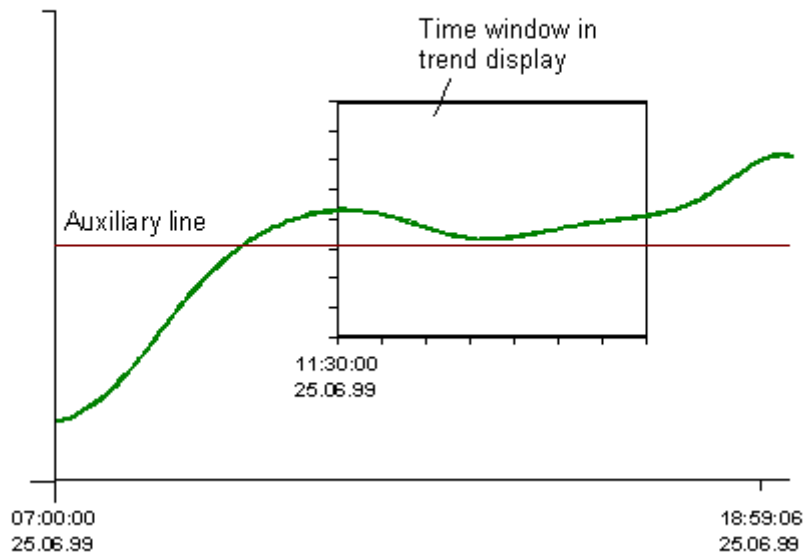
You can display archive data (in other words, tags and messages) on the operating unit, in the form of trends (for tags) or a message display (for messages).

If you want to quickly and easily display archive data within a particular time slot, you can enter the required date and time in an input field.

- In the "message view", the selected messages are displayed over a range before and after the set point of time.
- In a "trend display", create a time slot by means of a trend display on the screen and display the archived trend data up to the set point of time.

The following figure shows an example of a trend display:

**25.06.99 11:30:00**



*Example of a trend display and input field*



## 5.11.4 Options for configuring archives

### What can an archive contain?

Windows-based systems offer you the opportunity to archive process data (in other words, store them permanently and analyze them).

Depending on the process data saved, there are different types of archive:

- **Message archives**A message archive is used to store the message events of one or more message classes (event messages/alarm messages/system messages). You are only able, however, to assign one specific archive to each message class. Since login and logout operations are automatically output as system messages, you can also archive all logins and logouts with the message archive.
- **Tag archives**A tag archive contains the values of one or more tags at specific points in time. You are only able, however, to assign one specific archive to each tag.

The properties of an archive, such as its storage location, etc., are defined in the "Archives" dialog box. You will find an archiving model for a project in "Archiving process data".

### How do you archive process data?

Usually, messages and tags are archived over the entire duration of the process. If an archive is full, either the sequence archives configured for it are filled or the archive is written again ("FIFO Buffer" option).

With the aid of the ProTool functions, you are able to customize the way in which your project is archived. Application examples for this include:

- The operator can perform unscheduled archiving of messages and tags.
- The operator can change the archive at any time he wants.
- Additional functions are executed when a sequence archive starts.
- A message archive or a tag archive is displayed on the operating unit.
- The archive data are stored in a database and processed further.
- The archive file is copied and processed further in another program.

## 5.11.5 Examples of using archives

These examples illustrate how to:

- Create and structure an archive for messages.
- Structure an archive for tags.
- Display archive data with a time window.

### 5.11.5.1 How to create an archive for messages

#### Objective

You want to create archives for messages each having a size of 300 entries.

#### Procedure:

Procedure:	
1	Double-click the "Archives" object type in the project window to open the "Archive" dialog box.
2	On the "General" tab, select "Messages".
3	At "Behavior at Restart", select "Reset Archive". This means that in the event of a restart, the archive will be written from the beginning again.
4	Select "file" as the "storage location" and specify the path on the "Settings" tab.
5	For "Size", enter 300 entries.
6	At "Behavior When Archive Full", select "Automatically Create Sequence Archive" and enter 7 as the number (in other words, 8 archives will be created). When these have all been filled, the first one is then overwritten.

### 5.11.5.2 Structure of an archive for messages

#### Structure

Here is an example of a file for messages:

```
"Time_ms";"MsgProc";"StateAfter";"MsgClass";"MsgNumber";"Var1";...;"Var8";"TimeString";"MsgText";"PLC"
```

```
36341550590,27;1;1;3;110001;"";...;""; "06.30.99 13:12:51";"Change to 'Online' mode";
```

```
36341550682,87;1;1;3;140010;"";...;""; "06.30.99 13:12:59";"Connection established: PLC_1, Station 2, Rack 0, Slot 2";
```

```
36341550682,87;1;1;3;140001;"2";...;""; "06.30.99 13:12:59";"User administrator logged in with level 9";
```

```
36341550682,87;1;1;3;170003;"PLC_1.";...;""; "06.30.99 13:12:59";"Change to 'Online' mode";
```

```
36341601759,25;1;1;3;190011;"";...;""; "06.30.99 14:26:32";"Connection established: PLC_1, Station 2, Rack 0, Slot 2";
```

The following is entered in the various columns:

Time_ms	Time stamp as decimal value <sup>*)</sup> Converted
Msg_Proc	Message procedure: 0 = unknown message procedure 1 = system message 2 = message bit procedure (event messages) 3 = ALARM_S message number procedure 4 = diagnostic event 100 = message bit procedure (alarm messages)
StateAfter	Message event: 0 = active/cleared 1 = active 2 = active/acknowledged/cleared 3 = active/acknowledged 4 = active/cleared/acknowledged
Msg_Class	Message class: 0 = no message class 1 = alarm message 2 = event message 3 = system message 4 = diagnostic event
MsgNumber	Message number
Var1 ... Var8	Value of the message tag as a string
TimeString	Time stamp as a string, that is, in a readable date format
MsgText	Message as readable STRING
PLC	Message localization (PLC affected)

<sup>\*)</sup> Note on converting values in the Time\_ms column:

If you wish to process a value in another program, it must be divided by 1 000 000.

Example: 36343476928: 1 000 000 = 36343.476928

A date from 31.12.1899 onward is derived by calculation from the value before the decimal point.

The time is derived from the value following the decimal point: 0.476928. Multiply by 24 to give the hours; multiply any remainder by 60 to give the minutes; again multiply any remainder by 60 to give the seconds; any further remainder multiplied by 1000 gives milliseconds.

Result: 07. 02.1999 11:26:46

### 5.11.5.3 Structure of an archive for tags

#### Structure

Here is an example of a file for tags:

```
"VarName";"TimeString";"VarValue";"Validity";"Time_ms"
"Var_107";"01.04.98 11:02:52";66,00;1;35886460322,81
"Var_108";"01.04.98 11:02:55";60,00;1;35886460358,73
"Var_109";"01.04.98 11:02:57";59,00;1;35886460381,22
```

The following is entered in the various columns:

VarName	Name of the ProTool tag
TimeString	Time stamp as a string, that is, in a readable date format
VarValue	Value of the tag
Validity	Validity: 1 = value is valid 0 = error has occurred
Time_ms	Time stamp as decimal value <sup>*)</sup> Converted: required only for displaying the tag values as a trend.

<sup>\*)</sup> Note on converting values in the `Time_ms` column:

If you wish to process a value in another program, it must be divided by 1 000 000.

Example: 36343476928: 1 000 000 = 36343.476928

A date from 31.12.1899 onward is derived by calculation from the value before the decimal point.

The time is derived from the value following the decimal point: 0.476928. Multiply by 24 to give the hours; multiply any remainder by 60 to give the minutes; again multiply any remainder by 60 to give the seconds; any further remainder multiplied by 1000 gives milliseconds.

Result: 07. 02.1999 11:26:46

### 5.11.5.4 How to display archive data with a time window

#### Objective

On the operating unit you wish to display a trend showing the archive data from a certain point in time onward. This means that the operator will be able to access the particular time slot quickly and easily, and without having to spend a great deal of time scrolling on the screen.

#### Configuring an input field for an archive trend

Procedure:	
1	Configure a Date/Time field for entering the start time for displaying the archive data:
2	Select a tag without PLC connection as the "data source" in Date/Time format: Name: VAR_15 PLC: <No PLC> Type: DATETIME
3	In the trend display, go to the "Axis Scaling" tab and select the tag VAR_15 .

#### Result

When the operator enters a date and a time on the operating unit in this input field, the archive data for the specified moment in time onward will be displayed.

---

#### Note

For a message display, go to the "General" tab and select a tag with the starting time.

---

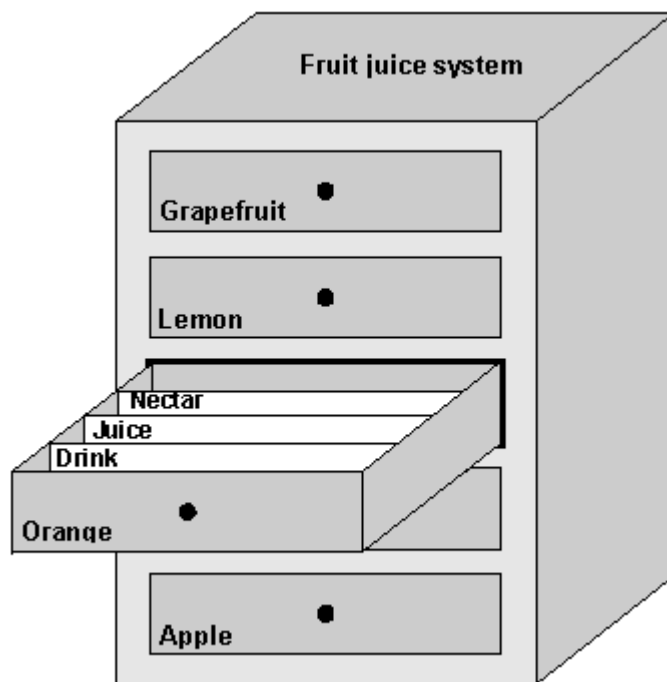
## 5.12 Creating recipes

### Purpose

The purpose of recipes is to be able to transfer collectively and synchronously multiple items of data that belong together from the operating unit to the PLC and vice versa.

### Principle

The terms "recipe" and "data record" should be defined using the example of a filing cabinet, as they are important for later understanding:



*Recipe and data record illustrated by analogy with a filing cabinet*

- Recipe  
Recipes correspond to the drawers of the filing cabinet shown (for example, Orange or Lemon). The reference value fields (tags) belonging to the particular recipe are defined in each drawer. You use the recipe to define the data structure in your ProTool project.  
You cannot change this structure subsequently on the operating unit.

- Data record  
 Data records correspond to the filing cards in the drawers of the cabinet (Drink, Juice and Nectar). A data record contains the values for a recipe.  
 The operator edits data records on the operating unit, for example, creating, saving and deleting data records.  
 The Import/Export function gives the operator the additional option of using external applications to edit exported data records and then importing them back on the operating unit.

**Example of a recipe**

An example of a recipe is the filling station of a fruit juice system. The filling station is used to produce orange drink, orange juice and orange nectar. The mixing proportions for each of these are different. The ingredients are always the same.

Let us assume that a recipe called "Mixture" is created, containing the following data structure:

Tag	Designation
Var_2	l orange
Var_3	l water
Var_4	kg sugar
Var_5	g flavor

The designations l orange, g flavor, etc. of the tags are what are referred to as the "entry names". The entry names are displayed as well on the operating unit. Thus the tag Var\_2, for example, is identifiable as the tag which defines the proportion of orange in the mixture.

The data records contain the values for the different drink types. The data records could be as follows, for example:

Mixture	Orange juice	Orange nectar	Orange drink
l orange	95	90	70
l water	5	10	30
kg sugar	3	10	45
g flavor	100	300	600



## 5.12.1 Using and configuring recipes

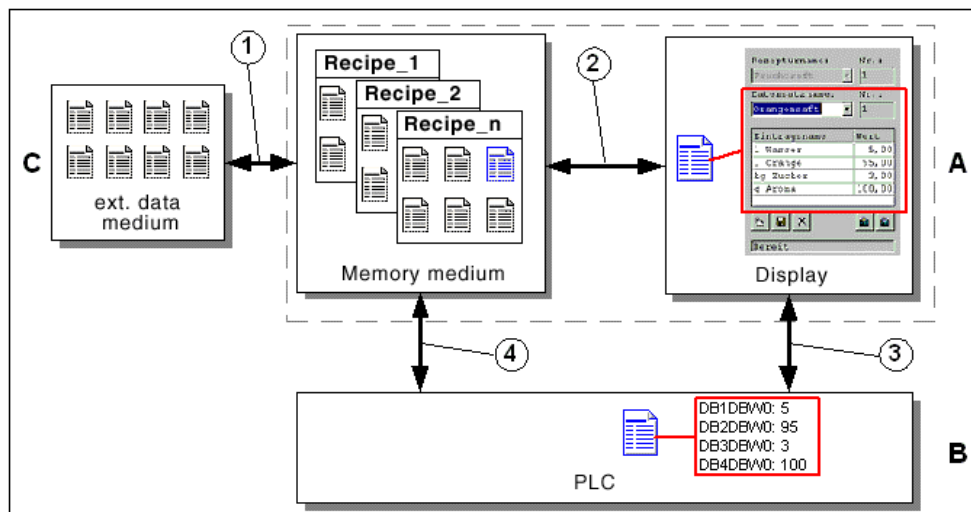
### Using recipes

With regard to using recipes, you will find three scenarios, among other things, at "Examples", which demonstrate the use of recipes with brief practical examples. They are intended to provide you with a top level overview of the possibilities open to you as a result of using recipes:

- The first scenario shows a recipe configuration with which entered data are not transferred immediately to the PLC. You set this configuration when, for example, you want to enter data on the operating unit without interrupting an active process.
- In the second scenario, data are immediately transferred to the PLC. This configuration can be used, for example, to run in a machine and correct positioning data online when doing so.
- The third scenario uses data record functions for automatically controlling production by means of a job list.

### Data flow with recipes

The manner in and the route used for transferring data records between the different components (A, B, and C) will depend on the recipe and the functions used. The figure shows the data flow that might occur when recipes are used:



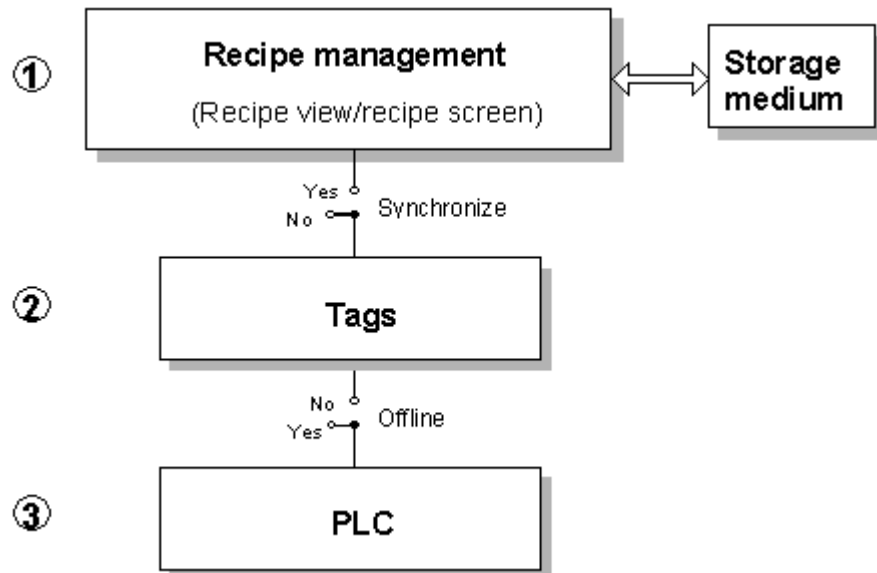
The operating unit (A) stores recipe data records on a memory medium, such as a FLASH memory or hard disk. You can then edit a recipe data record in a recipe view or a recipe screen on the operating unit display:

- You can enter a new data record, save it on the memory medium (2), for example, and transfer (3) it to the PLC (B).
- You can export the data record to an external data medium (C), to a PC, for example. The data record is stored on it independently of the recipe in \*.CSV format. You can load this file in MS Excel, for example, and continue editing it.
- Similarly, you can load an existing data record from the memory medium of the operating unit (2) or import it as a \*.CSV file from an external data medium (1).

You can also directly transfer a recipe data record from the memory medium to the PLC (4) or load it from the PLC to the storage medium. In the same way, you can transfer a recipe data record being shown on the display, to the PLC (3) or read out from it.

### Configuration options with recipes

In recipe properties, you can control the recipe tag properties on the "Options" tab. The figure shows the basic differences when working with data records and the effect of the options "Synchronize" and "Offline".



*The "Synchronize" and "Offline" options when editing data records*

#### 1. Recipe without "Synchronize Tags"

The data of a data record which has been read is only displayed in the recipe view table and can also only be edited in the recipe view. If you use the same tags outside the recipe view, this does not affect their values.

**2. Recipe with "Synchronize Tag" and with "Offline"**

With the "Synchronize Tags" option, you specify that the data of a data record which has been read from the PLC or storage medium has been written to the tags or has been read from the tags you configured for the recipe.

The "Offline" option ensures that the entered data are only saved in the tags, but not transferred directly to the PLC on being input.

**3. Recipe with "Synchronize Tag" and without "Offline"**

As 2. above). However, entered or read in data are immediately transferred to the PLC.

## 5.12.2 Recipe view

### Purpose

You want to be able to work straightforwardly and quickly with recipes and data records, with minimal configuration.

### Usage

The recipe view is a compact screen object which enables you to handle recipes and data records quickly and straightforwardly, with minimal configuration.

This method is particularly recommended for data records with relatively few entries, which are suitable for editing in tabular form.







You configure a recipe view on the open screen in the "Screen Objects" toolbar by selecting the button shown.




### Note

You can enable the recipe view by using SHIFT and double-click or from the "Edit" option of the shortcut menu. This means that you can move a column to the required position and set the column width during the configuration phase.

### Using a recipe view

ProTool provides the following buttons for using the recipe view at run time. You can set individually, for each of these buttons, whether it will be displayed on the operating unit.

Button	Purpose
 New	<p>"Create New Data Record"</p> <p>The operator uses this button to create a new data record on the operating unit. The specified values are assigned the default values specified as initial values for the tags concerned in ProTool.</p>
 Save	<p>"Save Data Record"</p> <p>The operator uses this button to save the current values of the data record located in the tags on the data medium. The storage location is defined when you configure the recipe in the recipe editor ("Properties" button, "Data Medium" tab).</p>
 Save As	<p>"Save Data Record As"</p> <p>As "Save Data Record". However, the operator has the option of changing the current data record name or selecting an existing data record name.</p>
 Delete	<p>"Delete Data Record "</p> <p>The operator uses this button to delete from the data medium the data record displayed on the operating unit for the currently set recipe.</p>

Button	Purpose
 Synchronize	"Synchronize Data Record" The operator uses this button to establish consistency between the recipe view and the tags. The modified values of the tags are written from the table to the appropriate tags. Following that, all values are read from the tags and updated in the table.
 To PLC	"Write Data Record to PLC" The operator uses this button to write the current data record from the tags to the PLC.
 From PLC	"Read Data Record from PLC" The operator uses this button to read the data record for the currently set recipe from the PLC and display the values on the operating unit.

---

#### Note

In ProTool 5.2, the data in the recipe view were used for the actions "Save " and "Download to PLC". These data are not always up-to-date, however. In ProTool 6.0, the data in the tags are used.

---

### Changing screens in recipe views

On changing screens, the recipe view retains its present contents. If the recipe view contains unsaved values at the time the screen is changed, you are prompted to save them. The recipe concerned and the data record are displayed for guidance. If you return to the recipe view, its contents are updated automatically.

### Tips on operation without a mouse and without a touch screen

You can also use all the recipe view controls without a mouse and without a touch screen. You will find the relevant key combinations in the "ProTool/Pro Runtime" user manual and in the equipment manual of your operating unit.

Take the following steps to optimize recipe view operation for your operating unit:

- **Function keys instead of buttons:** In ProTool, deselect the button display. Instead, assign the corresponding functions to function keys or soft keys. Select the "Functions" tab for the function key. In the "Select Object" window, ProTool provides all the data record editing functions under the heading "Keyboard operation of screen objects".
- **Scrolling through recipe entries:** For the "recipe view", configure two function keys or soft keys, to which you assign the functions `Page_Up` and `Page_Down`. If you have selected the table, you will then be able to use the keys at run time to scroll forward or backward through the recipe entries, a page at a time.

### Specifying the recipe name

If you select a specific recipe at "Recipe name fixed" (in the "Recipe View dialog box, on the "General" tab), you will only be able to display and edit data records of the recipe you specify in the selection field.

Possible applications:

- You only wish a specific recipe to be edited or displayed on a particular screen.
- For example, only the data records of one specific recipe are to be transferred for a user group.

### Displaying and/or editing data records

You set the "Display table" setting to specify whether the configured entries of the current data record are to be displayed on the operating unit. If you suppress the display, you simultaneously rule out the possibility of using the recipe view to create or edit data records.

You use the "Edit" setting to specify whether you can edit the table at run time and directly enter or modify the data record name. If you inhibit editing, the recipe view is used only as an output field.

If you have selected both the "Display table" and "Allow Edit" check boxes, the configured entries of the data records will be displayed on the operating unit. At the same time, you have write access to the values.

### Writing recipe and data record numbers/names to tags

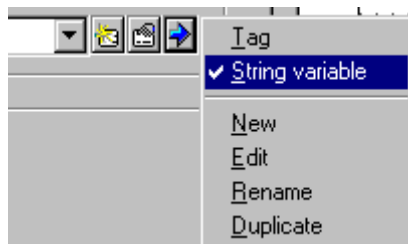
You can configure separate tags for the recipe and data record numbers/names. The numbers selected on the operating unit are written to them at run time. You can use the recipe and data record numbers/names stored in the tags as a parameter for functions and scripts to save the current data record, for example.

---

#### Note

Select "String tag" when you use recipe and data record names, and "tag" when you use recipe and data record numbers.

---



### 5.12.3 Recipe screen

#### Purpose

You want to configure custom screen forms for entering data records, for example, by visually simulating your system.

#### Usage

The recipe consists of screen objects, for which you configure recipe tags. You can distribute large recipes over several topic-specific screens and display them with graphic elements, for example.

This method is particularly recommended for data records with a relatively high number of entries. In addition, you have the option of linking several recipes in mutual dependence. You can configure a recipe view on a recipe screen in order to use it as a selection list for data records, for example.

#### Basic procedure for creating a recipe screen

1. Define the recipe structure:  
Assign tags to the structure of the recipe.  
Use these tags for the input/output fields in the recipe screens.  
Define a name for the recipe.  
You will use this name to select the recipe in your project and on the operating unit.  
  
Synchronize tags:  
In recipe properties, you can control the recipe tag properties on the "Options" tab. With the "Synchronize tags" option, you specify that the data of a data record which has been read from the PLC or data medium has been written to the tags or has been read from the tags you configured for the recipe.  
  
Tags offline:  
Also on the "Options" tab, use the "Tags offline" option to set (default case) that entered values are only saved in the tags, but not downloaded directly to the PLC on input. The requirement here is that the "Synchronize tags" option is also selected. If the "Tags offline" option is not set, the values are downloaded directly to the PLC on input.
2. Create the screen:  
Create one or more screens if you want to be able to create, save and transfer data records on the operating unit. Create input/output fields for the recipe tags. They can be used to display and change the values on the operating unit.
3. Assign functions:  
Assign the functions for editing data records (for example, saving or transferring) to function keys, soft keys or buttons. To do this, select the "Functions" tab. In the "Object Selection" window, ProTool provides all the data record editing functions under the heading "data record".

## 5.12.4 Configuring recipes

### Basic procedure

The basic steps to configuring a recipe are listed below:

1. Define the recipe structure:  
Assign tags to the structure of the recipe. Assign technological entry names to these tags. Define a name for the recipe. You will use this name to select the recipe in your project and on the operating unit.  
Set recipe tag properties:  
In the recipe properties, you can control the recipe tag properties on the "Options" tab. With the "Synchronize tags" option selected, you specify that the data of a data record which has been read from the PLC or data medium has been written to the tags or has been read from the tags you configured for the recipe. This establishes a connection between the tags configured in the recipe and the tags in the screens. When a data record is loaded, the values are written to the tags you are using on the screens.  
If the "Tags offline" option is also set, values you enter are only saved in the tags, but not downloaded to the PLC. If this option is not set, the values you enter are transferred directly to the PLC.
2. Set data record storage to the operating unit:  
You can configure the storage path for the data records. Check the set storage location if you copy recipes between configurations with different destination devices.

---

### Note

In the Recipe dialog box at "Properties" > "Archive", avoid specifying a path name that refers to a floppy disk drive, since the system is blocked during access to the floppy disk drive. Furthermore, non-defined types of behavior occur when a floppy disk is not inserted or when the floppy disk is removed during operation.

---

PC: Any path and drive except floppy disk drive  
Windows CE devices (except OP 170B and TP 170B):  
Depending on the hardware configuration of the operating unit, you can configure the following storage locations in ProTool CS:  
\Storage Card\... (PC-Card)  
\Storage Card2\... (CF-Card)  
\Flash\... (internal flash memory)  
Network path (when a network is created on an operating unit)

---

### Note

When creating data records on the CF card, make sure it is inserted. If not, the data will be stored in the RAM file system of Windows CE and are lost on power down.

---



---

**Note on the flash memory**

Cyclic write access is not allowed for the internal flash memory, since it reduces the life of the flash memory and thus the life of the operating unit.

---

3. Set transfer synchronization: You can set whether data records will be transferred with or without synchronization using the PLC. You require the "data mailbox" communication area for transfer with synchronization.

4. Create the recipe view/recipe screens:  
Configure one or more screens if you want to be able to create, save and transfer data records on the operating unit. For this, depending on the case, use either the recipe view or map your system to recipe screens with input/output fields.  
Recipe view:  
The recipe view enables you to handle recipes and data records quickly and straightforwardly, with minimal configuration. Typically, you will use it to edit data records of smaller recipes in tabular form.  
Recipe screens:  
You can use recipe screens to customize the user interface for editing data records and simulate the visual appearance of your system by using graphics and individual masks for entering the data records. Recipe screens are particularly recommended for data records with a relatively high number of entries.
5. You decide whether to use recipe view or recipe screen editing options in Recipe Properties, by setting the "Synchronize tags" option on the "Options" tab.  
"Synchronize tags":  
The data of a data record which has been read is written to the tags or read from the tags you configured for the recipe. If you want to be able to enter and change data records via a recipe screen, position the tags defined in the recipe in screens, by using input fields, for example.  
"Do not synchronize tags":  
The data of a data record which has been read are only displayed in the recipe view table and can only be edited in the recipe view. If you use the same tags in screens as well, this does not affect their values.

---

**Note**

Take note in this respect of "Editing data records".

---

**Text or graphic list for recipes**

When you create a project, a text or graphic list called `Z_RECIPES` is automatically created. The sequence in which the recipes are entered in the text or graphic list is determined by the parameter.

## Tags in recipes

The following configured properties of tags are not evaluated in recipes:

- Unevaluated properties with non-synchronized recipes: – Functions triggered by set events – Data entry cycle and update (Read Continuously) – Tag limit values – Archiving
- String tags: In string tags, the hexadecimal value 00h is interpreted as an identifier for the end of the string. Subsequent characters in the string are neither read nor displayed. When written to the PLC and when saved to the data medium, subsequent characters are padded with 0.
- Timers/counters: The range of values is not checked on the operating unit when data are entered in the recipe view in the case of "timer" and "counter" type tags. Remedy: Configure fixed limit values for the tag if you want the range of values to be checked when data are entered.
- Date/time: "Date", "Time of Day" and "Date and Time" tags are all displayed in the same format in recipes, namely DD.MM.YYYY/HH.MM:SS. The invalid part for the tag is assigned a default value when displayed.

## 5.12.5 Synchronization with the PLC

### Purpose

With synchronous transfer, both communication peers set status bits in the data mailbox used by them both. In this way, you can prevent uncontrolled, mutual overwriting of data in your PLC program.

### Usage

Synchronous data record transfer is then always available, for example, if:

- The PLC is the "active peer" in the transfer of data records
- Information about the recipe number/name and data record number/name have to be evaluated on the PLC.
- A transfer of data records is initiated by a function or a PLC job, for example with the functions `Data_Record_DAT_To_PLC` and `Data_Record_PLC_To_DAT`, or the PLC jobs `Write_Data_Record_To_PLC` and `Read_Data_Record_From_PLC`

---

### Note

Greater flexibility is achieved by using functions rather than PLC jobs.

---

### Requirements

If you want data records to be transferred between the operating unit and the PLC with synchronization, the following requirements must be met during configuration:

- The "Data Mailbox" area pointer is created in the project window at "Area pointer".
- The PLC, with which the operating unit synchronizes the transfer of data records is specified in the recipe properties.

You set the PLC in the recipe editor: "Properties" button, "Transfer" tab.

---

### Note

Communication with the PLC is described in the "Communication for Windows-based Systems" manuals, in the section called "Data mailbox for synchronized transfer."

---

## 5.12.6 Editing data records

### Creating data records

The following options are open to you for creating data records:

- Data records are created by operator input on the operating unit.
- You use the functions `Export_Data_Records` and `Import_Data_Records`.

### Procedure for external creation

1. Create a data record with a symbolic name.
2. Configure the function `Export_Data_Records` in order to export data records on the operating unit to the storage medium as CSV files. In this way you can easily edit data records using external tool, such as a spreadsheet program or text editor. Information such as recipe name, recipe number, data record name and data record number is stored in the first two lines. That is followed by the values for the data record; this then provides a pattern for the structure.
3. You then reimport the data records on the operating unit with the `Import_Data_Records` function.

Detailed instructions on how to create data records and transfer them to the PLC are given in the manual "Recipes for Windows-Based Systems".

---

#### Note

Only use Western European characters in data record names.

---

### Transferring data records

If you want to transfer data records, you have the following options:

- Operating the recipe view buttons on the operating unit.
- PLC jobs 69 (read data record from the PLC) and 70 (write data record to PLC).
- The configured functions: `Data_Record_DAT_to_PLC` and `Data_Record_PLC_to_DAT`

Detailed information on the "data mailbox" communication area can be found in the manual "Recipes for Windows-Based Systems".

### Deleting data records without recipe view

Using the function `Delete_Data_Record` you can delete data records without having to configure a recipe view. The function `Delete_Data_Record` allows you to delete all or individual data records in a recipe.

When the function is triggered or terminated, a system message is issued ("Started to delete data record" or "Stopped deleting data record").

## 5.12.7 Structure of the "data mailbox" area pointer

### Principle

The "data mailbox" communication area is required on the PLC for the transfer of data records with synchronization.

The values of a data record tags are written directly to the addresses of the tags configured for the PLC. Before transferring, the system checks whether this transfer is allowed and issues a system message indicating whether or not errors occurred during transfer.

---

### Note

For further information about the direct transfer of data records refer to the "Communication for Windows-based Systems" manual.

---

You create the data mailbox in your project at "Area pointer" in the project window.

The data mailbox is created with a length of 5 data words and has the following structure:

<b>Word 1</b>	Current recipe number (1 – 999)
<b>Word 2</b>	Current data record number (1 – 65535)
<b>Word 3</b>	Reserved
<b>Word 4</b>	Transfer status
<b>Word 5</b>	Reserved

## 5.12.8 Compatibility of recipes

### Special features with recipes

Please take into consideration the following notes on copying recipes via the clipboard:

- Windows-based system > Windows-based system During copying, all fields and their properties are assigned the values used in the source project.
- "Windows-based system" > "graphics display" It is not possible to copy the recipes of Windows-based systems to projects for graphics displays.
- "Graphics display" > "Windows-based system" Properties not available in the source project (for example, storage location) are given the default settings in the destination project. Settings no longer available in the destination project are discarded. The standard screens and data record functions of the devices having a graphics display are not converted.

Please note that if you are using the same PLC, the tag addresses may be discarded even when changing the connection type.

You can prevent this by taking an intermediate step, for example:

Source:	OP 37 project	using	SIMATIC S5 L2-DP
	↓		
Intermediate step 1:	OP 37 project	using	SIMATIC S5 AS511
	↓		
Intermediate step 2:	PC project	using	SIMATIC S5 AS511
	↓		
Destination project:	PC project	using	SIMATIC S5 DP

---

#### Note

For further information refer to the brief instructions "Configuration information for upgrading".

---



---

#### Note

The following applies to Windows CE devices:

In ProTool 2 and higher, the regional options such as format of the date and time and decimal and list separators are "German" by default. Modify this setting or replace the separators in the file, as necessary, if you are using archives or exported recipe data records in ProTool 5.2 or higher that you created in ProTool 5.1. The reason for this is that the operating unit can read archives and exported recipe data records correctly only if the set decimal and list separators are identical at the time of reading and writing.

---

### **5.12.9 Examples and scenarios for creating and configuring a recipe**

The examples "How to create a recipe" and "How to configure a recipe view" show you how to create a recipe for the mixing unit of a fruit juice system.

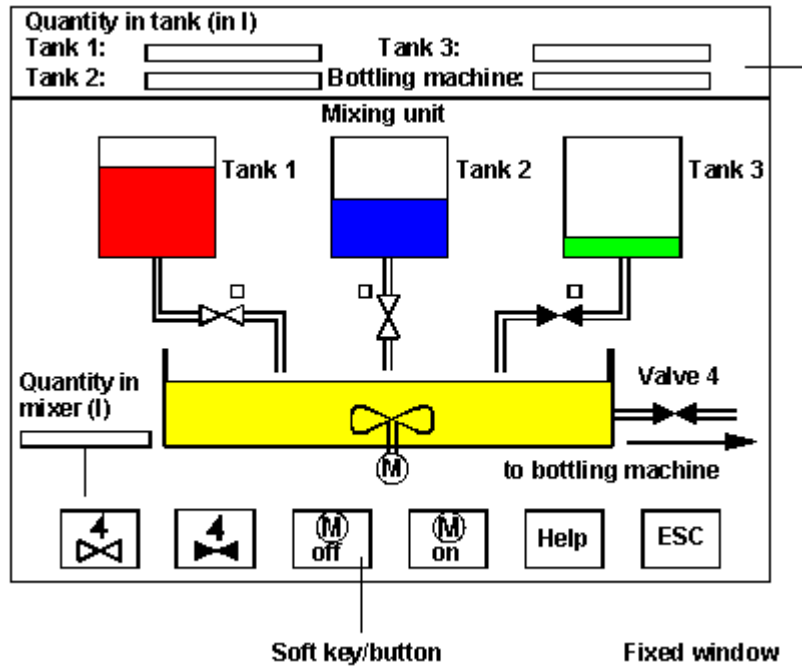
The three scenarios are typical applications in practice concerning the subject of recipes.



### 5.12.9.1 How to create a recipe

#### Task

In this example you create a recipe for the mixing station of a fruit juice system. Different fruit juices are to be mixed using the same system. The ingredients are the same; only the mixing proportions are different.



#### Example of a Screen: Mixing Station

You begin by creating a recipe called "mixture" and then a data record with the name "juice". This data record contains the mixing proportions for orange juice.

The data record is transferred to the PLC using the synchronized transfer option, i.e. a data mailbox is required. During transfer of the data record, the operating unit sets bits in the status word of the data mailbox. You then have to set the status word to zero in the PLC program in order to enable the data mailbox again.

The task involves the following steps:

- Creating a recipe
- Configuring the recipe view
- Transferring a project file to the operating unit
- Creating a data record on the operating unit
- Transferring a data record to the PLC

This example shows you how to create a recipe. The other subtasks are dealt with in the example "How to configure a recipe view".

**Procedure:**

The example recipe is to be created for a PC 670 connected to the SIMATIC S7-300 and SIMATIC S7-400 PLC through an MPI.

**Creating a recipe**

Procedure:																					
1	Create a new project and save it under a name of your choice.																				
2	In the project window, double-click "Recipes" to create the recipe described below.																				
3	<p>Create four REAL tags, Var_2, Var_3, Var_4 and Var_5, for the ingredients orange (in liters), water (in liters), sugar (in kilograms) and flavor (in grams) and assign the following addresses:</p> <pre>Var_2 DB 120, DBD 0 Var_3 DB 120, DBD 4 Var_4 DB 120, DBD 8 Var_5 DB 120, DBD 12</pre> <p>Assign the tags the illustrated entry names in the "Recipe" dialog box and add the entries for the recipe structure. You define the structure of the recipe by means of the order of the tags.</p> <table border="1"> <thead> <tr> <th>Tag</th> <th>Entry Name</th> <th>Offset</th> <th>Length</th> </tr> </thead> <tbody> <tr> <td>VAR_2</td> <td>l_Orange</td> <td>0</td> <td>4</td> </tr> <tr> <td>VAR_3</td> <td>l_Water</td> <td>4</td> <td>4</td> </tr> <tr> <td>VAR_4</td> <td>kg_Sugar</td> <td>8</td> <td>4</td> </tr> <tr> <td>VAR_5</td> <td>g_Flavor</td> <td>12</td> <td>2</td> </tr> </tbody> </table>	Tag	Entry Name	Offset	Length	VAR_2	l_Orange	0	4	VAR_3	l_Water	4	4	VAR_4	kg_Sugar	8	4	VAR_5	g_Flavor	12	2
Tag	Entry Name	Offset	Length																		
VAR_2	l_Orange	0	4																		
VAR_3	l_Water	4	4																		
VAR_4	kg_Sugar	8	4																		
VAR_5	g_Flavor	12	2																		
4	Click the "Properties" button in the "Recipe" dialog box. On the "General" tab, change the name of the recipe to <i>mixture</i> . Set the recipe number to 1.																				
5	Select your PLC on the "Transfer" tab.																				
6	On the "Options" tab, deselect the "Synchronize tags" check box. This will mean that the data of a data record which has been read is only displayed in the recipe view table and can only be edited in the recipe view. If you use the same tags in screens as well, this does not affect their values.																				
7	<p>Create the "data mail box" communication area at "Area pointer" in the project window. Click the OK and set the address:</p> <p>The length is fixed at 5. Area: DB100, DBW 0</p> <p>Make sure that you create data block 100 with a length of at least 5 data words in your PLC program as well.</p>																				
8	On the "Data Medium" tab, enter C:\TEMP\REC as the path name.																				

### 5.12.9.2 How to configure a recipe view

#### Task

In this example you configure the recipe view for the mixing unit of a fruit juice system. You transfer the project file to the PLC, create a data record and transfer it to the PLC.

#### Configuring the recipe view

Procedure:	
1	On a screen, configure the "recipe view" screen object with the following buttons: "New", "Save", "Delete", "From PLC" and "To PLC".
2	At "Data record", select the settings "Display table" and "Allow edit".

#### Transferring a project file to the operating unit

Procedure:	
1	Establish a physical connection with the operating unit.
2	Transfer the project to the operating unit.

#### Creating a data record on the operating unit

Procedure:	
1	Select the screen with the "recipe view".
2	At "Recipe name", select the recipe called <code>mixture</code> .
3	Click the "New" button. Position the cursor in the "Data record name" input field and enter <code>Juice</code> as the name.
4	Move the cursor to the input field at "data record name". Then enter the following values: <code>Orange 95</code> <code>Water 5</code> <code>Sugar 3</code> <code>Flavor 250</code>
5	Click the "Save" button. This saves the data record to the specified path.

**Transferring a data record to the PLC**

<b>Procedure:</b>	
1	Press the "To PLC" button to transfer the data record from the operating unit to the PLC. During transfer of the data record, the operating unit sets bits in the status word of the data mailbox.
2	Confirm the transfer in the PLC program.

**Result**

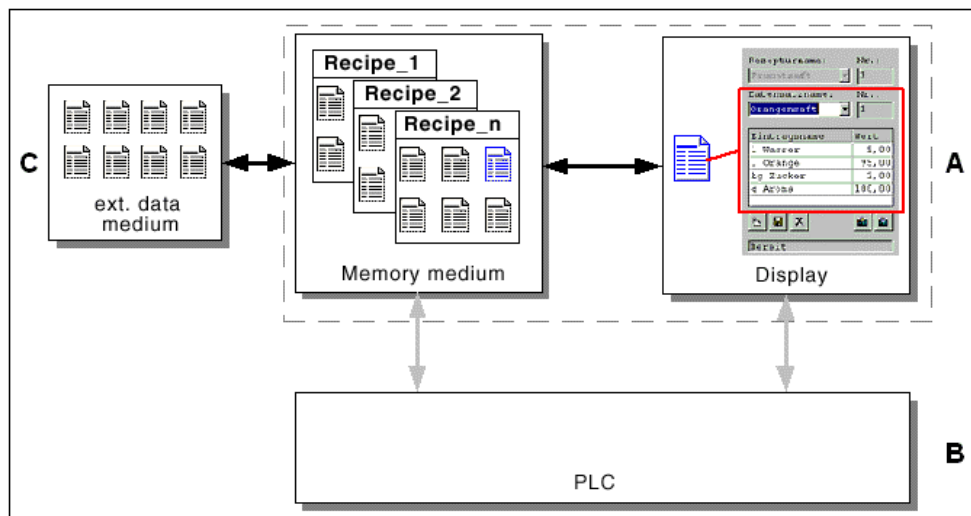
The values for the "Juice" data record are now in the memory area of the PLC.

### 5.12.9.3 Scenario 1: How to enter data records

#### Objective

You want to enter production data on the operating unit (A) without interrupting the active process. The production data are not to be transferred to the PLC (B) for that reason.

#### Process

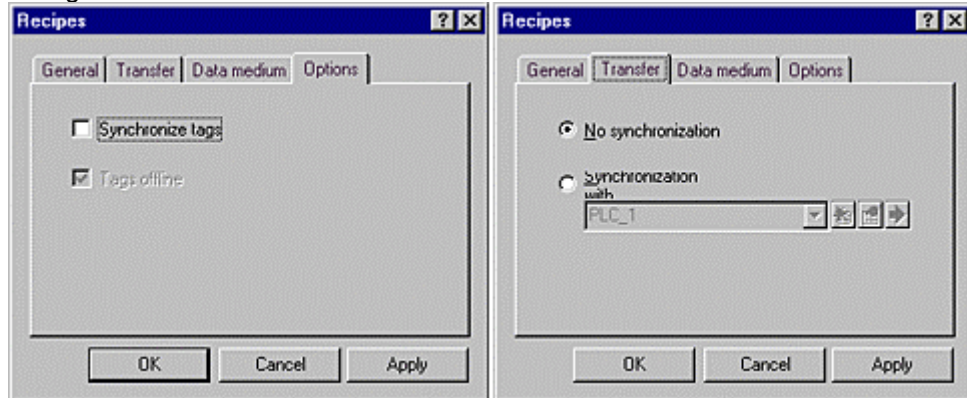


Enter the production data in the recipe view or on the recipe screen, assign a data record name and save the data record to the storage medium of the operating unit.

## Implementation options in ProTool

In ProTool, you configure the structure of the recipe with the associated tags:

1. Since we do not want the production data to be transferred to the PLC, synchronization with the PLC is not necessary. We do not want to have the tags transferred to the PLC. Click "Properties" and set the following configuration:



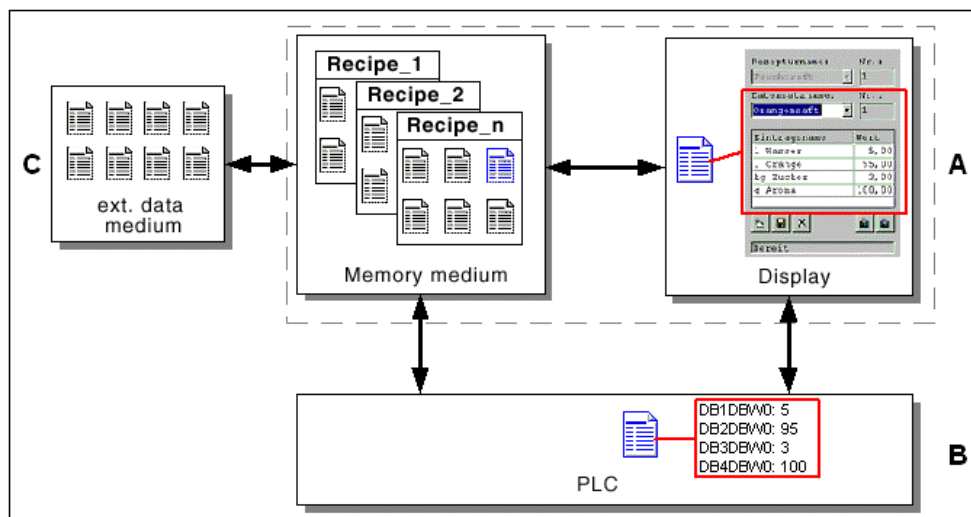
2. Depending on the size of the recipe, configure either a recipe view or create a recipe screen.

### 5.12.9.4 Scenario 2: Manual production process

#### Objective

The production data have to be requested from the PLC (B) as a function of the work piece we want to process and shown on the display of the operating unit (A) for inspection. If necessary, you want to be able to correct the transferred production data online.

#### Process

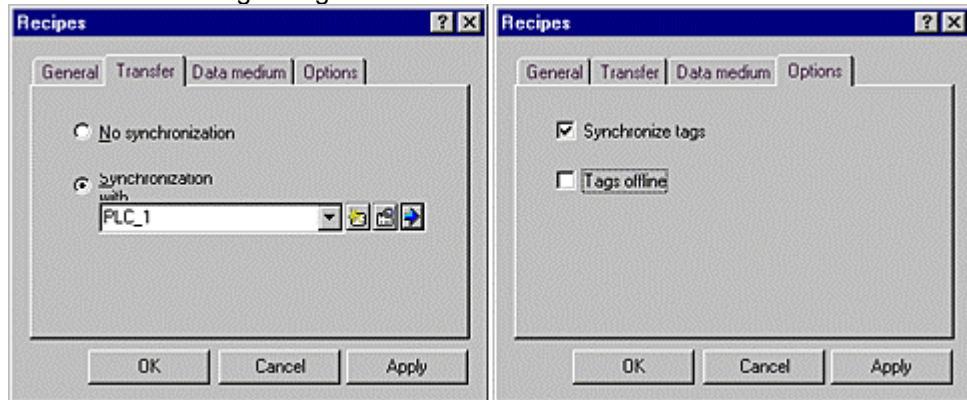


A reader connected, say, to the PLC reads out a barcode on the work piece we want to process. The data record names correspond in this instance to the respective barcode IDs. This means that the PLC can load the required data record from the storage medium of the operating unit. The data record is shown on the display for inspection purposes. Changes are transferred immediately to the PLC.

## Implementation options in ProTool

In ProTool, you configure the structure of the recipe with the associated tags:

1. Since the production data are required to be transferred to the PLC, synchronization with the PLC is necessary to prevent uncontrolled mutual overwriting. We want to have the tags transferred to the PLC. Click "Properties" and set the following configuration:



2. Depending on the size of the recipe, configure either a recipe view or create a recipe screen.

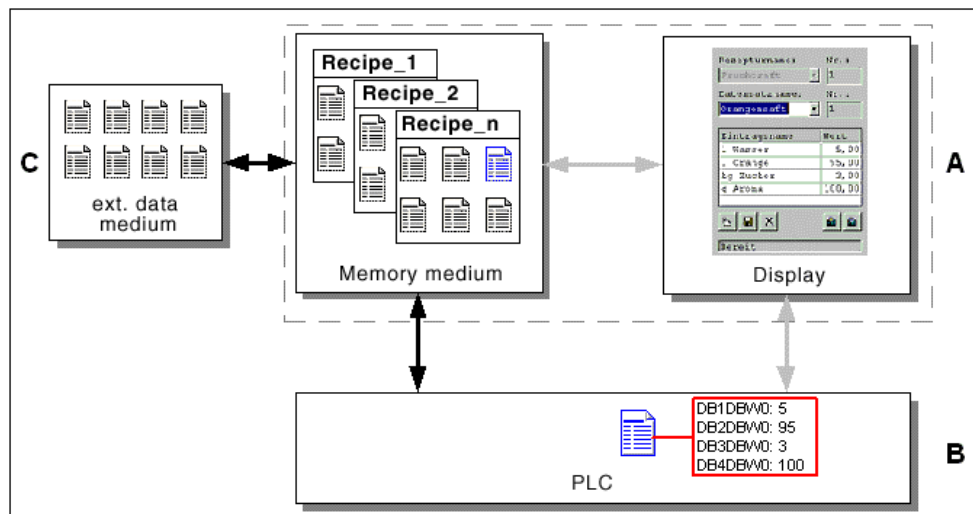


### 5.12.9.5 Scenario 3: Automated production process (job list)

#### Objective

You want to run production that is automated by means of a job list. The production data are required to be transferred either from the data medium of the operating unit (A) or from an external data medium (C) directly to the PLC (B). There is no need for data to be shown on the display.

#### Process



Production can be controlled by means of one or more VB scripts, which automatically transfer the production data records to the PLC. The process can be monitored by means of return values of the functions being used.

#### Implementation in ProTool

You can implement the automated production process with the available data record functions: The function `Import_Data_Record` loads data records from a \*.csv file to the data medium. The function `Data_Record_DAT_To_PLC` transfers a data record from the data medium to the PLC.

You will find more information about programming with VBScript at "Creating VBScripts".

## 5.13 Prompting the operator

### Purpose

In addition to the option of customizing the user interface of the operating unit to make it easier for you to use, ProTool features other options for supporting and prompting you on your operating unit as a function of the situation. This means you can implement mechanisms and decision-making aids that can effectively prevent incorrect operations on the operating unit.

ProTool makes the following methods available for the implementation of operator prompting:

- Providing Help text
- Displaying the on-screen keyboard
- Assigning icons to local function keys
- Displaying and hiding screen objects
- Assigning dynamic attributes
- Driving light-emitting diodes
- Configuring the tab order
- Dynamically positioning screen objects
- Assigning operator authorization

### 5.13.1 Help information on the operating unit

#### Purpose

Help text provides additional information and operator instructions on screens, input and output fields, and buttons and messages to the operator at run time. Help text provides additional information that can be defined for screens, input/output fields, buttons and messages when configuring with ProTool. For example, Help text may be in the form of the permissible range of values for an input field or the cause and elimination of a malfunction in the case of an alarm message. Help text is displayed on the operating unit in the language you set.

You configure Help text on the "Help Text" tab in the "Properties" dialog box of the object in question.

In addition to Help text, you can configure tooltips on some devices for certain screen objects; the tooltips are displayed at run time for limit values in input fields, for example.

#### Displaying Help text on the operating unit

- All devices except PC The availability of Help text is indicated on the operating unit by the lit "HELP" LED. On touch panels, the "HELP" button is displayed. The configured Help text can be output by pressing the "HELP" key. Detailed information about activating Help texts is given in the equipment manual for your operating unit.
- PCs Assign the function "Show Help Text" to function key "F1", for example. By pressing that key, the operator can output the configured Help text at run time.

---

#### Note

The function `Display_Help_Text` normally only makes sense with the function keys. When you operate a function key, you receive help on the screen object for which the focus is currently activated. If you configure the `Display_Help_Text` function for a button, you get either the configured Help for the button or for the current screen when you operate the button by clicking the mouse or by touch operation.

---

## 5.13.2 On-screen keyboard

### Requirements

The on-screen keyboard can be enabled on all operating units (except the OP 170B).

### Purpose

An on-screen keyboard can be displayed for entering values on the operating unit. The operator can click the illustrated keys with the mouse or (with touch-operated devices) or touch them to fill out the input field.

Depending on the selected input field, the on-screen keyboard features keys for purely numerical input or for entering alphanumeric characters.

In the case of numeric input fields, the operator is shown the minimum and maximum values allowed for input. The displayed values are determined from the limit values configured in the tag dialog box for the input field.

The figure shows an example of a German on-screen keyboard for PCs and Panel PCs for entering alphanumeric values.



*On-screen keyboard for PCs and Panel PCs*

### Enabling the on-screen keyboard

The manner in which the on-screen keyboard is enabled depends on the operating unit:

- On touch-operated devices, the on-screen keyboard is always displayed automatically as soon as the operator selects an input field. After completing the input by means of RETURN, the on-screen keyboard disappears again.
- For all other devices, you have to configure on-screen keyboard display.

In addition, you can configure a key with which the operator can enable or disable automatic display of the on-screen keyboard as required. This is practical, for example, when an external keyboard is connected to the operating unit and the on-screen keyboard is therefore not required for input.

**Position of the on-screen keyboard on the screen**

The default position of the on-screen keyboard is at the lower border of the screen.

- Panel PCs and standard PCs: The on-screen keyboard can be positioned, minimized and maximized just like any other window on the screen.
- Windows CE devices: The position can be set on the operating unit on the Control Panel in Windows CE.

### 5.13.3 Icons for local function keys

#### Purpose

Icons are fixed-size pixel graphics that are placed in the immediate vicinity of screen-specific function keys (soft keys). This means that you can illustrate the function of a locally assigned function key. Icons can only be assigned to soft keys that border on the screen of the operating unit.

You can prevent incorrect operation of a key to a large extent by means of an informative icon.

---

#### Note

Not possible on a standard PC.

---

### 5.13.4 Showing and hiding objects

#### Purpose

ProTool features an option of displaying and hiding controls and display elements at run time. For example, you can hide an output field on the operating unit when the actual value is within the specified setpoint range.

Alternatively, display an input field only when an operator input is expected for the control in a specific situation for example, during startup. The user interface of your operating unit remains neat and tidy in this manner.

#### Index tag

You can either link display and hiding to a constant range of values or to the tag of an input/output field, for instance. You can influence several input/output fields simultaneously by using a separate tag.

If you use a tag with PLC connection to enable and disable, you can activate certain project versions for different operating units from the PLC.

### 5.13.5 Dynamic attributes

#### Purpose

To draw the operator's attention on the operating unit to specific situations for example, specified limit values have been reached or exceeded you can configure attributes for input and output elements. This means that you can dynamically modify the foreground and background colors, for example, of an input/output field at run time as a function of the value of a tag, or you can enable or disable flashing for the text that is being displayed.

---

#### Note

The background and foreground colors are automatically assigned by the system to the graphic view, graphic list and graphic input fields. A background color or foreground color configured in ProTool is therefore not displayed at run time.

---

#### Index tag

You can either link the specified attributed directly to the tag of an input/output field, for instance, or you can define a separate index tag for it. You can influence several input/output fields simultaneously with a separate index tag.

### 5.13.6 Light-emitting diodes on the operating unit

#### Purpose

The light-emitting diodes (LEDs) on the function keys of the operator panel can be driven on the PLC. By means of a light-emitting or flashing LED, you can indicate to the operator that, for example, the device is expecting a certain function key to be operated.

#### Requirements

For the PLC to be able to drive the LEDs, you have to create the "LED assignment" data area on the PLC and specify it in your project, in your project window, at "Area pointers". Set the assignment of the individual LEDs to the bits in the LED assignment when you configure the function keys.

You will find a description of the LED assignment and the LED functions for the different operating units in the "Communication User's Manual".



## 5.13.7 Tab order

### Tab order

To make it possible for a keyboard device without a mouse to access and operate each operated screen object (such as a trend display or output field), the operated screen objects can be accessed with the Tab key. You can see which screen object has been made active, because it may be highlighted with a color or surrounded by a border.

The expression "Tab order" means the order in which the operated screen objects are made active when the Tab key is pressed.

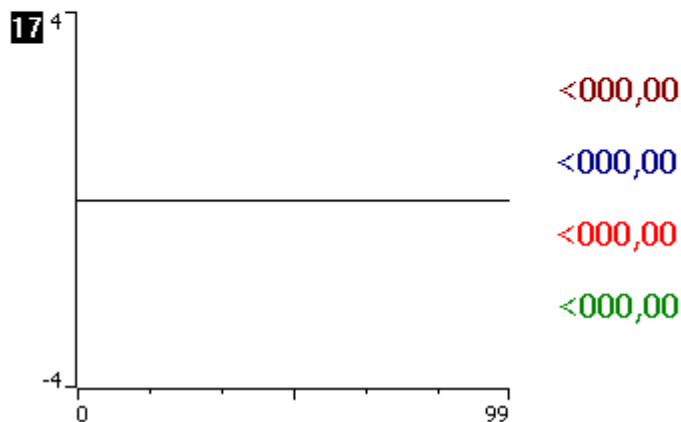
### Configured tab order

Usually the tab order is set automatically: in the order in which the screen objects were created.

In ProTool you can also configure the tab order yourself. This can be useful if you want to enable the operator to switch to a certain screen object quickly, or if a particular sequence is dictated by the screen purpose itself.

To do this, go to the ProTool screen editor and switch to a different editing mode by choosing "Edit" > "Tab Order" from the menu. In this mode you can edit only the tab order of screen objects, not the screen objects themselves.

a: **1**=000.00 **2**=000.00 **3**=000.00 **4**=000.00  
 b: **5**=000.00 **6**=000.00 **7**=000.00 **8**=000.00  
 c: **9**=000.00 **10**=000.00 **11**=000.00 **12**=000.00  
 Calculation: **13**=00000 **14**=00000 **15**=00000 **16**=00000



Screen in Tab Order mode

In Tab Order mode a number is displayed upper left of each operated screen object; this is the "tab order number". Using the mouse, you can then freely edit the assignment of these numbers. You can also exclude screen objects from the Tab order. With the Tab order number, you can now set which field will be active after a change of screens, which might be triggered by `Select_Screen_Permanent` or `Select_Screen_Variable`. Press ESC or use the menu to exit Tab Order mode.

### **Tab order of imported screens**

Screens from projects created with earlier versions are given a standard sequence: top left to bottom right. Thereafter, newly added elements are numbered in the order in which they are created.

### 5.13.8 Width and order of columns (screen objects)

#### Purpose

So that the operator himself does not have to set the column width and order

This means that you can design screen objects such as message view, password list, recipe view and Status/Force in a more comprehensible manner for the operator. You can place important items on top and optimally adjust column width to the contents.

For this, there is an editing mode in the screen editor which activates a screen object (that is, makes it operable).

#### Procedure:

With SHIFT and double-click on the desired screen object or by choosing <Screen Object Name>" > "Edit" on the shortcut menu, you can activate the screen object, that is then surrounded by a hatched border. The zoom factor is set to 100% to ensure that the screen object is displayed in its entirety. If you have set the width and order of the columns, simply click on a vacant point in the screen editor. The old zoom factor is reset automatically.

## 5.13.9 Dynamic screen objects

### What do we mean by dynamic positioning?

You can dynamically modify the position of objects that you use on a screen. This means that the values set in your project for the horizontal and vertical positions can be changed at run time. Dynamic positioning is possible with screen objects such as bar graphs or input/output fields, text or graphics.

In this way, you can move a graphic on the screen, for example, in order to visualize the current position of a conveying belt on the operating unit.

### How dynamic positioning works

When you are configuring, you normally set the X and Y coordinates for the different objects. You define these values either explicitly in the Properties dialog box for the object concerned on the "Position" tab or indirectly by positioning with the mouse. You can view the X and Y coordinates while you are configuring by choosing "View" > "Size/Position" from the menu.

The X and Y values for the initial position and the size of the object are applied automatically and displayed in the dialog box.

To position an object dynamically, specify tags for the X and/or Y coordinates at "Offset". The current values of these tags determine the "step length", which is added at run time to the X and Y values of the start position. In this way the position of the object varies with time on the screen of the operating unit.

If the coordinate values exceed those permissible at run time, they are adapted automatically so that the object remains visible on the screen.

## 5.13.10 Assigning operator permissions

### Purpose

ProTool allows you to use a password to prevent controls such as input fields and function keys from being used by those unauthorized to do so. This means that when you are creating your project you can restrict the use of functions that relate to security to specific people or groups of operators. Important parameters and settings can thus only be changed by authorized personnel.

The access protection that you configure allows you to guard against incorrect usage and increase the security of the system or machine.

### Password hierarchy

During the configuration phase you can assign operator authorization to specific groups. At run time, individuals can be allocated to one of these groups, as appropriate, and they thus automatically receive the access rights of that group.

ProTool provides hierarchically organized password levels from level 0 to 9. If a user is assigned password level 4, for example, this user is authorized to execute the functions of password levels 0 to 4.

- Password level 0  
Password level 0 is the default in ProTool. Use this lowest level in the hierarchy for functions that have little or no effect on the operational sequence. These are generally functions that do not require any input, such as the display of messages. There is no need to enter a password on the operating unit in order to execute functions with password level 0.
- Password levels 1 to 8  
Assign functions to password levels 1 to 8 according to their importance. Before you execute these functions, the operating unit prompts the operator to enter a password.
- Password level 9  
The authorization to execute functions at password level 9 is granted only to the system administrator or service engineer. He has access to all the functions of the operating unit.

You define the password of the system administrator by choosing "System" > "Settings" from the menu. The default setting is 100.

The administrator's password setting cannot be changed on the operating unit at run time. All other passwords are entered at run time; you have to configure a password list for this.

You can find more information on password administration in the equipment manual for your operating unit.

### Logging into and out of the operating unit

- When a password-protected operation is executed, the operating unit automatically prompts the operator to enter an appropriate password. In order to eliminate the possibility of unauthorized persons gaining access, a password level greater than 0 should not remain active on the operating unit for any length of time.
- If you do not enter an input on the operating unit for a configured period of time (logout time), the operating unit automatically resets the current password level to 0. You set the logout time by choosing "System" > "Settings" from the menu. The system default setting is 5 minutes.

---

#### Note

Login and logout operations are automatically output as system messages. If you have configured a message archive, all login and logout operations on the operating unit will be archived.

---

### Setting the password level on the operating unit

The following PLC jobs are available for setting a defined password level on the operating unit:

- You can use PLC job 23 to set a password level of your choice on the operating unit from the PLC in order to allow access by a defined user group, for example.
- You can use PLC job 24 to reset the password level back to 0 from the PLC.

In online Help, you will find a list of all possible PLC jobs with job numbers and parameters in PLC jobs in alphabetical order and PLC jobs in numerical order.

### 5.13.11 Password administration

#### Purpose

So that an operator can use the operating unit to enter and change passwords and user names including their associated password levels for operators, you configure a password list.

Password	Level



You configure a password list on the open screen in the "Screen Objects" toolbar by selecting the illustrated button or by choosing "Insert" > "Password List" from the menu.

---

#### Note

You can enable the password list while you are configuring by using SHIFT and double-click or from the "Edit" option of the shortcut menu. In active mode you can drag and drop a column to the required position and set the column width during the configuration phase.

---

#### Operation at runtime

At run time, the operator has to confirm each input with "Enter" to ensure that it is applied.

Since not every operator is entitled to manage users and their passwords at will, display of the password list at run time depends on the password level of the logged in operator. An operator logged in at password level 2 will only see entries at this and a lower password level in the list and can only create the same.

---

#### Note

If you want to make the password list available only to the administrator as a general rule for user management, configure it on a separate screen protected by the administrator's password.

---

## Functions to manage passwords

You can use a password to protect control elements such as input fields or function keys from unauthorized persons. This prevents operator error and improves the security of the system or machine.

In addition, there is the option of the operator logging on and off the operating unit irrespective of a control. For this there are the following globally configurable functions:

- `Write_Password_To_Tag` The a user's password is written to a tag when a user logs on, it can be announced to the PLC in this manner. This means that certain permissions can be enabled on the PLC.
- `Report_Password_Change` Login and logout operations on the operating unit can be traced in connection with an archiving function by means of this function.
- `Write_Password_Level_To_Tag` The value of the current password level on the operating unit is written to a tag and made available to the PLC. In this way user-independent enabling of certain functions can be implemented.
- `Write_User_Name_To_Tag` This function writes the current user name to a tag. If the tag is connected to the PLC, certain functions can be enabled on the PLC as a function of the user.



## 5.14 Creating VB scripts

### Overview

ProTool features an interface for scripting with Microsoft Visual Basic Script (VBScript). VBScript allows you to add your own functions to the ProTool configuration software. All the data types, operators and functions offered by VBScript are available. Scripts which you created yourself in Visual Basic can be used in ProTool as functions.

Exhaustive Microsoft documentation on VBScript is supplied with ProTool and can be accessed from ProTool Help

Only the essentials of script creation in ProTool are described here. These are:

- Creating scripts in ProTool
- Using scripts in ProTool
- ProTool objects that can be used in scripts
- Use of internal script tags and ProTool tags
- Debugging scripts in ProTool
- Constraints with VBScript

---

### Note

You should read the Microsoft documentation before programming with VBScript and be clear about the consequences in ProTool. VBScript not only enhances the functionality of ProTool; it can also lead to unwanted errors in online operation.

---

### Constructs of VBScript

VBScript is a simple script language in which the following constructs are used:

- Tags These can be either ProTool tags or internal VBScript tags
- VBScript operators such as +, -, \*, or, and
- Conditional statements such as If...then...else, Select Case
- Looping through code such as Do...Loop, While...Wend, For ... Next, For each ... Next
- VBScript procedures such as Sub and Function
- Built-in VBScript procedures such as Date, Sin, Sqr

A VBScript recognizes two types of procedure. These are featured on the "General" tab:

- Subroutine
- Function

In contrast to a function, a subroutine does not have a return value. Parameters can be passed with both types of procedure.

### Brief overview of the use of scripts

ProTool objects can be used and edited when programming scripts. You can also call ProTool functions and scripts you yourself have created in scripts. A script could, for example, compare two values with each other and supply the greater of the two as the return value.

In the left half of the project window you will find the "Scripts" object type. You save the script with a symbolic name, which you see in the right half of the project window.

ProTool supports you with color highlighting when you program scripts. Tags, statements, objects and comments are colored appropriately.

- VBScript instructions are blue
- Comments are green
- ProTool function and script names are gray
- ProTool tags are red
- Screens are yellow
- Archives are violet

Before a script can be used, it must be compiled. ProTool checks the syntax and tags.

In the case of all objects to which functions can be attached, you can also use your programmed scripts. For scripts, just as with functions, events are also specified on whose occurrence the script is to be executed (for example, "Value Change" or "Key Pressed").

### Limits

The number and scope of the scripts depends on the operating unit you are using (refer to "System limits").

## 5.14.1 Using ProTool objects in scripts

### Overview of ProTool objects

The following ProTool objects can be used and edited when programming scripts

- Tags
- Functions
- Scripts
- Screens

### Notes on using tags

- You can use a script, for example, to read a value from a tag, convert it and assign it to the tag again.
- Query BOOL tags: The two logic states "TRUE" and "FALSE" are represented in VBScript by the values -1 ("TRUE") and 0 ("FALSE").  
Example: This is how you query the logic state "TRUE" of tag "Bool\_Var":  
`IF Bool_Var <> 0 ...`

Please note that the logical states in VBScript are not the same as the definitions for BOOL tags in ProTool:

Status	VBScript	BOOL tag in ProTool
TRUE	-1	1
FALSE	0	0

### Tag types permitted for OLE applications

Only use Western European characters in data record names. If you wish to read to or write from tags in the Runtime software with another application such as a spreadsheet program, note that Visual Basic only supports the following programs:

- Unsigned char (8-bit, unsigned)
- Short int (16-bit, signed)
- Long int (32-bit, signed)
- Float (4-byte, floating point)
- Bstr (OLE Automation, string)
- Bool (-1: TRUE; 0: FALSE)
- Arrays with the above-named types

You will find an example of access to Runtime tags on the ProTool CS CD-ROM at "Docs\FSpec".

### Notes on using functions

- Do not program any scripts that already exist as ProTool functions.
- If you want to use a function of the `Set_Bit_In_Tag` type in a script, note that the bit count direction always runs from right > left and starts at 0, regardless of the PLC used.

## 5.14.2 Internal script tags and ProTool tags

### Tags in scripts

Tags are placeholders that occupy storage space on the operating unit and to which values can be written or from which they can be read during the processing of a script. There are two types of tag:

- ProTool tags ProTool tags appear red in the script. When a ProTool tag is used in a script, the tag must have already been created in ProTool. Otherwise, an error message appears at compilation.
- Internal script tags Internal script tags have to be defined with the "DIM" statement and apply only in the VBScript procedure.

ProTool supports the following data types for tags:

Format	data type
8 bits with sign	CHAR, CHAR_ARRAY
8 bits without sign	BYTE, BYTE_ARRAY
16 bits with sign	INT, INT_ARRAY
16 bits without sign	UINT, UINT_ARRAY
32 bits with sign	LONGINT, LONGINT_ARRAY
32 bits floating point	FLOAT, FLOAT_ARRAY
64 bits floating point	DOUBLE, DOUBLE_ARRAY
64 bit	DATETIME
1 bit	BOOL, BOOL_ARRAY
Character strings	STRING

### 5.14.3 Debugging scripts in ProTool

#### Syntax check

When you click the "Compile" button, the script is examined for syntax errors. Since the VB debugging tool is not available in ProTool, we recommend that you test complex scripts directly in Visual Basic or VBA. Bear in mind that VBScript does not offer the full scope of Visual Basic as a language. In addition, ProTool tags must be simulated by local VBScript tags.

As in any other program, however, programming errors may not appear until runtime. These are displayed as system errors at script runtime, preceded by `VBScript Error:`. The name of the script and the line number in which the error occurs are displayed.

#### Check criteria

To avoid errors, you should check your scripts for the following:

- Are direct or indirect recursions used correctly?
- Are there any continuous loops?
- Are the function parameters correct?
- Are the tag types used for the parameters correct?
- Is the number of parameters in a script correct?
- Are you using time-consuming routines?
- Do you call the scripts very often?

To debug, you use the function `Display_System_Message_Text`. This function allows you to check which program steps have been carried out. Call the function in the script at the places where you would like to check the debugging session. Transfer text or a value to the function. It will be read out as a system message while the script is running. The script can thus be processed step by step.

---

#### Note

Only one script can be executed at any particular time.

---

## 5.14.4 Constraints with VBScript

### Assignment of symbolic names

The names of VBScript constructs, such as If, Then, Loop, Sub and Dim, cannot be used by ProTool objects as symbolic names. In the symbolic names of VBScript tags (local tags) and procedures, no special characters or language dependent characters, such as ä or é, can be used. All names beginning with `pt` are reserved for internal ProTool objects.

If you cannot avoid using the names of VBScript constructs for the symbolic names, you must give the names object-specific prefixes. These are:

- In the case of ProTool tags (global tags), the prefix `PtVar`
- In the case of ProTool functions, the prefix `PtFct`
- In the case of ProTool screen names, the prefix `PtScn`
- In the case of ProTool archive names, the prefix `PtArc`
- In the case of ProTool script names, the prefix `PtScp`

Example of using tags: `PtVar.Convert`

### Global tags

In ProTool tag names (global tags), all characters except quotation marks are allowed. If you want to use ProTool tags in a script, you must always use the form `PtTag."Tag_Name"`. If you insert a ProTool tag from the "Select Object" dialog box in the script editor, it will be formatted automatically.

### Processing multiple scripts

Only one script can ever be executed at any one time in online operation. If there are several scripts due to be executed, they are arranged in a queue. If another script is called in a script, the parent script must wait until the script within it is completed.

---

#### Note

Do not configure too many scripts that have to be processed simultaneously, since this can result in too severe constraints at run time. If a maximum number of simultaneous or nested scripts is exceeded, a corresponding error message is issued by ProTool during compilation.

---

**No value assignment to script parameters**

Value assignments cannot be made to script parameters in a script. If you pass a parameter to a script, the value of the parameter is used in the script (call by value). However, processing in the script does not change the input parameter.

The result of a script can be returned in two ways:

- The script is implemented as a function. The result is returned by means of the value assignment to the function name of the script.
- The result of the script is assigned explicitly to a ProTool tag.

**Unsuitable functions**

The VBScript functions "MsgBox" and "InputBox" cannot be used within ProTool.

## 5.14.5 Examples of creating VBScripts

These examples show a subroutine and a function of Visual Basic scripts.

### 5.14.5.1 Example of a subroutine

#### Objective

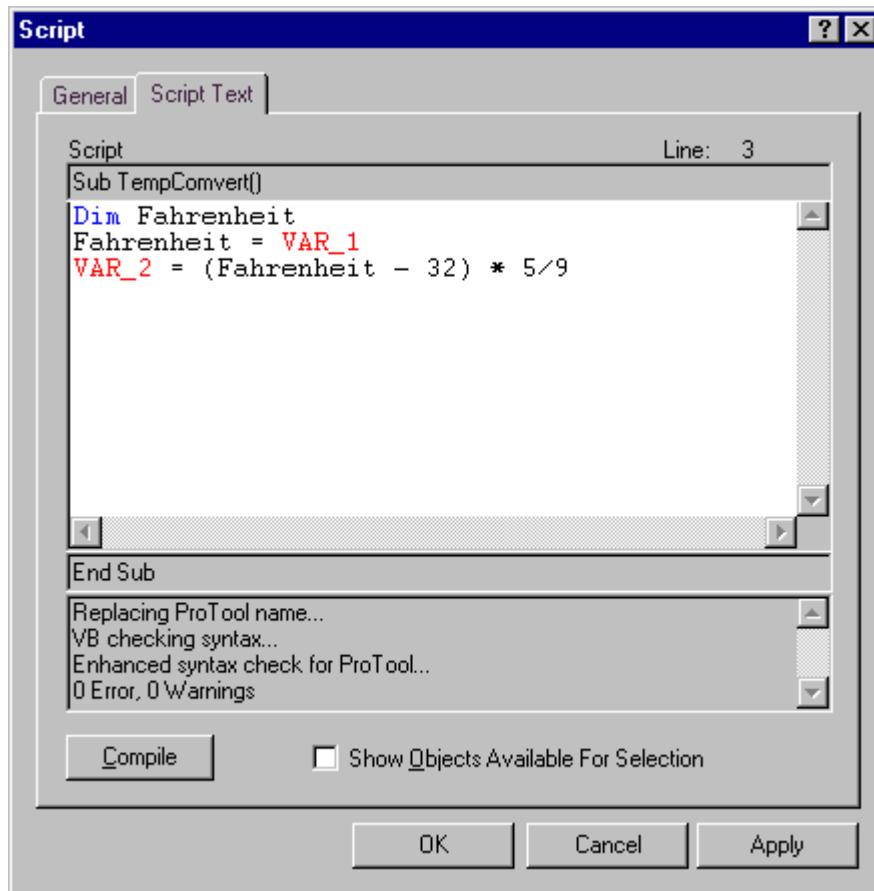
A value in Fahrenheit is read from the PLC using ProTool tag `Var_1`. The value is converted to Celsius by means of the `TempConvert` script and written to the internal ProTool tag `Var_2`. You can associate tag `Var_2` with an output field and display it on a screen.

#### Creating the script

Procedure:	
1	First create in ProTool the tag <code>Var_1</code> with connection to the PLC and select "Read continuously" on the "Options" tab.
2	Configure tag <code>Var_2</code> without a connection to the PLC (<"no PLC>").
3	Call "Scripts" from the process window.
4	Select the "General" tab and then the "Sub" option.
5	Enter the following script name at "Name": <code>TempConvert</code> .
6	At "Description", enter a short description of the function, such as <code>Conversion of Fahrenheit to Celsius</code> .
7	Select the Script Text tab.
8	Then edit the script. <pre>Dim Fahrenheit Fahrenheit = Var_1 Var_2 = (Fahrenheit - 32) * 5 / 9</pre>
9	Click the "Compile" button to compile the script.

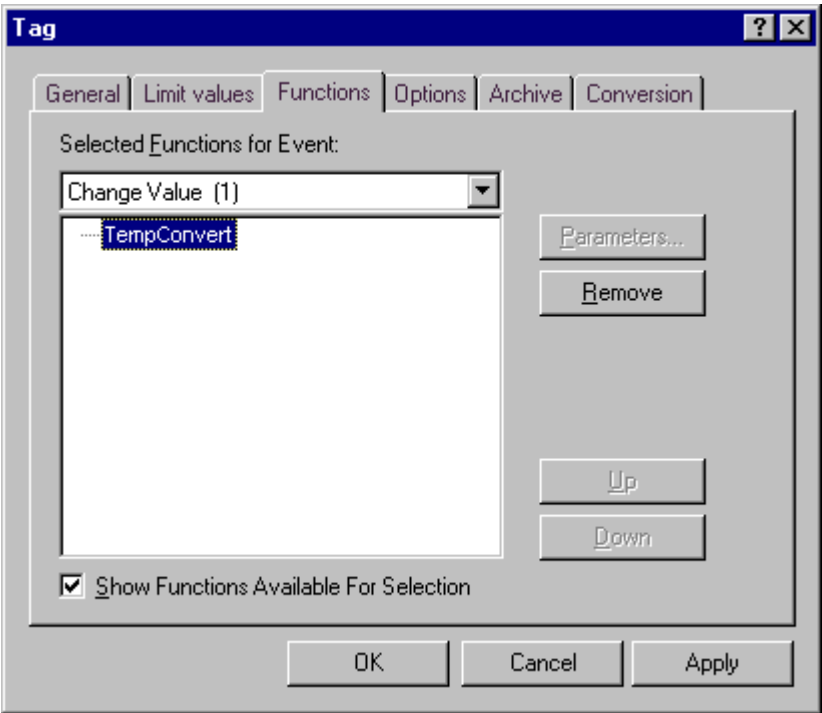


The figure below shows you the "Script Text" dialog box with the edited and compiled script.



*"Script Text" dialog box*

Using the script

	<p><b>Procedure:</b></p>
1	<p>In the project window, select the tag <code>Var_1</code> and go to the "Functions" tab.</p>
2	<p>Assign the script <code>TempConvert ()</code> to the "Value Change" event by clicking the "Add" button.</p> <p>The figure below shows the "Functions" tab together with the script mounted.</p> 
3	<p>Configure an output field in a screen. Assign the output field the tag <code>Var_2</code>. Every time tag <code>Var_1</code> changes, the script is executed and the result written to tag <code>Var_2</code>. The current value is displayed in the output field.</p>

### 5.14.5.2 Example of a function

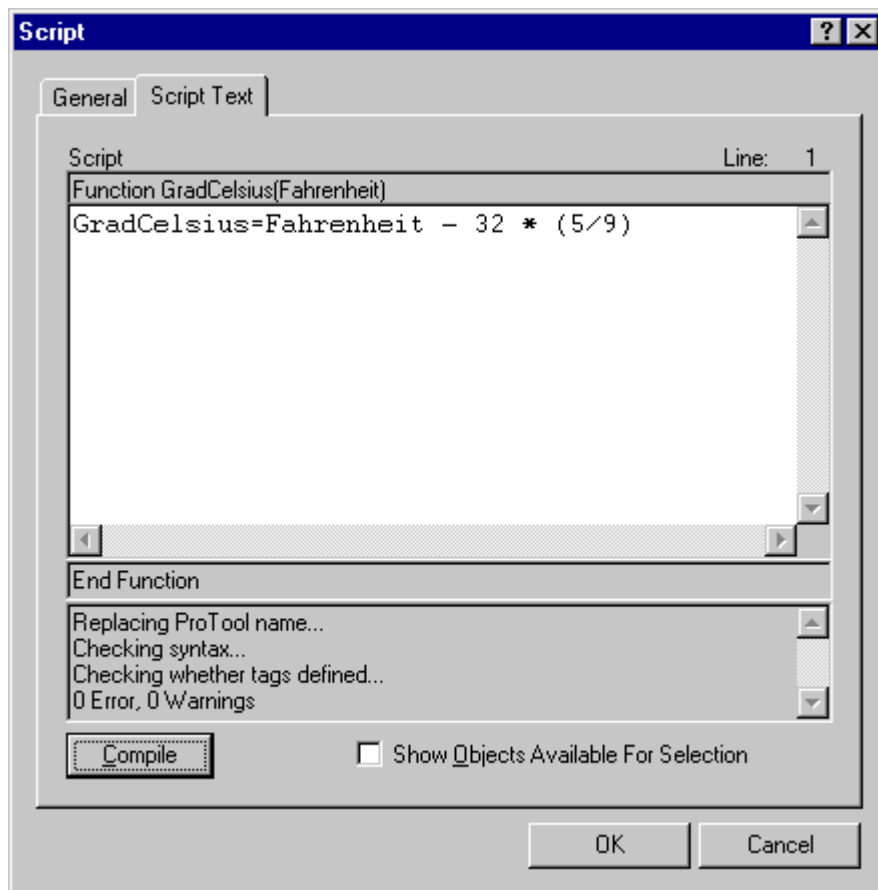
#### Objective

A value in Fahrenheit is read from the PLC using ProTool tag `Var_1`. The value is converted to Celsius by the script `DegreesCelsius(Fahrenheit)` and written to the internal ProTool tag `Var_2`. You can associate tag `Var_2` with an output field and display it on a screen.

#### Creating the script

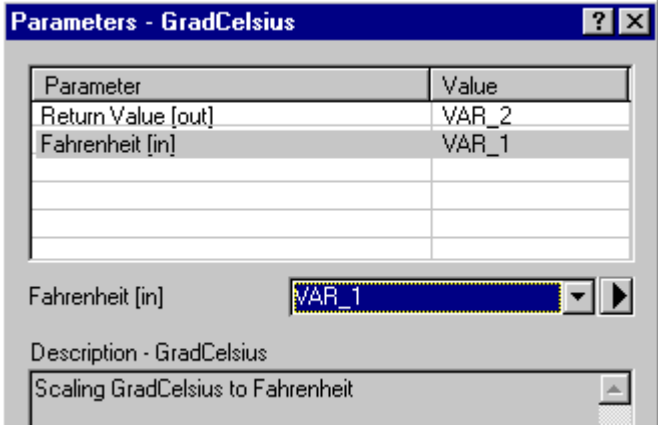
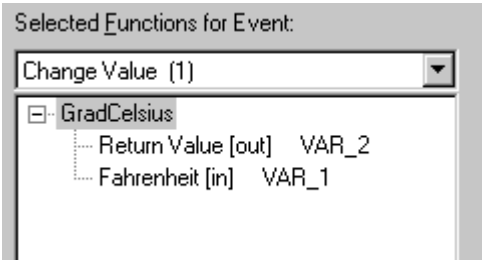
Procedure:	
1	First create in ProTool the tag <code>Var_1</code> with connection to the PLC and select "Read continuously" on the "Options" tab.
2	Configure tag <code>Var_2</code> without a connection to the PLC (<"no PLC>").
3	Call "Scripts" from the process window.
4	Select the "General" tab and then the "Function" option.
5	Enter the following script name at "Name": <code>DegreesCelsius</code> .
6	At "Description", enter a short description of the function, such as Conversion of Fahrenheit to Celsius.
7	Enter <code>Fahrenheit</code> in the parameter field, and click the "Add" button.
8	Select the Script Text tab.
9	Then edit the script. <code>DegreesCelsius = (Fahrenheit - 32) * 5 / 9</code>
10	Click the "Compile" button to compile the script.

The figure below shows you the "Script Text" dialog box with the edited and compiled script.



"Script Text" dialog box

## Using the script

Procedure:	
1	In the project window, select the tag <code>Var_1</code> and go to the "Functions" tab.
2	Assign the script <code>DegreesCelsius</code> to the event "Value Change" by clicking the "Add" button.
3	Specify tag <code>Var_1</code> as the input parameter <code>Fahrenheit</code> (identified by <code>[in]</code> ).
4	Specify the tag <code>Var_2</code> for the return value (identified by <code>[out]</code> ). The figure below shows the parameter assignment. 
5	Click OK to conclude parameter input. The figure below shows the "Functions" tab together with the script mounted. 
6	Configure an output field in a screen. Assign the output field the tag <code>Var_2</code> . Every time tag <code>Var_1</code> changes, the script is executed and the result written to tag <code>Var_2</code> . The current value is displayed in the output field.

## 5.15 Configuring in foreign languages

This chapter shows you what you need to know in order to create a project with a user interface in a foreign language.

You can create a monolingual or a multilingual project. In a multilingual project you can decide:

- Whether to make several languages available on an OP and provide the user with a key for switching between them.
- Whether to download only one language to a specific OP.

### 5.15.1 System requirements for foreign languages

#### Introduction

Depending on your Windows language, you can configure nearly all foreign languages without having to perform changes to your Windows configuration.

The exceptions are languages with special fonts such as Greek, Polish, Russian, Slovenian, Czech and Hungarian. To be able to use these languages, you have to enable language support under Windows 98 SE ("Control Panel" > "Software" > "Windows Setup" > "Language Support" > "Details"). This is not necessary in Windows NT.

---

#### Note

If the runtime software is running on a different computer from the configuration software - in other words, if the computer used for operating is not the same one that is used for configuring, you may have to reload the language support on both computers. Alternatively, you can install Windows completely in one of these languages.

---

---

#### Note

The ProTool installation CD-ROM contains support for other languages not supplied as standard with Windows such as Albanian, Bulgarian, Romanian, Turkish, etc. in the directory WINLANGS. More detailed information is provided in the file MULLANGO . INF.

---

To create projects in Asian languages, you require an Asian Windows system. You will find detailed information in "Requirements for configuring in Asian Languages".

## 5.15.2 Languages in ProTool

### User interface language and project languages

ProTool makes a distinction between two language levels:

- The "user interface language" in ProTool This is the language in which the menu and dialog box text blocks are displayed when you configure in ProTool/Pro CS. The user interface language is selected from ProTool's Setup.
- The "project languages" These are the languages in which the blocks of text in your project appear on the operating unit. You can create the project in any language available in Windows on the configuration computer.

The two language levels are completely independent of each other. For example, you can create French projects with an English ProTool, or English projects with a French ProTool.

### Multi-lingual projects

If you want your project to run in different languages, store all language-dependent objects in the desired languages in the project. This applies, for example, to:

- Event messages
- Alarm messages
- Screens
- Recipes
- Text lists
- Help texts
- Reports

You can store text for any project in 32 project languages. In doing so, you can select any 32 languages from all the languages supported by the operating system installed on the configuration computer. When creating your configuration, however, note that the languages supported at run time depend on the operating unit.

You cannot edit system messages in ProTool. They have been stored for every operating unit for a number of languages already. If you are configuring a language for which no system messages are available, all the system messages will appear in English.

### Editing language

The project language in which you edit text at any given time on the configuration computer is the "editing language".

The toolbar and the status bar show the current editing language setting.

## Reference language

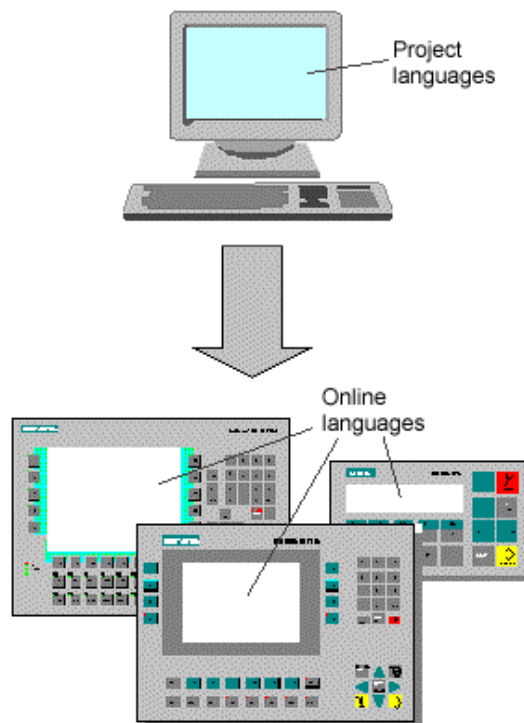
One of the project languages can be used as the "reference language". You can first create all the blocks of text in the Reference language and then use them as a basis for other translations when you enter your translated text directly into the project.

When you are configuring, you can toggle between the reference language and the editing language you have set.

## Online languages on the operating unit

You can even download several of these project languages simultaneously to the operating unit, depending on the operating unit. For the operator to be able to toggle between these languages, you must provide him with a suitable control.

When creating your configuration, note that the languages supported at run time depend on the operating unit.



*Language levels in ProTool*

## Exchanging text with translators

If you wish to send the blocks of text in the reference language for translation, you can export all blocks of text in a project from ProTool. The translator can edit the blocks of text outside ProTool, either in Excel, for example, or a text editor. The translated blocks of text are then imported into your ProTool project and automatically inserted at the correct places.



### 5.15.3 System message languages

#### System messages

You cannot edit system messages in ProTool. They have been stored for every operating unit for a number of languages already. If you are configuring a language for which no system messages are available, all the system messages will appear in English.

#### Languages

System messages for your operating unit are available in the following languages:

- Chinese (PRC)
- Chinese (Taiwan)
- Danish
- German
- English
- Finnish
- Flemish
- French
- Greek
- Italian
- Korean
- Norwegian
- Polish
- Portuguese
- Russian
- Swedish
- Spanish
- Czech
- Turkish
- Hungarian
- Japanese

## 5.15.4 Language dependent keyboard assignment

### Language dependence

The characters on a PC keyboard are language dependent. For example, no German or French special characters are available on an English keyboard. Further, the letters are arranged somewhat differently.

As soon as you change the editing language when you are configuring, ProTool modifies the assignment of your keyboard to the layout of the foreign language concerned.

### On-screen keyboard

In order to facilitate the assignment of the characters printed on your keyboard and those actually entered, ProTool displays a language-dependent on-screen keyboard. You can see on the on-screen keyboard where the deviating characters are located.

There are two options to choose from for entering a language-specific character:

- Orientate yourself on the on-screen keyboard and enter the character on the keyboard of the configuration computer.
- Click your mouse to enter the character directly via the on-screen keyboard.



*Example of language-dependent on-screen keyboard with French as the editing language*

The on-screen keyboard is hidden automatically as soon as you reset the editing language back to the current Windows language. You can also enable and disable display of the on-screen keyboard by choosing "View" > "Keyboard".

You can also directly start the on-screen keyboard on the operating unit as a stand-alone application. The following parameters are available for the executable file "TouchInputPC.exe":

- "NumPadOff" starts the alphanumeric keyboard (default)
- "NumPadOn" starts the numeric keyboard
- "NoInfo" starts the keyboard without Copyright notice
- "Close" closes the open keyboard

## Sizing

You can size the on-screen keyboard displayed on PC systems:

1. Double-click on the keyboard icon on the taskbar. This displays the "ProTool Input Panel Options" dialog box.
2. Select the "Start Resize" button.
3. Drag the on-screen keyboard by the Resize icon to the size you require.
4. Select the "Stop Resize" button.
5. To save the setting you have just performed, click the "Save" button.

---

### Note

You can only use the on-screen keyboard if the corresponding keyboard layout has been installed under Windows. Enable the keyboard at "Start" > "Settings" > "Control Panel" > "Keyboard". If you have not done it, ProTool will resort to the international (US) layout.

---

---

### Note

Compound Korean characters can only be written by key combinations (such as a, b, c) is only possible on the on-screen keyboard in the following operating systems:

- Windows 2000
- Windows NT

---

---

### Note

If the operating system on a Panel PC is set in a European language, data record names cannot be displayed with Asian characters.

---

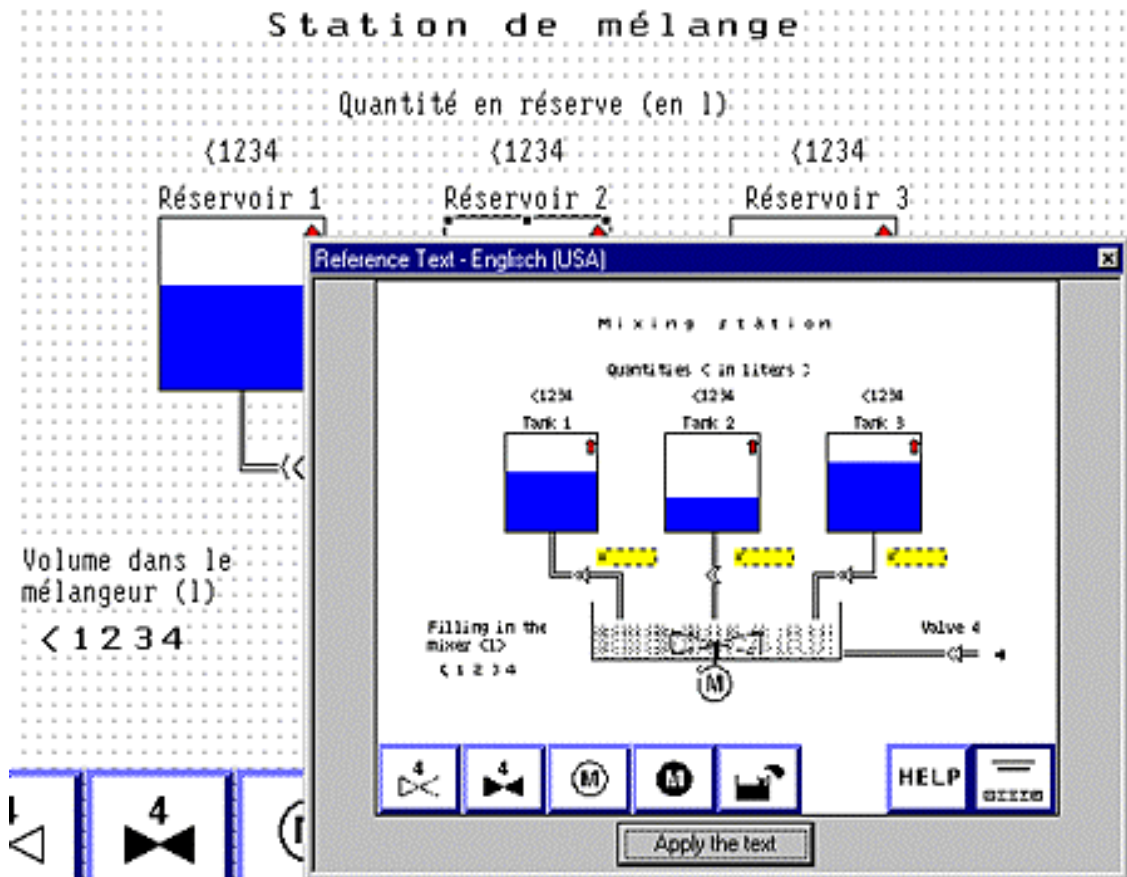
### 5.15.5 Reference text

#### Principle

When you create a project for several languages, you normally configure all the pieces of text in your native tongue first.

If you then change the editing language to enter text in a foreign language, all the text fields appear blank again.

ProTool features a user-friendly reference text function so that you have a basis for your translations. In dialog boxes and in screen editors, you can display an additional window with blocks of reference text in the original language.



Reference text

In the editor you can insert the translation of the blocks of text in the reference window text without having to switch from one language to another.

---

**Note**

Text can be exported by choosing "Edit" > "Language" > "Export" from the menu or by using the shortcut menu in the project window. The blocks of text can be translated independently of ProTool in this way, too, and then imported in your project.

---

## 5.15.6 Steps to creating a multilingual project

### Scenarios

The basic approach is identical no matter whether:

- You configure a project in a language other than that of your ProTool installation. (Example: you have ProTool in German and would like to create a French project.)
- You want to sell a project to several countries, in the language of the countries concerned. (Example: you are sending a machine to Germany, England and France. You want all pieces of text to be displayed on the operating unit in the language of the countries concerned.)
- You supply a project to a multilingual country. (Example: you are selling a machine to Switzerland. You want the operator to be able to choose between German, French and Italian on the operating unit.)

In any case, you should always attempt to create and test the project in one language first. This language then acts as your reference language for the translations.

### Steps

---

#### Note

You can access the "Set Language" dialog box in two different ways: by choosing "Edit" > "Languages" > "Select" or "System" > "Assign Language" from the menu. The option used below is always "Edit" > "Languages" > "Select".

---

Configuring in foreign languages consists of the following specific work steps:

1. Satisfy the necessary system requirements.
2. Define the configured languages (choose "Edit" > "Language" > "Select" from the menu).
3. Select a language as the first editing language (choose "Edit" > "Language" > "Select" from the menu). Create and test the complete project in this language first, before proceeding to translate all the pieces of text together later .
4. Set the font (choose "System" > "Fonts" from the menu).
5. Configure a language switch (only if you want to offer several languages simultaneously on one operating unit).

6. Translate the pieces of text.  
If you want to enter blocks of text in a foreign language, switch the "editing language" (choose "Edit" > "Language" > "Select" from the menu). You select the language in which you have created the project to be your "reference language" (likewise option "Edit" > "Language" > "Select"). You can now have reference language text displayed and enter the text in the editing language. If you want to send text for translation, you can export it in csv format (choose "Edit" > "Language" > "Export" on the menu). After being translated external to ProTool, you can accept them into your project (choose "Edit" > "Language" > "Import" from the menu).
7. Select the languages to be downloaded to a specific operating unit (choose "System" > "Language Assignment" from the menu). You can select up to five languages simultaneously.
8. Compile the project.
9. Download the project to the operating unit.

---

**Note**

Avoid moving fields in event messages and alarm messages when you modify a project that has been created in several languages. Since there is no permanent assignment between the field and its position within the text, you should move the pieces of text – if necessary – instead of the fields.

---

## 5.15.7 Exchanging text with translators

### Editing blocks of text external to ProTool

If you configure a project in several languages, the simplest approach is to create and test the complete project in one language first.

You can then pass all the blocks of text to translation and, after that, include the translated text in your project.

The advantages of this procedure are as follows:

- The workload of the configuration engineer is reduced, since he only has to create the user interface in a language familiar to him.
- The translator can edit the blocks of text outside ProTool, either in Excel or a text editor.
- The translated blocks of text will be complete and consistent.

### Scope of text export/import

Using the ProTool's text export/import function, you can export and import blocks of text for the whole project, text from specific editors or text of separate objects. The scope of export/import depends on the command:

- Export/import blocks of text for the complete project. When you issue a command to export/import text by choosing "Edit" > "Language" > "Export" or "Import" on the menu, all blocks of text for the complete project are exported or imported. The import operation cannot be undone. But you can remove all blocks of text in a language from the project by deleting that language. You will find detailed information by referring to "Deleting a project language".
- Export/import blocks of text of a certain editor or object. You can start exporting/importing text in the project window by means of the shortcut menu (right-click) of the editors and objects. Only those blocks of text can be exported or imported that belong to the selected editor or object. A text import operation started from the shortcut menu can be undone.

### Exporting different text types

When exporting, you can select the types of text you would like to export:

- Text visible on the screen (such as labeling, names) It is stored in the file called "Labeling.csv".
- Info text (such as Help text for screen objects and messages) It is stored in the file called "InfoTexts.csv".
- Message text (event messages and alarm messages) It is stored in the file called "MessageTexts.csv".
- Other text (message settings, text lists) It is stored in the file called "OtherTexts.csv".

The default setting is export of all text types.



The classification of the text types is done in accordance with the requirements that have to be taken into account in translation. For example, the placeholders have to be preserved in message text, and when text visible on the screen is translated it might be necessary to take the limited

### Importing text

You can insert translated blocks of text in your project by choosing "Edit" > "Language" > "Import" or by using the shortcut menu of the editors and objects in the project window. The individual blocks of text are automatically assigned to the objects in the project.

---

#### Note

You can undo an import of alarm message text.

---

To view imported text in foreign languages, set the editing language to the language concerned.

- If you did not specify a target language for the translation when you exported the text, you must specify the language into which the text was translated when you import it.
- If this language is not available as a "configured language" in your project, it is created automatically.

### Constraint with importing/importing text with Cyrillic, Greek or Asian characters

It will depend on the operating system of the configuration computer whether you can export or import text containing characters not in the Western character set (such as Greek, Cyrillic or Asian characters).

- Windows 2000 multilingual A requirement for export and import is that the system standard and the system character set is set on the configuration computer to the desired language (Windows Start menu > Settings > Control Panel > Regional Options > input locale for the system, "Set default" button). With this setting you can export and import text with non-Latin characters without any difficulty.
- Windows98, Windows NT, Windows ME Non-problematic export and import is possible only with a localized operating system, meaning that you can export and import text in Cyrillic characters only with a Russian operating system. If you do not have a localized operating system, you can configure text with non-Latin characters in ProTool but you cannot export or import it. Non-Latin characters are displayed a question marks when text is saved. If you import text in a csv file ProTool with non-Latin characters in ProTool, these characters are imported as question marks.

---

#### Note

Remember not to change or add text manually in the time between exporting and importing the text but to have these changes translated in good time. Manually changed or added text will be overwritten by the text in the import file on being imported.

---

## 5.15.8 Requirements for configuring in Asian languages

### Asian Windows system

A requirement for creating a project with Asian blocks of text is an Asian Windows system or Windows 2000 multilingual as the operating system. The necessary character sets can only be accessed in these Windows systems.

Even the compilation must be carried out on an Asian Windows system, otherwise the project language will be ignored during compilation.

Under certain conditions, it is possible to download projects in which an Asian project language has been selected, on a non-Asian Windows system.

For this, the \*.fud or \*.fwd file created during compilation on an Asian Windows system has to be available in the same directory as the \*.pdb project file. The \*.pdb project file can only be opened before the download operation, but not modified any more.

---

### Note

Many projects are multilingual and use, for example, English in addition to an Asian language. In such cases you can first create and test your project on a non-Asian Windows system. Not until it is completed do you change to an Asian system, on which you configure and compile the Asian text and perform the download operation.

You can work most efficiently if you have both an Asian and a non-Asian Windows system installed on your computer or if you work on a network with different computers for the languages concerned.

---

## 5.15.9 Constraints on projects in Asian languages

### Supported languages

Chinese (PRC), Chinese (Taiwan), Korean and Japanese are the languages currently supported.

### Configuration memory

Use different character sets as sparingly as possible in a project. Every character set you use reduces the amount of memory available for projects.

64 kB of configuration memory are available per character set, which approximates to 1900 different Chinese characters, for example. If you configure more different Chinese characters, you obtain an error message during compilation to the effect that the maximum number of characters has been exceeded. At the same time, a list of the most infrequently used characters in the project is output.

### Special characters

Special characters are ASCII characters higher than 127 for example, ã, ±, ä, è, ó.

If you use a non-Chinese character set in text in an Asian Windows system, you should not configure any special characters, because they are not displayed correctly in a large number of cases.

---

#### Note

The following applies to Windows CE devices:  
Only West European characters must be entered for Asian languages as titles for message windows, lines and buffers. Other characters are not displayed on Windows CE devices.

---

---

#### Note

If you enter Asian characters in English text in a multi-language project that includes Asian languages, they will not be displayed on these Windows CE devices.

---

### Messages

All system messages are also available in Asian languages.

Message number, message status (active, cleared, acknowledged), date/time and acknowledgement group remain in alphanumeric characters in Asian projects and cannot be displayed in Asian characters.

**Switching between Asian languages under Windows 2000**

With language support installed in Windows 2000, you can switch between the Asian languages offered in order to display a project. You can only enter text in the displayed language, however, when the operating system has also been switched to the desired language.

**Field length**

Field lengths depend on the width of the characters used. With Asian character sets, the number of configurable characters in fields can be reduced as a result of the double character width.

## 6 Testing projects


### Principle

Once you have completed your entire project or self-contained parts within it, the test phase starts.

The following steps must be taken:

1. Compile your project: Your project is transformed into a file that is capable of running on the operating unit.

2. Testing a project without PLC connection: To test the project without a PLC, you must install the simulation program. Start the simulator by choosing

"File" > "Test" > "Start Simulator" from the menu or click this icon: 

Specify suitable values in the simulation table for testing the tags or area pointers. The values in the simulation table are read by Runtime during the simulation and thus a PLC connection is simulated. Testing a project with PLC connection: If you have connected your configuration computer to a PLC and have installed ProTool/Pro RT on the configuration computer, you can test your project directly in ProTool/Pro RT. You then get an exact simulation of the destination device.

3. Choose "File" > "Test" > "Start Runtime" from the menu or click this icon: 

Download the project: The project is copied to the directory from which it will be started. This directory might be on the configuration computer or on the operating unit.

Should you find other errors in the project, troubleshoot them and start again at step 1.

---

#### Note

If you are using a SIMATIC S5 or S7, configure "Status/Force" in order to test specific tags in the PLC program.

---

---

#### Note

If you configure with ProTool integrated in SIMATIC STEP 7, you can simulate a PLC connection with PLCSIM. The S7-PLCSIM cannot be used in conjunction with the PC station of STEP7, however. For further information, please consult your SIMATIC STEP 7 documentation.

---

## 6.1 The Windows CE Control Panel

---

### Note

The functionality available for the Windows CE Control Panel depends on the device. For detailed information about the Windows CE Control Panel please refer to the equipment manual of the operating unit concerned.

---

### Principle

The Windows CE Control Panel is only available on operating units running under the Windows CE operating system.

Using the Windows CE Control Panels service personnel can perform tasks such as the following:

- Perform backup/restore
- Perform communication settings
- Set date and time
- Configure the input panel (on-screen keyboard)
- Configure keyboard and mouse settings
- Configure network settings
- Set the printer
- Format the FLASH memory
- Perform device-specific settings (for example, brightness and calibration of touch panels)
- Perform Download settings
- Set the screen saver

To make it easy to call the Windows CE Control Panel, you can provide a relevant control in your ProTool project. In all other cases, the Control Panel can only be opened from the Start menu while the device is starting up.

We recommend that you protect selection of this control by means of a password so that it is only accessible to service personnel. In this way, you stop an operator from changing the basic settings of the operating unit by mistake.

---

### Note

You can also protect all the controls intended for service purposes, configure them on a separate screen and protect selection of this screen by means of a password.

---

**Example application - Displaying S7 settings**

The Windows CE Control Panel requires the service personnel when a new device, for example, on an MPIbus is integrated into a system to which several devices have already been connected. In this case it is probable that the default MPI address on the new device is probably already being used by another bus node. The MPI address can be displayed and modified, if necessary, by means of the "S7 Settings" applet on the Control Panel of the device.

## 6.2 Downloading the executable project file

### Download options

You can download the project file from the configuration computer to the operating unit as follows:

Download	CE devices	PC systems
Serial	X	X
USB	X	X
Ethernet (TCP/IP)	X	X
Modem	X	X
MPI	X	X
DP	X	X
File transfer	--	X

---

### Note

Some USB cables with network function cannot be used for a direct connection.

---

- Direct connection (USB cable, serial cable)
- Network connection (Ethernet or modem) If the destination device (a PC, for instance) and the configuration computer are located on the same network, you can also download the executable project file over Ethernet by establishing a network connection to the destination device.
- MPI/PROFIBUS DP

Floppy disk download (PC only) In your project, choose "File" > "Download" > "Settings" from the menu and select drive A: \ as the destination directory. The executable project file is then compressed and packed onto one or more disks, which you unpack again on the destination device with the disk download.

### Download mode with Windows CE systems

A requirement is that the operating unit is in "Download mode" or goes automatically to Download mode when the download operation from the configuration computer begins. For this, the corresponding download option must have been selected from the configuration menu of the operating unit. You will find more detailed information about setting Download mode on the operating unit in you equipment manual.



In the case of initial commissioning, the operating unit runs up automatically in Download mode. For every further download, you can reboot the operating unit and call the download applet on the Start menu, or configure the *Change Mode* function, if the operating unit does not go automatically to Download mode.

---

**Note**

If the destination device is a PC which is still without a project, you must enable Download mode manually in the PC loader before the first transfer.

---

**Basic procedure**

1. Set the operating unit to Download mode.
2. Select the memory location for the project file on the operating unit.
3. Perform the Download settings on the configuration computer in ProTool.
4. Download the project file from the configuration computer to the operating unit.

---

**Note**

In the case of Serial Download, you should always select the highest transfer rate possible. This applies in particular to the Backup/Restore function to the transfer of option packs, image files or large configurations (especially for initial commissioning). At low transfer rates, it can easily take hours to transfer the quantities of data.

---

## 6.3 Simulation options

### Requirements

For you to be able to simulate your project on the configuration computer, ProTool/Pro RT or the simulator must be installed, or you transfer the project to a PC on which Runtime is installed.

### Principle

There are several ways in which you can simulate your finished project:

- **Simulation with PLC connection** You can simulate your project by letting it run directly in your Runtime software. Tags and area pointers will only be able to work if your configuration computer has a connection to a suitable PLC. If your computer is connected to a PLC, you will achieve authentic simulation of your configured operating unit with the Runtime software. To simulate with ProTool/Pro RT, choose "File" > "Test" > "Start Runtime" from the menu.
- **Simulation without PLC connection** Using the simulation program installed with ProTool/Pro RT, you can simulate your project including tags and area pointers without a connection to a PLC. You enter the parameters of area pointers and tags in a simulation table, which is read during simulation by ProTool/Pro RT. To simulate with the simulator, choose "File" > "Test" > "Start Simulator" from the menu.
- **Simulation in integrated mode** If you configure with ProTool integrated in SIMATIC STEP 7, you can simulate a PLC connection with PLCSIM. For more precise details, please consult your SIMATIC STEP 7 documentation.

## 6.4 Simulation of the project with the simulator

### Purpose



Using the simulation program, you can simulate your project on the configuration computer without a PLC and without an operating unit. The simulation program is a stand-alone program that is installed together with the runtime software. If you have not installed the Runtime software, you must install the simulation program separately from the ProTool installation CD-ROM.

You can simulate the following parts of the project with the simulator:

- Tags
- Area pointers, which are organized bit by bit: event messages and alarm messages.

### Basic procedure

The simulation program parameterizes the tags and/or area pointers that have to be simulated and activates them.

	<b>Procedure:</b>
1	Create a project first, in the way you want it to run later with a connected PLC.
2	Save and compile your project.
3	 <p>Start the simulator directly from the running configuration software by clicking the illustrated symbol or by choosing "File" &gt; "Test" &gt; "Start Simulator" from the menu.</p> <p>If you start the simulator without compiling the project beforehand, compilation is initiated automatically.</p> <p>When you simulate the project for the first time, the simulator starts with a new, blank simulation table. If you have already created a simulation table for this project, this is opened.</p>
4	<p>Enter your project-specific settings in the simulation tables and save them for later simulation operations. Choose "File &gt; "Save" on the simulator and specify a file name for the simulation table (*.sim).</p> <p>In this way you can open this file at a later point of time and simulate your project once again.</p>
5	Enter the parameters for tags and area pointers of your project in the simulation table.
6	 <p>Start the Runtime software using the icon illustrated or by choosing "File" &gt; "Test" &gt; "Start Runtime" from the menu.</p> <p>Switch tasks from the simulator to the current project if you wish to observe the variation in the value.</p>

---

**Note**

When simulating configurations for Windows CE devices, not all functions can be simulated, such as contrast settings. Scripts can be simulated but their behavior might be exactly the same as that on the real device.

It must similarly be remembered that the run-time behavior in the simulation is not the same as the run-time behavior on the actual destination device.

---

**Simulation table**

Tag	Data Type	Current va	Format	Write cyle	Simulation	Set value	MinValue	MaxValue	Cycle	Start	
VAR_1	UINT	0000 0000	C Bin	1,0	Sine		0000 0000	(1111 1111	10,000	<input type="checkbox"/>	
VAR_2	INT	0000 0000	C Bin	4,0	Random		1000 0000	(0111 1111		<input type="checkbox"/>	
VAR_3	INT	0	Dez	2,0	Increment		-32768	32767	10,000	<input type="checkbox"/>	
VAR_4	REAL	0	Dez	10,0	<Display>					<input type="checkbox"/>	
VAR_5	INT	0000	Hex	2,0	Shift bit					<input type="checkbox"/>	
VAR_7	INT	0000	Hex	4,0	Decrement		8000	7FFF	10,000	<input type="checkbox"/>	
Event mes	UINT	0000 0000	C Bin	1,0	Shift bit					<input type="checkbox"/>	
Alarm mes	UINT	0000 0000	C Bin	1,0	Shift bit	0000 0000 0				<input checked="" type="checkbox"/>	
* ---										<input type="checkbox"/>	

*Example of a simulation table*

All settings that you perform in this table for the simulation of your project can be saved in a file. For this, choose "File > "Save" on the simulator and specify a file name (\*.sim).

In this way you can simulate your project once again at a later point of time using the saved settings. A condition is that you will not have changed the tags and area pointers to be simulated in your project in the meantime.

**Tags**
**Featured tags**

Not all configured tags are featured in the simulation table. Only referenced tags are available for simulation, meaning only the tags that are used in your project in a screen object, for example.

**Archive tags**

To be able to simulate archive tags, select the "Read Continuously" option while configuring the tags in the Tag dialog box on the Options tab.

**Tags in messages**

Tag objects are not created for tags that only occur in messages. These tags are therefore not offered in the simulator tag list. Therefore, activate the option "Read continuously" at least during the test phase for the tags you want to use in the simulator.

## Data formats

### PLC-specific data formats

Since the simulator can only simulate the project in Offline mode – in other words, without a connection to the PLC – the data formats are converted to internal ProTool/Pro CS formats. This means that PLC-specific data formats cannot be implemented.

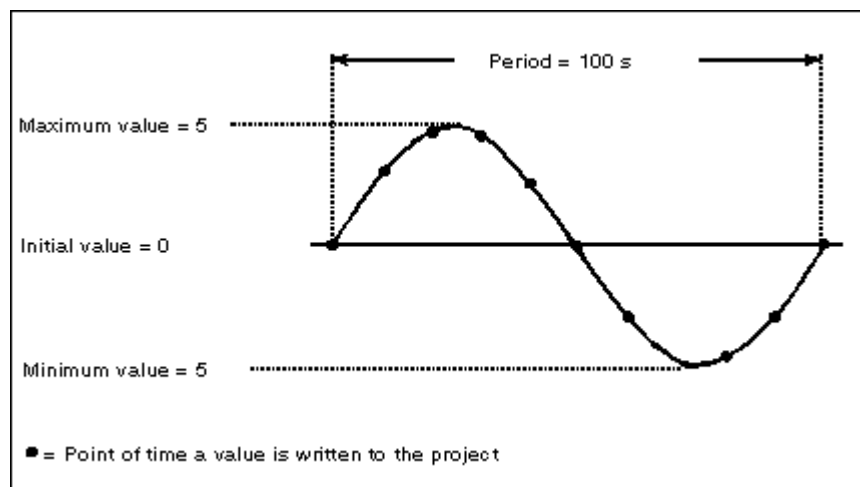
### Date and time

The simulator converts data formats for date and time, such as DATE, DATE AND TIME and TIME OF DAY for example, to an internal format that contains both the time and the date. If you only enter the time for a tag such as this, for example, the missing component – such as the date – can likewise be modified. During the simulation of tags with time- or date-dependent format, therefore, enter the complete value with date and time, as necessary.

## Example of a simulation

The example in the figure below shows you, on the basis of a sine curve, how the values of a tag are written to the project in accordance with the settings in the simulator. The following settings have been performed in the simulator in this context:

- Set value = 0
- MaxValue = 5
- MinValue = -5
- Write cycle = 10 \* 1s
- Cycle = 100 \* 1s



## More information

You will find more information on running the simulation program in its online Help and in the "ProTool/Pro Runtime User's Guide".

## 6.5 Uploading projects

### Constraint

Projects cannot be uploaded in the case of the TP 170A.

### Usage

As a result of a download operation, the executable project is normally downloaded to the operating unit. If you want to use the original project data for developing the project further or for error analysis, they must remain on the configuration device.

On Windows CE devices (only with external storage medium), you can store not only the compiled project but also the source code of the project so that subsequent retrieval (uploading) of the project from the operating unit is possible.

#### **Advantage:**

In this way you can later perform analyses and changes to an existing project even if the original programming device cannot be reached or the source file (\*.pdb) for the project is no longer available on the configuration computer.

### Requirements

- Sufficient storage space must be available on the operating unit.
- The storage location for the source file must have been set on the operating unit. Different storage locations can be set for the project file and source file: internal flash memory or PC card (refer to equipment manual).
- The project file must be downloaded with the source file.

## What happens during download and upload operations?

When the source file is transferred during a Download operation, the project is compressed from the source format (\*.pdb) and transferred as a \*.pdz file to the external storage medium of the operating unit or directly to the PC. Following uploading, the file is decompressed in ProTool. You have to give the new project a name on the configuration device.

---

### Note

The uploaded and decompressed project file (\*.pdb) can only be opened in a version ProTool whose version number is higher than or equal to the version number of the program with which the project was created.

---

---

### Caution

ProTool has no way of checking whether the source file on the operating unit really does belong to the project operated on it. If you have performed a download without simultaneously downloading the source file at some time or other, old project data may still be on the operating unit. It is possible that the data no longer belong to the current project.

---

## Upload

If you wish to download your configuration with the "Enable upload" option, the configuration computer also transfers the compressed source file (\*.pdz) to the external memory card of the operating unit. If there is no memory card on the external device, the download operation is terminated. All data relevant to the project are transferred before this happens, however, so that Runtime can be started with the downloaded project data.

---

### Note

For Windows-based devices, a memory card is required for storing the \*.pdz file.

---

You should use this option preferably for small and medium-sized configurations in order to keep transfer time short.

There is the following option for large project files:

Transfer the project file as a condensed \*.arj file with the Backup function of Project Manager to CF card, for example. Before you do this, switch your project to Stand-alone mode, as necessary, if you are working with ProTool integrated in STEP7. To do this, operate the "Switch ProTool integration in STEP7" button in ProTool Setup.

## 6.6 Directly accessing values on the PLC (Status/Force tag)

### Purpose

At run time you can have direct access to the connected PLC and I/O from the operating unit to read and write values.

This means you can easily use the operating unit to monitor and control the operands of the PLC program without also having to connect a programming device or PC to the PLC.

This is very advantageous particularly during the testing and commissioning phase of your project.

### Requirements

In order to be able to access values on the PLC directly at runtime, the following requirements must be met:

- The connected PLC must be a SIMATIC S5 or SIMATIC S7.
- You must have configured Status/Force on a screen.

### Status/Force Tag

The figure shows an example of "Status/Force" on the operating unit:

Connection	Type	DB Number	Offset	Bit	Data Type	Format	Status Value	Control Value
PLC_1	E		0		CHAR	BIN	0000 0...	
PLC_1	DB	10	10		WORD	DEC	42994	
PLC_1	M		25	0	BOOL	HEX	0	
PLC_1	A		34		WOR	DEC		

*Example of Status/Force with tags*

"Status" In the "status value" column you can read the current values of the displayed operands from the PLC.

"Force" In the "force value" column, enter a value which will be written into the operand address.



## 6.7 Troubleshooting

If error messages are issued in the course of working with your project, such as when you are compiling or downloading your project, you will learn here how you take remedial action.

### 6.7.1 Error messages during compilation

#### Possible messages

While the ProTool compiler is compiling a project, status messages on the progress of the compilation process are output to a window.

You can seek help with the different messages by positioning the mouse pointer in the corresponding message line and pressing F1.

The following messages make a check of your project necessary:

- **Notes:** It is possible that logical operations in the project are either not correct or do not exist. Check the project for logical relationships in accordance with the messages. The compilation process has been properly completed.
- **Warnings:** The project contains errors that might result in incomplete display on the operating unit. The full compilation process has been completed.
- **Errors:** Correct the errors listed and recompile the project. The compilation process is not completed.

## 6.7.2 Error messages while downloading

### Possible problems while downloading

While a project is being downloaded to the operating unit, status messages on the progress of the compilation process are output to a window.

If problems occur while the project is being downloaded to the operating unit, the cause in the majority of cases is too high a baud rate or the simultaneous use of other drivers on the configuration computer for example, when connected to a network.

The most common causes of discontinuation of the download process are listed below.

### Downloading is aborted after a Line Error message

Possible Cause	Action
The connecting cable is too long.	Select a lower baud rate.
The connecting cable is poorly shielded or spurious signals are imported onto the cable by other devices.	Select a lower baud rate.
Computer performance is inadequate.	Select a lower baud rate.
Another driver is running in the background for example, with networking.	Select a lower baud rate.

### Downloading stops and crashes

Possible Cause	Action
Computer performance is inadequate.	Select a lower baud rate.

### The transfer aborts and a message reporting a compatibility contention is issued

Possible Cause	Action
Contention between the versions of the used configuration software and the operating system of the operating unit.	Synchronize the operating system of the operating unit with the version of the configuration software you are using. The files for updating the operating unit will be found on the ProTool installation CD-ROM in the Images directory.

**Note**

You can perform an operating system update with the ProSave service application. The procedure for updating a device operating system can be found in the ProSave documentation.

**Downloading does not take place**

Possible Cause	Action
Connection to the operating unit cannot be established (message in the system message window).	Check the physical connection between the configuration computer and the operating unit. The operating unit must also be in Download mode.
The standard communication driver is not entered in the Windows device manager.	Check the device status of the COM port in the Properties window of the device manager.
Interrupt problems. Various modules are using the same interrupt.	Modify the configuration of your computer or install ProTool on a stand-alone computer.
Other programs block the interface for example, SIMATIC STEP 5.	Exit from the programs and reboot your computer, if necessary.

**MPI/DP Download does not work**

Possible Cause	Action
For the CP which has to be used for the Download operation the "configured mode" is set (such as when using the SIMATIC NET CD 7/2001).	Go to the "Configuration Console" tool and set "PG operation" for the CP. Check the network parameters (baud rate, MPI address). Perform the download from ProTool CS. Reset the "configured mode" for the CP.
On the PU/PC Panel, the S7ONLINE access point is not on a physical device (CP5611 (MPI), for example). This might be the case with an installed SIMATIC NET CD 7/2001, for example.	Go to the "PU/PC Panel" tool or, if you are using the SIMATIC NET CD 7/2001, to the "Configuration Console" tool and set the S7ONLINE access point to the device you require. Check the network parameters (baud rate, MPI address). Perform the download from ProTool CS. Reset the S7ONLINE access point to the original device.



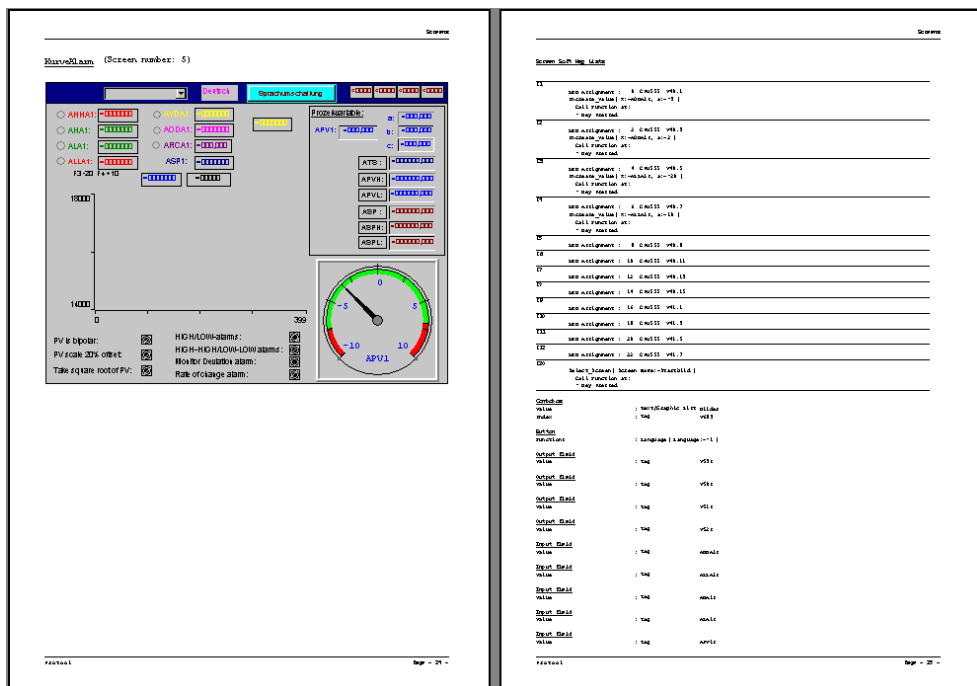
## 7 Documenting and managing projects

This chapter details the multiplicity of functions offered by ProTool for printing out, documenting and managing and archiving your project data.

### 7.1 Printing project data

#### Print function

ProTool provides a comprehensive printout function that can be accessed by choosing "File" > "Print" from the menu. It provides detailed lists for documenting all project-relevant data such as screens, messages, tags, symbol tables, etc.



*Print function in ProTool*

In this way it provides the facility for documenting your complete project. An up-to-date printout can also be very helpful while you are configuring.

#### Note

When configuring, as an alternative to the printout function, there is also the user-friendly cross-reference function offered by ProTool.

## Structuring the printout

ProTool subdivides printouts into chapters arranged by subject according to object types. For example, one chapter contains all the definitions on the subject of screens, another contains a list of all the tags and yet another a list of all the defined text or graphic lists.

You can print several chapters at once or just print single chapters.

## Reports

ProTool offers you the facility of customizing printouts to suit your individual requirements.

- You can limit a printout to single chapters or single pages.
- You can set the order in which the chapters appear.
- You can decide which data you want to output within a chapter.
- You can set margins, define your own headers and footers and embed your own graphic in the cover.

These definitions are stored in a report. Frequently required reports have been defined in ProTool in advance. But you can also create your own reports, as you wish. All reports are common to all projects.

Every time you want to print, you choose the report with which you would like your output to conform.

## 7.2 Project management with integrated operation

### ProTool integrated in SIMATIC STEP 7

If you are working with ProTool integrated in SIMATIC STEP 7, use SIMATIC Manager to manage your projects. You can then copy, move, back up and restore your projects in the same way as you have been used to from SIMATIC STEP 7. For further information refer to the documentation on SIMATIC Manager.

---

**Note**

ProTool's Project Manager is not available to you in the event of integrated operation. ProTool data can no longer be viewed independently in this event, since the data is always linked to a STEP 7 project. It therefore has to be managed and backed up using this application.

---

## 7.3 Managing projects in stand-alone operation

### Project Manager

If you installed ProTool as a stand-alone application – in other words, if you are not operating it under SIMATIC STEP 7 – there is a user-friendly project manager integrated in ProTool instead of SIMATIC Manager. You can use it to manage your projects in a user-friendly way.

### Usage

With Project Manager, it is simple for you to:

- Archive projects (Backup), using more than one floppy disk too
- Restore archived projects (Restore)
- Open projects
- Delete projects

### Backup and restore

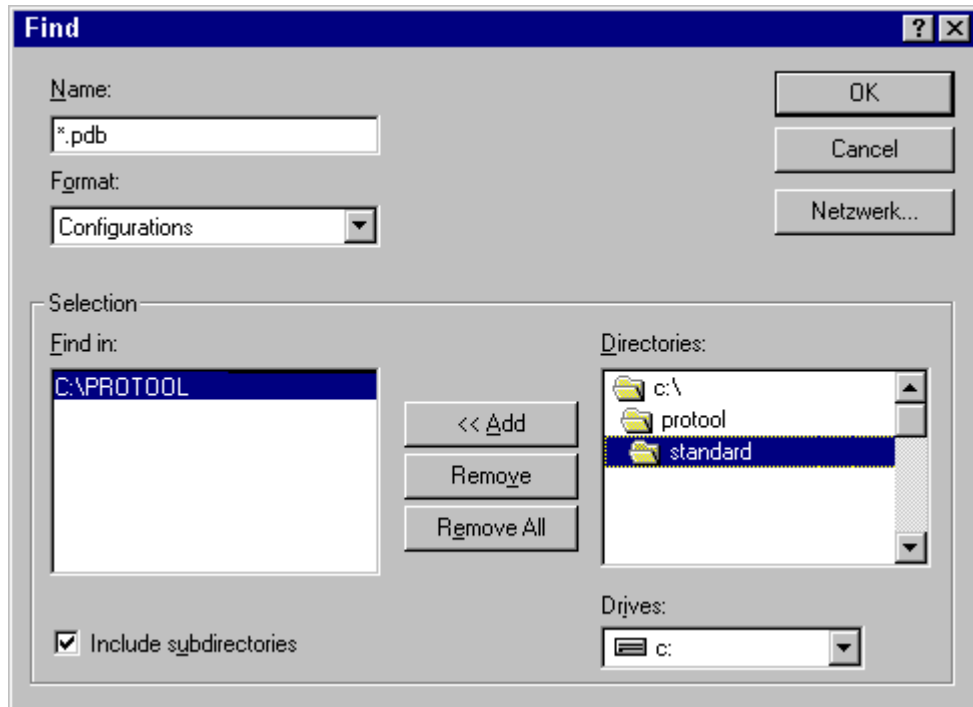
The Backup and Restore functions are not possible between devices of different types. A backed up version of a MP 270 project can therefore only be restored on an MP 270.

The Backup and Restore functions are also available for your projects on the operating units. Refer to your equipment manual for further information.



**Call**

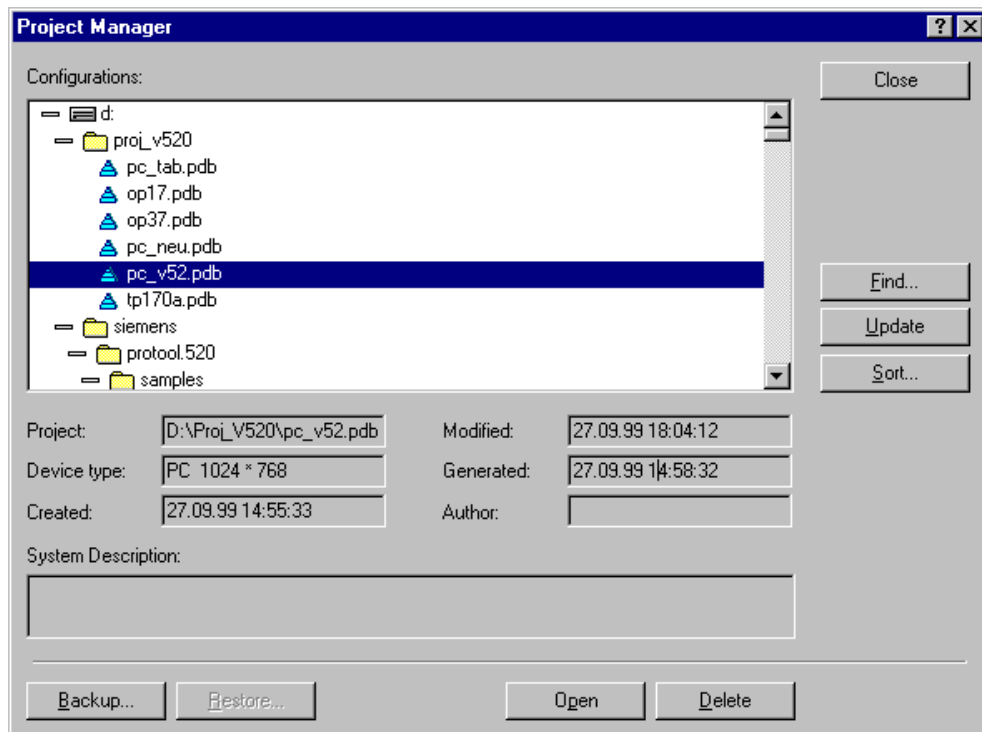
You start Project Manager by choosing "File" > "Project Manager" from the menu. When you call Project Manager for the first time, the "Find" dialog box is displayed initially. Here you choose the drives and directories which have to be searched for ProTool files. Only those projects that are located within these directories are displayed by Project Manager.



*"Find" dialog box*

After you have selected the appropriate directories, or when you call Project Manager later again, the Project Manager window proper appears.

## Display



### *Project Manager*

In the left segment of the Project Manager you will see a hierarchical structure of all the projects located in the directories in which Project Manager searched. Here you can select a project in order to open it, delete it or back it up.

In the center segment of the window (Project data) you will find detailed information on the project highlighted on the list.

Using the buttons in the area on the right, you can extend the list of scanned directories ("Find" button), have the directories searched again ("Update" button) and have the list sorted by different project data such as "project name", "device type", "date created", etc. ("Sort" button).

## 7.4 Constraints with printing

### Printer drivers

Note the following constraints with certain printer drivers:

- It might not be possible to print the project with CANON drivers. Printing will be discontinued in this case.
- With the Apple laser printer, the first line is not printed. This problem does not occur with drivers for the HP LaserJet III, PostScript or PostScript printers.

### ASCII character set

With some printers, it is not enough simply to set the ASCII character set in the project. Make sure the ASCII character set is set on the printer too.

## 7.5 Example of documenting a project

This example shows you how to create a customized printout.

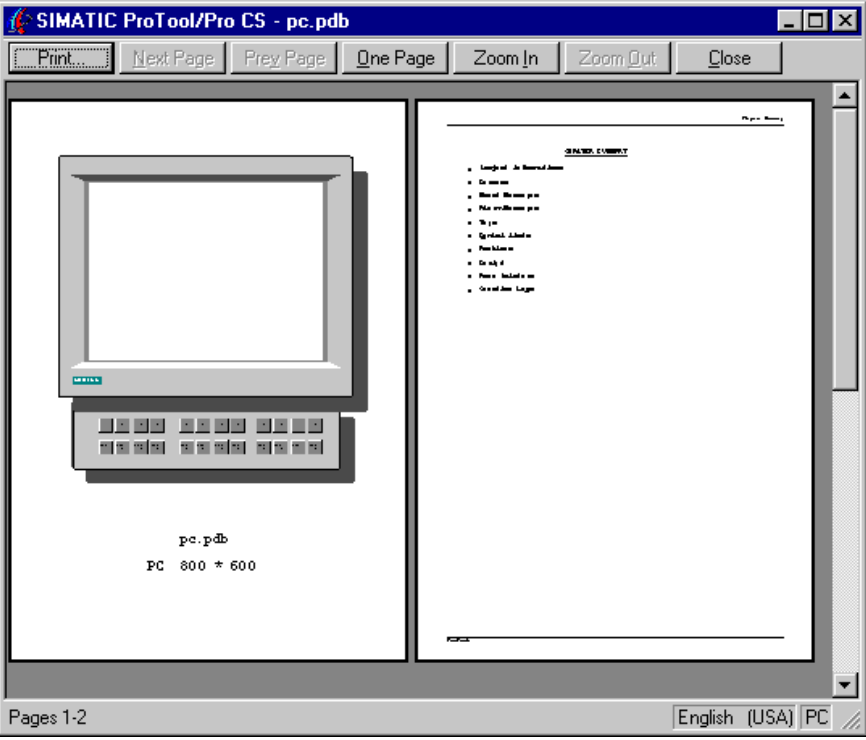
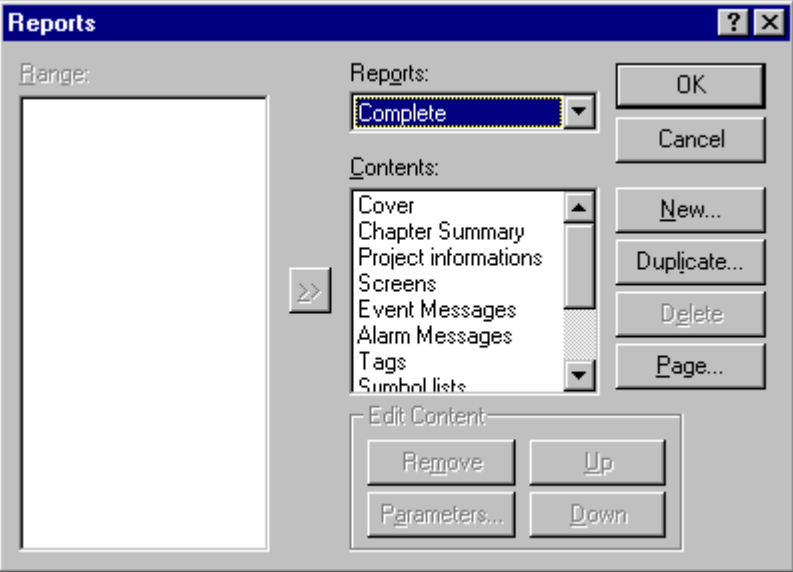
### 7.5.1 How to create a customized printout

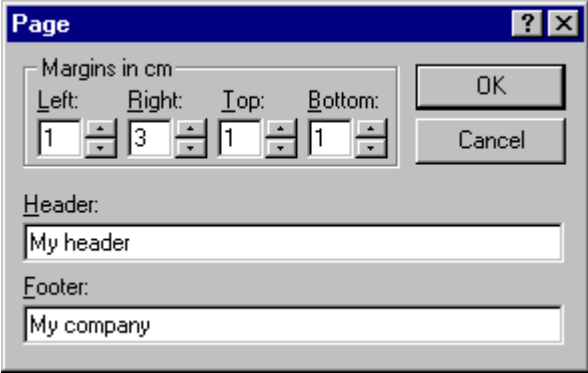
#### Objective

You wish to print all the data in your project. Unlike the default setting, you do not want the ProTool graphic but your company logo, which you have already used in your project under the name of LOGO, to be printed on the cover sheet. You want to leave a margin for handwritten comments on the right side of the printout.

#### Procedure:

Procedure:	
1	Open the project from which you wish to print data.
2	Choose "File" > "Print" from the menu.
3	In the "Print" dialog box at "Reports", choose the <code>Complete</code> report. You will now see the individual chapters in the "Contents" list in the order in which they will later be output.

	<p><b>Procedure:</b></p>
<p>4</p>	<p>Click the "Preview" button. The print preview shows you how your printout will look later. You want to replace the graphic on the first page (cover) with your company logo. You want the right border to be much wider.</p> 
<p>5</p>	<p>Exit from the preview by clicking the "Print" button.</p>
<p>6</p>	<p>To perform the changes you require, click the "Reports" button. The "Reports" dialog box is opened.</p> 
<p>7</p>	<p>To modify the margin, click the "Page" button.</p>

Procedure:	
8	<p>Enter the value you require for the right border at "Right" - for example, 3 for 3 cm. If you like, you can specify a piece of text here for the headers and footers at the same time.</p>  <p>The screenshot shows a 'Page' dialog box with a title bar containing a question mark and a close button. Below the title bar, there is a section titled 'Margins in cm' with four spinners for 'Left', 'Right', 'Top', and 'Bottom'. The values are 1, 3, 1, and 1 respectively. To the right of these spinners are 'OK' and 'Cancel' buttons. Below the margins section, there are two text input fields: 'Header:' containing 'My header' and 'Footer:' containing 'My company'.</p>
9	Confirm your input by clicking OK. You are now back in the "Reports" dialog box.
10	To modify the definition for the cover sheet, select the entry <i>Cover Sheet</i> in the "Contents" list and click the "Parameters" button.
11	In the "Cover Sheet" dialog box, select the graphic called <i>LOGO</i> with your company logo at "Graphic".
12	Click OK twice until you are back in the "Print" dialog box.
13	Finally, click the OK button to activate the printing process.

You have now modified the definitions for the report called *Complete*. The changes will be available in future when you print other projects.

# Appendix

**A HMI System Messages**

**B Abbreviations**





# A HMI System Messages

## In this chapter

This chapter contains a selection of important HMI system messages for Windows-based systems. The table indicates when the messages occur and how they, or their cause, can be cleared. Not every message is relevant for each operating unit.

## HMI system message parameters

The HMI system messages may contain parameters which are not decoded for the user but which are relevant in respect of the cause of an error since they provide a reference to the source code of the runtime software. These parameters are issued according to the text "Error code:".

---

### Notice

HMI system messages are issued in the language currently set on the operating unit.

---

## Procedure in the case of "internal errors"

Please proceed as follows in the case of all system messages related to "internal errors":

Start up the operating unit again.

Download the configuration again.

Switch the operating unit off, stop the PLC and then restart both.

If the error occurs again, please contact the SIMATIC Customer Support. Make reference to the specified error number and message tags.

Number	Effect/Cause	Remedy
10000	The print job could not be started or was terminated for an unknown reason. The printer is incorrectly configured. Or: There are no rights for a network printer available.	Check the printer settings and cable connections. If a fault occurs repeatedly, contact the hotline.
10001	No printer has been installed or no standard printer configured.	Install a printer and/or mark one as standard printer.
10002	The intermediate buffer for printing graphics is full. Up to two graphics can be buffered.	Do not issue print jobs so quickly in succession.
10003	Graphics can be buffered again.	-

Number	Effect/Cause	Remedy
10004	The intermediate buffer for printing lines in text mode (e.g. messages) is full. Up to 1000 lines can be buffered.	Do not issue print jobs so quickly in succession.
10005	Lines of text can be buffered again.	-
10006	The Windows print system reports an error. For information on the cause, refer to the text printed and, if available, the error number. Printing is not performed or it is incorrect.	Repeat the action, if necessary.
20010	A fault has occurred in the script line called in from the specified script. Execution of the script function was, therefore, terminated. In this case, it is advisable to check any previous system messages too.	Select the specified script line in the configuration. Check tags, whether the types used are permissible. Check Functions, whether the number and types of parameter are correct.
20011	An error has occurred in a script called in by the script specified. Execution of the script function– has, therefore, been terminated in the subscript. In this case, it is advisable to check any previous system messages too.	Select the scripts from the configuration which are called in directly or indirectly via the specified script. Check tags, whether the types used are permissible. Check Functions, whether the number and types of parameter are correct.
20012	Inconsistent configuration data is present. Therefore, the script could not be created.	Compile the configuration again.
20013	VBScript.dll is not correctly installed. Therefore, no scripts can be executed.	Re–install ProTool/Pro RT.
20014	A value is returned by the script function which is not written in any configured return tag.	Select the specified script in the configuration. Check whether the script name has been assigned a value.
20015	Too many scripts have been triggered in quick succession. If more than 20 scripts are queued to be processed, any subsequent scripts are rejected. In this case, the script indicated in the message is not executed.	Check where the scripts are being triggered from. Extend the times, e.g. the polling time of the tags, which trigger the scripts.
30010	The tag could not accept the function result, e.g. in the case of exceeding the value range.	Check the tag type of the function parameter.

Number	Effect/Cause	Remedy
30011	A function could not be executed because the function was assigned an invalid value or type in the parameter.	Check the parameter value and tag type of the invalid parameter. If a tag is used as a parameter, check its value.
40010	The function could not be executed since the parameters could not be converted to a common tag type.	Check the parameter types in the configuration.
40011	The function could not be executed since the parameters could not be converted to a common tag type.	Check the parameter types in the configuration.
50000	The operating unit receives data faster than it is capable of processing. Therefore, no further data is received until the data currently available has been processed. Data exchange then resumes.	-
50001	Data exchange has been resumed.	-
60000	This message is generated by the function "Display system message text". The text to be displayed is transferred to the function as a parameter.	-
60010	The file could not be copied in the direction defined because one of the two files is currently open or the source/target path is not available. It is possible that the Windows NT user has no access rights to one of the two files.	Restart the function or check the paths of the source/target files. Using Windows NT with NTFS: The user executing ProTool/Pro RT must be granted access rights for the files.
60011	An attempt was made to copy a file to itself. It is possible that the Windows NT user has no access rights to one of the two files.	Check the path of the source/target file. Using Windows NT with NTFS: The user executing ProTool/Pro RT must be granted access rights for the files.
70010	The application could not be started because it could not be found in the path specified or insufficient memory space was available.	Check whether the application exists in the specified path or close other applications.

Number	Effect/Cause	Remedy
70011	<p>Could not change the system time. The error message appears in conjunction with Area Pointer Date/Time PLC.</p> <p>Possible causes:</p> <ul style="list-style-type: none"> <li>- an illegal time was forwarded in the PLC job,</li> <li>- the Windows NT user does not have the user right to change the system time.</li> </ul> <p>If the value 13 is shown as the first parameter in the system message, the second parameter identifies the byte having the incorrect value.</p>	<p>Check in the time that you want to set the values for the year, month, day, hour, minute, second, and day of the week.</p> <p>In Windows NT: The user executing ProTool/Pro RT must be given the right to change the Windows NT system time (Administrator/User Manager, conventions).</p>
70012	<p>An error occurred when executing the function Exit Runtime with the option Exit also Windows.</p> <p>Windows and ProTool/Pro RT are not terminated.</p> <p>A possible cause is that other applications cannot be terminated.</p>	<p>Terminate all applications currently running.</p> <p>Then terminate Windows.</p>
70013	<p>The system time could not be modified because an invalid value was entered.</p> <p>Incorrect separators may have been used.</p>	<p>Check the time which is to be set.</p>
70014	<p>The system time could not be modified.</p> <p>Possible causes:</p> <ul style="list-style-type: none"> <li>- an impermissible time was transferred</li> <li>- the Windows NT user has no user rights to modify the system time,</li> <li>- Windows rejects the setting request.</li> </ul>	<p>Check the time which is to be set.</p> <p>Under Windows NT: The user executing ProTool/Pro RT must be assigned the rights to modify the system time from Windows NT (administration/user manager, guidelines).</p>
70015	<p>The system time could not be read because Windows rejects the reading function.</p>	-
70016	<p>An attempt was made to select a screen by means of a function or job. This is not possible because the screen number specified does not exist.</p> <p>Or: a screen could not be generated due to insufficient system memory.</p>	<p>Check the screen number in the function or job with the screen numbers configured.</p> <p>Refer the number to a screen, if necessary.</p>
70017	<p>Date/Time is not read from the area pointer because the address set in the PLC is either not available or has not been set up.</p>	<p>Change the address or set up the address in the PLC.</p>
70018	<p>Acknowledgement that the password list has been successfully imported.</p>	-

Number	Effect/Cause	Remedy
70019	Acknowledgement that the password list has been successfully exported.	-
70020	Acknowledgement for activation of message recording.	-
70021	Acknowledgement for deactivation of message recording.	-
70022	Acknowledgement to starting the Import Password List action.	-
70023	Acknowledgement to starting the Export Password List action.	-
70027	Backing up the RAM file system has been started.	-
70028	Backing up of the RAM file system has been completed successfully. The files from the RAM have been copied in the Flash memory. Following a restart, these saved files will be copied back in the RAM file system.	-
70029	Backup of the RAM file system has failed. No backup copy of the RAM file system has been made.	Check the settings in the OP Properties dialog and save the RAM file system using the Save Files button in the Persistent Storage tab control.
70030	The parameters configured for the function are faulty. The connection to the new PLC was not established.	Compare the parameters configured for the function with the parameters configured for the PLC and correct them as necessary.
70031	The PLC configured in the function is not an S7 PLC. The connection to the new PLC was not established.	Compare the S7 PLC name parameter configured for the function with the parameters configured for the PLC and correct them as necessary.
80001	The archive specified is filled to the size defined (in percent) and must be stored elsewhere.	Store the file or table by executing a 'move' or 'copy' function.
80002	A line is missing in the specified archive.	-
80003	The copying process for archiving was not successful. In this case, it is advisable to check any subsequent system messages, too.	-
80006	Since archiving is not possible, this causes a permanent loss of the functionality.	In the case of databases, check whether the corresponding data source exists and start up the system again.

Number	Effect/Cause	Remedy
80009	A copying action has been completed successfully.	-
80010	Since the path was incorrectly entered in ProTool, this causes a permanent loss of the functionality.	Configure the path for the respective archive again and restart the system when the full functionality is required.
80012	Archive values are stored in a buffer. If the values are read to the buffer faster than they can be physically written (e.g. using a hard disk), overloading may occur and recording is then stopped.	Archive less values. Or increase the recording interval.
80013	The overload status no longer applies. Archiving resumes the recording of all values.	-
80014	The same action was triggered twice in quick succession. Since the process is already in operation, the action is only carried out once.	-
80015	ProTool issues error messages from within the Windows operating system. These error messages are issued in the Windows system language. Refer to the Windows documentation for more information about the error messages.	-
80016	The archives are separated by the function Close archive and the incoming entries exceed the defined buffer size. All the jobs in the buffer are deleted.	Reconnect the archives.
80017	The incoming entries cause the defined buffer size to be exceeded. This can be caused, e.g. by several copying actions being activated at the same time. All the copy jobs in the buffer are deleted.	Terminate the copy process.
80018	All the archives are reconnected by means of the DB layer, e.g. after executing the function Open_archive. Values are then written back into the tables.	-
80019	All the archives are separated from the DB layer and all connections terminated, e.g. after executing the function Close_archive. Values are temporarily buffered and written in the tables when the connection is re-established. There is no connection to the storage medium and a change can take place.	-

Number	Effect/Cause	Remedy
80020	The max. number of simultaneously activated copy actions has been exceeded. Copying is not executed.	Wait until the current copying actions have been completed and restart the last copy action.
80021	An attempt was made to delete an archive which is still involved with a copy action. Deletion has not been executed.	Wait until the current copying action has been completed and restart the last action.
80022	An attempt was made to start a sequence archive, which is not a sequence archive, from an archive using the function Start_sequence_archive. No sequence archive is created.	Check the project for the following: - the function Start_sequence_archive is correctly configured. - the variable parameters are being correctly provided at the operating unit.
80023	An attempt was made to copy an archive to itself. The archive is not copied.	Check the project for the following: - the function Copy_archive is correctly configured. - the variable parameters are being correctly provided at the operating unit.
80024	The function Copy_archive is configured not to permit copying when the target archive already contains data (Parameter: Write mode). The archive is not copied.	Modify the function Copy_archive in the project, if necessary. Before initiating the function, delete the target archive.
80025	The copy action is interrupted. Data written up to this point is retained. Deletion of the target table (if configured) is not executed. The cancellation is documented by an error entry \$RT_ERR\$ at the end of the target table.	-
80026	The message is issued after all the archives have been successfully initialized. Values are written in the archives from this moment on. Prior to this, no values are archived even though the runtime software is operating.	-
80027	The internal Flash memory has been specified as the memory location for an archive. This is not permissible. No values will be archived for this archive and the archive will not be created.	Configure "Storage Card" as the memory location or a network path.

Number	Effect/Cause	Remedy
80028	The message serves as a status acknowledgment that initialization of the archives is currently running. No values are archived until system message 80026 is issued.	-
80029	The number of archives specified in the message could not be initialized. Initialization of the archives has been completed. The faulty archives are not available for archiving jobs.	Evaluate the additional system message, related to this message, which is also issued. Check the configuration, the ODBC (Open Database Connectivity) and the specified drive.
80030	The structure of the table(s) available does not match the archiving structure expected. The archiving process is stopped for this archive.	Delete the existing tables manually, in advance.
80032	Archives can be configured with a function trigger. This is triggered as soon as the archive is full. If runtime is started and the archive is already full, the trigger function will not be initiated. The archive specified no longer archives data since it is full.	Stop the runtime, delete the archive and restart the runtime again. Or: Configure a button in the runtime which contains the same actions as the function trigger and press it.
110000	The operating mode status has been changed. The operating mode is now offline.	-
110001	The operating mode status has been changed. The operating mode is now online.	-
110002	The operating mode status has not been changed.	Check the connection to the PLCs. Check whether the address area for the area pointer "Coordination" in the PLC is available.
110003	The operating status of the PLC specified is changed by the function PLC Connect Isolate. The operating mode is now offline.	-
110004	The operating status of the PLC specified is changed by the function PLC Connect Isolate. The operating mode is now online.	-



Number	Effect/Cause	Remedy
110005	An attempt was made to use the function PLC Connect Isolate to switch the specified PLC to operating mode online although the entire system is in operating mode Offline. This switch-over is not permissible. The PLC remains in operating mode offline.	Switch the complete system to operating mode online and execute the function again.
110006	The Colorado system has been extended by the addition of the User Version area pointer. If the user version is not correct, the runtime is stopped.	Check the user versions. Either the wrong version was entered in the PLC or configuration or the wrong configuration was started for the PLC user version.
120000	The trend is not displayed because an incorrect axis to the trend, or incorrect trend, has been configured.	Change the configuration.
120001	The trend is not displayed because an incorrect axis to the trend, or incorrect trend, has been configured.	Change the configuration.
120002	The trend is not displayed because the tag assigned tries to access an invalid PLC address.	Check whether the data area for the tag exists in the PLC, the configured address is correct or the value range for the tag is correct.
130000	The action was not executed.	Close other applications. Delete files no longer required from the hard disk.
130001	The action was not executed.	Delete files no longer required from the hard disk.
130002	The action was not executed.	Close other applications. Delete files no longer required from the hard disk.
130003	No target data carrier is inserted. The process is stopped.	Check, for example, whether: - access has been made to the correct data carrier - the data carrier has been inserted
130004	The target data carrier is write-protected. The process is stopped.	Check whether access has been made to the correct data carrier. Remove the write protection.
130005	The file is write-protected. The process is stopped.	Check whether access has been made to the correct file. Modify the file attributes, if necessary.

Number	Effect/Cause	Remedy
130006	No access to file is possible. The process is stopped.	Check, for example, whether: <ul style="list-style-type: none"> <li>- access has been made to the correct file</li> <li>- the file exists</li> <li>- a different action prevents simultaneous access to the file</li> </ul>
140000	Online connection to the PLC has been successfully established.	-
140001	Online connection to the PLC has been disconnected.	-
140003	No tag updating or writing is executed.	Check the connection and whether the PLC is switched on. Check the parameter definitions in the Control Panel using "Set PU/PC interface". Restart the system.
140004	No tag updating or writing is executed because the access point or the subrack configuration is incorrect.	Check the connection and whether the PLC is switched on. Check the access point or the subrack configuration (MPI, PPI, PROFIBUS) in the Control Panel with "Set PU/PC interface". Restart the system.
140005	No tag updating or writing is executed because the address of the operating unit is incorrect (possibly too high).	Use a different operating unit address. Check the connection and whether the PLC is switched on. Check the parameter definitions in the Control Panel using "Set PU/PC interface". Restart the system.
140006	No tag updating or writing is executed because the baud rate is incorrect.	Select a different baud rate in ProTool/Pro (according to subrack, profile, communication peer, etc.).

Number	Effect/Cause	Remedy
140007	<p>No tag updating or writing is executed because the bus profile is incorrect (see %1).</p> <p>The following parameters could not be entered in the registry:</p> <ul style="list-style-type: none"> <li>1: Tslot</li> <li>2: Tqui</li> <li>3: Tset</li> <li>4: MinTsdr</li> <li>5: MaxTsdr</li> <li>6: Trdy</li> <li>7: Tid1</li> <li>8: Tid2</li> <li>9: Gap Factor</li> <li>10: Retry Limit</li> </ul>	<p>Check the user defined bus profile.</p> <p>Check the connection and whether the PLC is switched on.</p> <p>Check the parameter definitions in the Control Panel using "Set PU/PC interface".</p> <p>Restart the system.</p>
140008	<p>No tag updating or writing is executed because the configuration data is incorrect:</p> <p>The following parameters could not be entered in the registry:</p> <ul style="list-style-type: none"> <li>0: General errors</li> <li>1: Incorrect version</li> <li>2: Profile cannot be entered in the registry.</li> <li>3: Sub-network type cannot be entered in the registry.</li> <li>4: Target rotation time cannot be entered in the registry.</li> <li>5: Highest address (HSA) incorrect.</li> </ul>	<p>Check the connection and whether the PLC is switched on.</p> <p>Check the parameter definitions in the Control Panel using "Set PU/PC interface".</p> <p>Restart the system.</p>
140009	<p>No tag updating or writing is executed because the module for the S7 communication was not found.</p>	<p>Re-install the module in the Control Panel using "Set PU/PC interface".</p>
140010	<p>No S7 communication peer could be found because the PLC is switched off.</p> <p>DP/T:</p> <p>The option "Is not active as the only master" is set in the Control Panel under "Set PU/PC interface".</p>	<p>Switch the PLC on.</p> <p>DP/T:</p> <p>If only one master is connected to the network, deactivate the option "Is not active as the only master" in "Set PU/PC interface".</p> <p>If the network has more than one master, switch this master on. Do not modify any settings here, otherwise a bus fault may occur.</p>
140011	<p>No tag updating or writing is executed because communication is interrupted.</p>	<p>Check the connection and that the communication peer is switched on.</p>

Number	Effect/Cause	Remedy
140012	There is an initialization problem (e.g. when ProTool/Pro RT has been terminated in the Task Manager). Or: another application (e.g. STEP7, WINCC) is active with different bus parameters and the driver cannot be started with the new bus parameters (e.g. baud rate).	Restart the operating unit. Or start ProTool/Pro RT first and then the other applications.
140013	The MPI cable is not plugged in and, thus, there is no power supply.	Check the connections.
140014	-	Modify the operating unit address in the configuration in PLC.
140015	Incorrect baud rate Or: incorrect bus parameter (e.g. HSA) Or: OP address HSA Or: incorrect interrupt vector (interrupt does not arrive at the driver)	Correct the incorrect parameters.
140016	-	Change the interrupt number.
140017	-	Change the interrupt number.
140018	The consistency check was deactivated by Simotion Scout. Only one appropriate note appears.	Activate the consistency check with Simotion Scout again and load the configuration in the project once more.
140019	Simotion Scout loads a new project in the PLC. Connection to the PLC is interrupted.	Wait until the end of the reconfiguration.
140020	The version in the PLC and that in the configuration (FWD file) do not match. Connection to the PLC is interrupted.	Remedy: - Load the current version in the PLC using Simotion Scout. - Generate the anew project using ProTool CS, end ProTool RT and start with the new configuration.
150000	No more data is read or written. Possible causes: - The cable is defective. - The PLC does not respond, is defective, etc. - Connection is established via the wrong interface. - The system is overloaded.	Check that the cable is plugged in, the PLC is operational, the correct interface is used. Reboot the system if the system message is displayed continuously.

Number	Effect/Cause	Remedy
150001	Connection is re-established because the cause of the interruption has been eliminated.	-
160000	No more data is read or written. Possible causes: - The cable is defective. - The PLC does not respond, is defective, etc. - Connection is established via the wrong interface. - The system is overloaded.	Check that the cable is plugged in, the PLC is operational, the correct interface is used. Reboot the system if the system message is displayed continuously.
160001	Connection is re-established because the cause of the interruption has been eliminated.	-
160010	There is no connection to the server because the server identification (CLS-ID) cannot be established. Values cannot be read or written.	Check the access rights.
160011	There is no connection to the server because the server identification (CLS-ID) cannot be established. Values cannot be read or written.	Check, for example, whether: - the server name is correct - the computer name is correct - the server is registered
160012	There is no connection to the server because the server identification (CLS-ID) cannot be established. Values cannot be read or written.	Check, for example, whether: - the server name is correct - the computer name is correct - the server is registered Information for experienced users: Interpret the value from HRESULT.
160013	The specified server was started as InProc Server. This has not been released and may possibly lead to incorrect behavior because the server is running in the same process area as the ProTool/Pro RT runtime software.	Configure the server as OutProc Server or Local Server.
160014	Only one OPC server project can be started on a PC/MP. An error message appears when an attempt is made to start a second project. The second project has no OPC server functionality and cannot be located as an OPC server from external sources.	Do not start a second project with OPC server functionality on the computer.

Number	Effect/Cause	Remedy
160015	Could not start deviceCOM software for OPC communication.	Perform an operating system download. To do this, observe the online Help for ProSave. Should the problem still persist, contact the Hotline.
160016	Could not start deviceCOM transport software for OPC communication.	If you have manually modified the deviceCOM parameters on the operating unit, check the changes you made. Should the problem still persist, contact the Hotline.
170000 <sup>1)</sup>	S7 diagnostics messages are not displayed because it is not possible to logon to the S7 diagnostics with this unit. The service program is not supported.	-
170001 <sup>1)</sup>	The S7 diagnostics buffer cannot be displayed because communication with the PLC has been switched off.	Switch the PLC online
170002 <sup>1)</sup>	The S7 diagnostics buffer cannot be displayed because reading in the diagnostics buffer (SZL) was terminated due to an error.	-
170003 <sup>1)</sup>	The display of an S7 diagnostics message is not possible. An internal error %2 has been reported.	-
170004 <sup>1)</sup>	The display of an S7 diagnostics message is not possible. An internal error with error class %2, error number %3 has been reported.	-
170007 <sup>1)</sup>	It is not possible to read in the S7 diagnostics buffer (SZL) because it was terminated with an internal error with error class %2 and error code %3.	-
<sup>1)</sup>	The optional parameter %1 at the start of the message may contain an identification for the S7 connection when several S7s are in parallel operation and are connected to diagnostics equipment.	
180000	A component/OCX receives configuration data with a version identification which is not supported.	Install a newer component.

Number	Effect/Cause	Remedy
180001	The system is overloaded because too many actions have been activated simultaneously. Not all the actions can be executed, some are rejected.	Remedy: - Increase the configured cycle times or basic clock. - Generate the messages slower (polling). - Trigger the scripts and functions at greater intervals. - If the message appears more frequently: Restart the operating unit.
180002	The screen keyboard could not be activated. Possible causes: - The screen keyboard is not generally supported under Windows 95. - The file "TouchInputPC.exe" was not registered due to an incorrectly executed Setup.	If Windows 95 is not available: Install the runtime software again.
190000	It is possible that the tag will not be updated.	-
190001	The tag is updated following an error status after the cause of the last error state has been eliminated (return to normal operation).	-
190002	The tag is not updated because communication to the PLC has been switched off.	Switch on communication via the function "Set Online".
190004	The tag is not updated because the configured address is not available for this tag.	Check the configuration.
190005	The tag is not updated because the configured PLC type does not exist for this tag.	Check the configuration.
190006	The tag is not updated because it is not possible to map the PLC type in the tag type.	Check the configuration.
190007	The tag values are not modified because the connection to the PLC has been terminated or the tag is offline.	Switch Online or re-establish connection to the PLC.
190008	The threshold values configured for the tag have been violated, e.g. by - an entered value, - a function, - a script.	Observe the configured or current threshold value of the tag.

Number	Effect/Cause	Remedy
190009	An attempt has been made to assign a value to a tag which is outside the value range permitted for this type. E.g. a value of 260 entered for a byte tag or a value of -3 for a signless word tag.	Observe the value range for the tag type.
190010	The tag is described with values too often (e.g. in a loop triggered by a script). Values are lost because the maximum of 100 event have been stored in the buffer.	Increase the time interval between the multi-writing tasks.
190011	Possible causes: - The value entered could not be written to the configured PLC tag because it was either above or below the value range. - The input is rejected and the original value is reset.	Ensure that the value entered is within the value range of the PLC tags.
190011	Possible causes: - Connection to the PLC has been interrupted.	Check the connection to the PLC.
190012	It is not possible to convert a value from a source format to a target format, e.g.: - A value should be assigned to a counter which is outside the valid, PLC-dependent value range. - A tag of the type Integer should be assigned a value of the type String.	Check the value range or type of the variable.
190100	The area pointer is not updated because the configured address for this area pointer is not available. Type: 1 Event messages 2 Alarm messages 3 PLC acknowledgment 4 Operating unit acknowledgment 5 LED assignment 6 Trend request 7 Trend transfer 1 8 Trend transfer 2 No.: is the consecutive number displayed in ProTool/Pro.	Check the configuration.
190101	The area pointer is not updated because it is not possible to map the PLC type in the area pointer type. Parameter type and no.: See message 190100	-



Number	Effect/Cause	Remedy
190102	The area pointer is updated following an error status after the cause of the last error state has been eliminated (return to normal operation). Parameter type and no.: See message 190100	-
200000	Coordination is not executed because the address configured in the PLC does not exist/has not been set up.	Change the address or set up the address in the PLC.
200001	Coordination is not executed because the address configured in the PLC cannot be written.	Change the address or set up the address in the PLC in an area which can be written.
200002	Coordination is not carried out at the moment because the address format of the area pointer does not match the internal storage format.	Internal error
200003	Coordination can be executed again because the last error status has been eliminated (return to normal operation).	-
200004	It is possible that coordination is not executed.	-
200005	No more data is read or written. Possible causes: - The cable is defective. - The PLC does not respond, is defective, etc. - The system is overloaded.	Check that the cable is connected and the PLC is in order. Reboot the system if the system message is displayed continuously.
210000	Jobs are not processed because the address configured in the PLC does not exist/has not been set up.	Change the address or set up the address in the PLC.
210001	Jobs are not processed because the address configured in the PLC cannot be written to/read from.	Change the address or set up the address in the PLC in an area which can be written to/read from.
210002	Commands are not executed because the address format of the area pointer does not match the internal storage format.	Internal error
210003	The job mailbox is processed again because the last error status has been eliminated (return to normal operation).	-
210004	It is possible that the job mailbox is not processed.	-
210005	A PLC job was triggered by an impermissible number.	Check the PLC program.

Number	Effect/Cause	Remedy
210006	A fault occurred while attempting to execute the PLC job. The PLC job is, therefore, not executed. Observe the subsequent/previous system message, if appropriate.	Check the parameter types in the PLC job. Compile the configuration again.
220000 <sup>2)</sup>	-	-
<sup>2)</sup>	A WinCC channel provides the message texts via an interface. This text is issued via this message. ProTool/Pro RT has no influence on this texts.	
220001	The tag is not downloaded because the associated channel/the unit does not support downloading the data type bool/bit.	Change the configuration.
220002	The tag is not downloaded because the associated channel/the unit does not support downloading the data type byte.	Change the configuration.
220003	The associated driver could not be uploaded. It is possible that the driver is not installed.	Install the driver by re-installing ProTool/Pro RT.
220004	Communication is terminated and no update is executed because the cable is not connected or is defect etc.	Check the connection.
220005	Communication is running.	-
220006	The connection is established to the specified PLC at the specified interface.	-
220007	The connection to the specified PLC at the specified interface is disconnected.	Check that: - the cable is plugged in - the PLC is OK - the correct interface is used - the configuration is OK (interface parameters, protocol settings, PLC address).  Reboot the system if the system message is displayed continuously.
220008	The PLC driver cannot access the specified interface or open it. It is possible that another application is using this interface or an interface is used which is not available on the target device. There is no communication with the PLC.	Terminate all the programs which access the interface and reboot the computer. Use another interface which is available in the system.

Number	Effect/Cause	Remedy
230000	The value entered could not be accepted. The entered value is rejected and the previous value is specified again. Either the value range has been exceeded or impermissible characters were entered.	Enter a permissible value.
230002	Since the current password level is inadequate or the password dialog box was closed with ESC, the entry is rejected and the previous value is specified again.	Activate an adequate password level using Login.
230003	Changeover to the specified screen is not executed because the screen is not available/configured. The current screen remains selected.	Configure the screen. Check the selection function.
240000 <sup>3)</sup>	Runtime is operating in Demo mode. There is either no Stopcopy license or it is defect.	Load the license.
240001 <sup>3)</sup>	Runtime is operating in Demo mode. Too many tags are configured for the installed version.	Load an adequate license / powerpack.
240002 <sup>3)</sup>	Runtime is operating with a time-limited standby authorization.	Restore the full authorization.
<sup>3)</sup>	The specified text comes from the component resources.	
240003	Authorization cannot be executed. ProTool/Pro RT is running in Demo mode.	Restart ProTool/Pro RT or reinstall it.
240004	Error during reading the standby authorization. ProTool/Pro RT is running in Demo mode.	Restart ProTool/Pro RT, install the authorization or repair the authorization (see Commissioning Instructions Software Protection).
250000	The tag in the specified line in Status/Control is not updated because the address configured for this tag is not available.	Check the set address and then check that the address has been set up in the PLC.
250001	The tag in the specified line in Status/Control is not updated because the PLC type configured for this tag is not available.	Check the set address.
250002	The tag in the specified line in Status/Control is not updated because it is not possible to map the PLC type in the tag type.	Check the set address.

Number	Effect/Cause	Remedy
250003	No connection could be established to the PLC. The tags will not be updated.	Check the connection to the PLC. Check that the PLC is switched on and online is activated.
260000	A password has been entered which is unknown to the system. Therefore, the lowest password level has been set. This corresponds to the status following Logout.	Enter a known password in the password input field (with corresponding level).
260001	A password has been entered whose assigned level does not permit execution of the function. The password level currently set is displayed for information purposes.	Modify the password level in the password input field enter a password with a sufficiently high level.
260003	The user has logged off from the system. If the password level is 0, no user is logged on.	-
270000	A tag is not displayed in the message because it attempts to access an invalid address in the PLC.	Check whether the data area for the tag exists in the PLC, the configured address is correct or the value range for the tag is correct.
270001	There is a unit-dependent limit as to how many messages may be queued simultaneously in order to be displayed (see GHB). This limit has been exceeded. The display no longer contains all the messages. However, all the messages are recorded in the message buffer.	-
270002	Messages are displayed from an archive are displayed for which there is no data in the current project. Placeholders are issued for the messages.	Delete older archive files, if necessary.
270003	The service cannot be set up because too many devices was to set up this service. A maximum of four devices can execute this action.	Connect fewer operating units which want to use the service.
280000	Connection is re-established because the cause of the interruption has been eliminated.	-

Number	Effect/Cause	Remedy
280001	No more data is read or written. Possible causes: - The cable is defective. - The PLC does not respond, is defective, etc. - Connection is established via the wrong interface. - The system is overloaded.	Check that the cable is plugged in, the PLC is operational, the correct interface is used. Reboot the system if the system message is displayed continuously.
280002	A connection is used which requires a function module in the PLC. The function block has replied. Communication can now proceed.	-
280003	A connection is used which requires a function module in the PLC. The function block does not reply.	Check that the cable is plugged in, the PLC is operational, the correct interface is used. Reboot the system if the system message is displayed continuously. The remedy is dependent on the error code: 1: The function block must set the COM bit in the response container. 2: The function block may not set the ERROR bit in the response container 3: The function block must respond within the specified time (timeout) 4: Establish an online connection to the PLC
280004	The online connection to the PLC has been interrupted. There is no data exchange at present.	Check the PLC parameters in ProTool Pro: baud rate, block length, station address. Check that the cable is plugged in, the PLC is operational, the correct interface is used. Reboot the system if the system message is displayed continuously.
290000	The tag could not be read or written. It is assigned the start value. The message can be entered in the message buffer for up to four more failed tags, if necessary. After that, the message number 290003 is issued.	Check in the configuration that the address has been set up in the PLC.

Number	Effect/Cause	Remedy
290001	An attempt has been made to assign a value to a tag which is outside the value range permitted for this type. The message can be entered in the message buffer for up to four more failed tags, if necessary. After that, the message number 290004 is issued.	Observe the value range for the tag type.
290002	It is not possible to convert a value from a source format to a target format. The message can be entered in the message buffer for up to four more failed tags, if necessary. After that, the message number 290005 is issued.	Check the value range or type of the variable.
290003	This message is issued when message number 290000 is triggered more than five times. In this case, no further individual messages are generated.	Check in the configuration that the tag addresses have been set up in the PLC.
290004	This message is issued when message number 290001 is triggered more than five times. In this case, no further individual messages are generated.	Observe the value range for the tag type.
290005	This message is issued when message number 290002 is triggered more than five times. In this case, no further individual messages are generated.	Check the value range or type of the variable.
290006	The threshold values configured for the tag have been violated by values entered.	Observe the configured or current threshold value of the tag.
290007	There is a difference between the source and target structure of the recipe currently being processed. The target structure contains an additional data record tag which is not available in the source structure. The data record tag specified is assigned its start value.	Insert the specified data record tag in the source structure.
290008	There is a difference between the source and target structure of the recipe currently being processed. The source structure contains an additional data record tag which is not available in the target structure and therefore cannot be assigned. The value is rejected.	Remove the specified data record tag in the specified recipe from the project.

Number	Effect/Cause	Remedy
290010	The storage location configured for the recipe is not permitted. Possible causes: Impermissible characters, write protected, data medium full or does not exist.	Check the path specification configured.
290011	The data record specified by the number does not exist.	Check the source for the number (constant or variable value).
290012	The recipe specified by the number does not exist.	Check the source for the number (constant or variable value).
290013	An attempt was made to save a data record under a data record number which already exists. The process is not executed.	Remedy: - Check the source for the number (constant or variable value). - Delete the data record beforehand. - Change the function parameter "Overwrite".
290014	The file specified to be imported could not be found.	Check that: - Check the file name. - Ensure that the file is in the specified directory.
290020	Acknowledgement that downloading of data records from operating unit to PLC has started.	-
290021	Acknowledgement that downloading of data records from operating unit to PLC has been completed without any errors.	-
290022	Acknowledgement that downloading of data records from operating unit to PLC has been terminated due to an error.	Check the configuration: - have the tag addresses been set up in the PLC? - does the recipe number exist? - does the data record number exist? - has the function parameters "Overwrite"?
290023	Acknowledgement that downloading of data records from the PLC to the operating unit has started.	-

Number	Effect/Cause	Remedy
290024	Acknowledgement that downloading data records from the PLC to the operating unit has been completed without any errors.	-
290025	Acknowledgement that downloading of data records from the PLC to the operating unit has been terminated due to an error.	Check the configuration: - have the tag addresses been set up in the PLC? - does the recipe number exist? - does the data record number exist? - has the function parameters "Overwrite"?
290026	An attempt has been made to read/write a data record although the data mailbox is not free at present. This error may occur in the case of recipes for which downloading with synchronization has been configured.	Set the data mailbox status to zero.
290027	No connection to the PLC can be established at present. Therefore, the data record can neither be read nor written. Possible causes: No physical connection to the PLC (no cable plugged in, cable is defect) or the PLC is switched off.	Check the connection to the PLC.
290030	This message is issued after reselecting a screen that contains a recipe display in which a data record has already been selected.	Reload the data record from the data medium or retain the current values.
290031	While saving, it was detected that a data record with the specified number already exists.	Overwrite the data record or cancel the process.
290032	While exporting data records, it was detected that a file with the specified name already exists.	Overwrite the file cancel the process.
290033	Confirmation request before deleting data records.	-



Number	Effect/Cause	Remedy
290040	A data record error with error code %1 has occurred which cannot be described in more detail. The action was canceled. It might be that the data mailbox has not been installed correctly on the PLC.	Check that the data carrier, the data record, the data mailbox and, if necessary, the connection to the PLC. Trigger the action again after waiting a short period. If the error occurs again, please contact the Customer Support. Specify the error code displayed.
290041	A data record or file cannot be saved because the data medium is full.	Delete files no longer required.
290042	An attempt was made to execute several recipe actions simultaneously. The last action was not executed.	Trigger the action again after waiting a short period.
290043	Confirmation request before storing data records.	-
290044	The data store for the recipe has been destroyed and will be deleted.	-
290050	Acknowledgement that the exportation of data records has started.	-
290051	Acknowledgement that the exportation of data records has been completed successfully.	-
290052	Acknowledgement that the exportation of data records has been terminated due to errors.	Ensure that the structure of the data records on the data medium and the current recipe structure on the operating unit are identical.
290053	Acknowledgement that the importation of data records has been started.	-
290054	Acknowledgement that the importation of data records has been completed successfully.	-
290055	Acknowledgement that the importation of data records has been terminated due to errors.	Ensure that the structure of the data records on the data medium and the current recipe structure on the operating unit are identical.
290056	The value in the specified line/column could not be read/written without errors. The action was canceled.	Check the specified line/column.

Number	Effect/Cause	Remedy
290057	The tags of the recipe specified have been switched from operating mode "offline" to "online". Each modification of a tag in this recipe is now immediately transferred to the PLC.	-
290058	The tags of the recipe specified were switched from operating mode "online" to "offline". Modifications to tags in this recipe are no longer immediately transferred to the PLC but must be transferred there explicitly by means of downloading a data record, if necessary.	-
290059	Acknowledgement that the specified data record has been stored successfully.	-
290060	Check-back message that the data record memory has been successfully erased.	-
290061	Check-back message, that erasing the data record memory was aborted with errors.	-
290068	Request to confirm whether all data records in the recipe should be deleted.	-
290069	Request to confirm whether all data records of all recipes should be deleted.	-
290070	The data record specified is not contained in the import file.	Check the source of the data record number or data record name (constant or tag value).
300000	Process monitoring (e.g. using PDiag or S7-Graph) has been incorrectly programmed: More messages are queued than specified in the technical data of the CPU. No further ALARM_S messages can be managed by the CPU and reported to operating systems.	Change the CPU configuration.
310000	An attempt is being made to print too many protocols simultaneously. Since only one protocol can be printed at a time, the print job is rejected.	Wait until printout of the last active protocol has been concluded. Repeat the print job, if necessary.

Number	Effect/Cause	Remedy
310001	An error occurred on triggering the printer. The protocol is either not printed or printed with errors.	Evaluate the additional system message, related to this message, which is also issued. Repeat the print job, if necessary.
320000	The movements have already been indicated by another device. The movements can no longer be served.	Select the movements on the other display units and select the movement screen on the required display unit.
320001	The network is too complex. The defective operands cannot be displayed.	Display the network in AWL.
320002	No diagnostics-capable alarm messages have been selected. The units related to the alarm messages could not be selected.	Select a diagnostics-capable alarm message in the message screen ZP_ALARM.
320003	No alarm messages exist in respect of the selected unit. No network can be displayed in the detail display.	Select the defective unit in the general view screen.
320004	The required signal statuses could not be read by the PLC. The defective operands cannot be established.	Check the consistency between the configuration on the display unit and the PLC program loaded.
320005	The project contains ProAgent partitions which are not installed. No ProAgent diagnostics can be performed.	In order to run the project, install the ProAgent option packet.
320014	The selected PLC cannot be evaluated for ProAgent. The message display configured with the Evaluate_message_display_fault function could not be found.	Check the parameters of the Evaluate_message_display_fault function.



## B Abbreviations

### Overview

The meanings of the abbreviations used in this documentation are as follows:

ANSI	American National Standards Institute
AS 511	Protocol of the PU interface on the SIMATIC S5
ASCII	American Standard Code for Information Interchange
CF	Compact Flash
CPU	Central Processing Unit
CSV	Comma Separated Values
DP	Distributed I/O
DSN	Data Source Name
HMI	Human Machine Interface
LED	Light-emitting diode
MP	Multi-panel
MPI	Multipoint Interface (SIMATIC S7)
OLE	Object Linking and Embedding
OP	Operator panel
OPC	OLE for Process Control
PC	Personal Computer
PU	Programming unit
RAM	Random access memory: memory with random access (working memory)
PLC	Programmable logic control
TP	Touch panel
TCP/IP	Transmission Control Protocol / Internet Protocol
PRC	People's Republic of China



# Index

## A

Accessing values on the PLC 6-12  
 Acknowledgement 5-147  
 Acknowledgement group 5-115  
 Acknowledging ALARM\_S messages 5-147  
 Acknowledging messages 5-116, 5-117, 5-147  
 Acquisition cycle 5-76  
 Address multiplexing 5-78  
 Addresses 5-91  
 Addresses of tag 5-91  
 ALARM\_S 5-133  
   Acknowledgement 5-147  
   Communication process 5-145  
   Display classes 5-136  
   Print messages 5-148  
   Update 5-142  
 ALARM\_S messages 5-133  
 Analog clock 5-47  
 Analog display 5-44  
 Applying tags in SIMATIC STEP 7 3-8  
 Archive 5-177, 5-178, 5-184  
   Display archive data 5-178  
   Display archive data (example) 5-184  
   Messages (example) 5-180  
   Settings 5-177  
 Archive messages 5-117  
 Archiving 5-174  
 Archiving process data 5-174  
 Archiving tags 5-80  
 Area pointers 4-11, 5-129  
   Data mailbox 5-200  
 Array 5-73  
 Array tags 5-71, 5-73  
 Asian languages 5-252, 5-253  
 Asian projects 5-252, 5-253  
 Assigning icons function keys 5-216  
 Assigning icons to local function keys 5-216  
 Assigning operator permissions 5-223  
 Available PLC drivers 4-9

## B

Bar graph 5-31  
   Overview 5-31

Bit trigger 5-30, 5-99  
 Button 5-34  
 Buttons 5-34  
 Buttons as direct keys 5-37

## C

Changing the dynamic position of screen objects 5-222  
 Changing the operating mode with a current display (example) 5-168  
 Changing the PLC in SIMATIC STEP 7 3-8  
 Clipboard 4-27  
   Copy 4-27  
 Clock 5-47  
 Clock trigger 5-99  
 Color category 5-13  
 Color presentation 5-93  
 Colors 5-13, 5-93  
 Colors of screen objects 5-13  
 Columns of a message view 5-124  
 Combining multiple functions 5-163  
 Communication areas for messages 5-129  
 Communication sequence in the event of ALARM\_S messages 5-145  
 Compatibility 5-201  
 Compatibility of recipes 5-201  
 Compilation 6-13  
 Complete devices 1-5  
 Components of a project 4-4  
 Components of a ProTool project 4-4  
 Configuration 2-3  
 Configuration at run time 4-7  
 Configuration computer 2-4  
 Configuration notes for touch devices 4-17  
 Configure access protection 5-223  
 Configuring a recipe view (example) 5-205  
 Configuring alarm message area (example) 5-150  
 Configuring alarm messages (example) 5-150  
 Configuring archives 5-179  
 Configuring behavior at run time 4-7  
 Configuring color change 5-217  
 Configuring control elements 5-20  
 Configuring display elements 5-20  
 Configuring message editing 5-126  
 Configuring message texts for ALARM\_S 5-133

Configuring multiplex tags (example)	5-85	Importing message text	5-141
Configuring operator prompting	5-212	Date	5-17, 5-22, 5-26, 5-46, 5-47
Configuring password lists	5-225	Display	5-22
Configuring the date and time for Panels	5-48	Enter (overview)	5-26
Configuring the date and time for XP170B (example)	5-66	Date display	5-46
Connecting to STEP 7 Lite	3-10	Date/Time	5-17
Constraints	7-7	Debugging scripts in ProTool	5-231
Constraints on projects in Asian languages	5-253	Decimal places with tags	5-89
Constraints with Asian languages	5-253	Define colors	5-13
Constraints with printing	7-7	Defining functions	5-164
Constraints with VBScript	5-232	Device overview for ProTool	1-5
Contents of message buffers	5-127	Device type	4-37
Control elements	5-20	Digital clock	5-47
Control Panel	6-2	Direct keys	5-37
Conversion functions	5-165	Direct transfer	6-4
Convert	4-23	Data records:	5-199
Key assignment	4-23	Directly accessing values on the PLC	6-12
Convert a project	4-21	Discard	4-35
Convertible projects	4-21	Display	5-111
Converting tags linearly	5-81	Actual values	5-22
Copy	4-27	Display actual values	5-22
Copying objects		Display classes	5-136
Between projects and within a project	4-27	Display classes for ALARM_S messages	5-136
Copying objects within a project	4-27	Display elements	5-20
Copying projects	3-9	Display in "Cross-Reference" window	4-36
Copying projects from SIMATIC STEP 7	3-9	Display message class	5-124
Create a project	4-6	Display messages on the operating unit	5-121
Creating archives for messages (example)	5-180	Displaying and setting date/time	5-17
Creating area pointers	4-11	Displaying archive data in a specified time window	5-178
Creating icons for function keys	5-216	Displaying archive data with a time window (example)	5-184
Creating recipes (example)	5-203	Displaying date and time on the TP 170A (example)	5-49
Creating reports	5-102	Displaying date and time on the TP 170A (example)	5-49
Cross-Reference	4-36	Displaying messages	5-121
Cross-reference (overview)	4-36	Displaying messages on the operating unit	5-121
<b>D</b>		Displaying operating and process states	5-111
Data mailbox	5-194	Displaying process states	5-111
Data mailbox (area pointer)	5-200	Downloading	6-4, 6-14
Data record		Downloading executable project files	6-4
Create	5-199	Downloading the executable project file to operating unit	6-4
Definition	5-185	Driving LEDs	5-218
Export	5-199	Driving light-emitting diodes	5-218
Import	5-199	Dynamic attributes	5-217
Transfer	5-199	Dynamic graphics	5-93
Data records	5-199	Dynamic positioning of screen objects	5-222
Data types	5-91		
STRING	5-88		
Database	5-141, 5-174		



**E**

Editing data records	5-199
Editing language	5-241
Editing projects	4-20
Enter setpoints	5-24
Entering Passwords	5-27
Error messages during compilation	6-13
Error messages while downloading	6-14
Events for triggering functions	5-159
Example of a function	5-237
Example of a function (example)	5-237
Example of a subroutine (example)	5-234
Exchanging text with translators	5-250
Exchanging texts	5-250

**F**

Fields	5-2
Filter messages	5-117
Filtering messages	5-117
Fixed window	4-14, 5-2
Flashing	5-218
Flashing fields	5-217
Fonts	5-15
Force VAR	6-12
Foreign languages	5-241
Export/import	5-250
System requirements	5-240
Full Screen mode	4-14
Function (example)	5-237
Function key	4-23, 5-28
Assign after conversion	4-23
Function keys	5-28
Function parameters	5-161
Functions	5-157
Archive	5-179
Change tags	5-81, 5-83, 5-84
For messages	5-115
For service work	4-19
Single-bit setting of tags	5-82
Functions for the linear	
scaling of tags	5-84
Functions to change tag values	5-83
Functions to change the value	
of a tag	5-83
Functions to set tags bit by bit	5-82
Fundamental considerations when	
creating a project	4-2

**G**

Getting started	2-3
Getting started with Protool	2-3

Global function key	5-28
Graphic displays	1-5
Graphic lists	5-92
Graphics	5-93
Bar graph	5-31
Overview	5-93
Trends	5-30

**H**

Help texts in messages	5-113
Hiding objects	5-216
Hours	5-47
How to create a customized printout	
(example)	7-8
How to create a printout (example)	7-8

**I**

Import	
Text for translation	5-250
Incorporating ALARM_S messages	5-141
Information about the project	4-37
Input field	
Enter date	5-26
Enter time	5-26
Overview	5-24
Input field for concealed	
password entry	5-27
Input field for password entry	5-27
Input fields	5-214
Instance DB	3-1
Integrated operation	7-3
Integrating and unintegrating	
projects in SIMATIC STEP 7	3-9
Integrating projects	3-9
Integrating projects in	
SIMATIC STEP 7	3-9
Internal script tags and ProTool tags	5-230
Invisible button	
Overview	5-34

**K**

Key assignment	4-23
Keyboard assignment	5-244
Language dependence	5-244
Keyboard on screen	4-17

**L**

Labeling function keys	5-216
------------------------	-------

Language dependence	5-244	Message text	5-113
Keyboard assignment	5-244	Message view	5-124
Languages	5-241, 5-253	Overview	5-124
Configurable	5-243	Message view for messages	5-124
Create multi-lingual projects	5-248	Message window	5-121
Export	5-250	Messages	5-121, 5-126
Import	5-250	Acknowledgement	5-147
Requirements for Asian languages	5-252	Communication areas	5-129
Standard screens	5-243	Edit	5-126
System messages	5-243	Edit alarm messages	5-115
Languages for reference texts	5-246	Overview	5-111
Languages in ProTool	5-241	Properties	5-115
LED is flashing	5-218	Structure	5-113
Libraries	5-10, 5-12, 5-19	Use of resources	5-144
In ProTool (overview)	5-10	Messages classes in ProTool	5-111
Libraries in ProTool	5-10	Minutes	5-47
Linear scaling of tags	5-84	Move screen objects	5-222
List box		Multilingual projects	5-246
Overview	5-24	Multiplex tag	5-85
Local function key	5-28	Multiplexing tags	5-79
Logging messages	5-117, 5-120		
Logout	5-223		

## M

Making optimum use of graphics	5-96
Managing projects	7-4
Managing projects in stand-alone operation	7-4
Memory requirements	4-37
Message archive	5-106
Message areas	5-152
Message areas for SIMATIC S5 (example)	5-152
Message areas for SIMATIC S7 (example)	5-153
Message bar	5-121
Message bit procedure	5-131
Message buffer	5-106, 5-127
Print in report	5-106
Message classes	5-117
Message event	5-124, 5-131
Acknowledged	5-131
Active	5-131
Cleared	5-131
Message indicator	4-14, 5-121
Message number	5-113
Message number procedure	
ALARM_S	5-133
Message numbers	5-113
Message procedure	5-130, 5-131
Message bit procedure	5-131
Message number procedure	5-131
Message properties	5-115
Message tags	5-113

## N

Notes on the configuration computer	2-4
-------------------------------------	-----

## O

Objects in the project window	4-5
ODBC	5-174
On-screen keyboard	4-17, 5-214, 5-244
Operating unit	4-14
Operator panel	1-5
Optimize	
System resources	2-4
Optimizing system resources	2-4
Optimum creation of graphics	5-96
Options for configuring archives	5-179
Output field	5-22
Display date	5-22
Display time	5-22
Overview	5-22
Output fields for graphics lists	5-22
Output fields for text lists	5-22
Overview of area pointers	4-11

## P

Page number (screen object)	5-108
Page numbers in reports	5-108
Password administration	5-225
Password hierarchy	5-223
Password level	5-223
Password lists	5-225

Passwords	5-225	ProTool objects	5-229
Pattern trend	5-30, 5-99	Using in scripts	5-229
Peculiarities with conversion		ProTool version	4-37
functions	5-165	Providing Help text	5-213
PLC	4-9	Providing Help text on operating unit	5-213
Select driver	4-9	Providing information text	5-213
PLC data types	5-91		
PLC drivers	4-9	<b>R</b>	
Print	7-7	Read in alphanumeric values	5-24
Project data	7-1	Read in numeric values	5-24
Print messages	5-115	Read in symbolic values	5-24
Printing (example)	7-8	Read in values	5-24
Printing ALARM_S messages	5-148	Read out alphanumeric values	5-22
Printing message archive in a report	5-106	Read out numeric values	5-22
Printing messages	5-115	Read out values	5-22
Printing messages in a report	5-106	Realtime trend	5-30, 5-99
Printing project data	7-1	Recipe	5-198, 5-203
Printing recipes in a report	5-107	Configure	5-194
Procedure function (example)	5-237	Data mailbox	5-194
Procedure sub (example)	5-234	Data record	5-185
Process data	5-174	Enter data records (example)	5-207
Project	4-6, 4-21, 4-26, 6-1, 6-7	Example	5-185
Convert	4-21	Overview	5-185
Converting an MP370 for PC		Set structure	5-194
(example)	4-26	Synchronization with the PLC	5-198
Create	4-6	Recipe printout	5-107
fundamentals of creating	4-2	Recipe screen	5-193
Test	6-1	Recipe view	5-190
Upload	6-10	Screen object	5-190
Project colors	5-13	Recipes	5-185, 5-187, 5-201
Project data	7-1	Application scenarios	5-187
Project Info	4-37	Compatibility	5-201
Project language	5-241	Redo	4-33, 4-35
Project management	7-3	Redoing the last action	4-35
With integrated operation	7-3	Reference language	5-241
Project management with integrated		Reference text	5-246
operation	7-3	Reference text for multilingual	
Project Manager	7-4	projects	5-246
Project window	4-5	Report	5-109
Objects	4-5	Create (example)	5-109
Prompting the operator	5-212	Print Recipe	5-107
Properties of tags	5-71	Report operating states	5-111
ProTool	1-1, 1-5, 2-3, 3-1, 3-3	Reports	5-102
Configuration software	1-1	Requirements for configuring in	
Connecting to STEP 7 Lite	3-10	Asian languages	5-252
Device family	1-5	Resource consumption of messages	5-144
Getting Started	2-3		
Integrated in SIMATIC STEP 7	3-1		
Overview	1-1		
Starting in integrated			
SIMATIC STEP 7	3-3	<b>S</b>	
ProTool for graphical displays	1-8	Scaling tags	5-81
ProTool for text-based displays	1-9	Scenario 2	
ProTool for Windows-based systems	1-6	Manual production process	
ProTool integrated in		(example)	5-209
SIMATIC STEP 7	3-1		

Scenario 3		Slider controls	5-42
Automated production process		Soft key	5-28
(example)	5-211	Special features with conversion	
Scheduler	5-101	functions	5-165
Screen editor	5-2	Standard clock pulse	5-76
Screen objects	5-5, 5-108	Standard colors	5-13
Page number	5-108	Start screen	5-2
Screen objects in ProTool	5-5	Starting ProTool	2-2
Screen partitioning on the		Starting ProTool in SIMATIC STEP 7	3-3
operating unit	4-14	State area	5-32
Screens	5-2	State button	5-35
Components of a screen	5-2	Static graphics	5-93
Example	5-2	Static text	5-97
Overview	5-2	Status tag	6-12
Select screen	5-2	Steps to creating a multilingual	
Script tags (internal) and ProTool tags	5-230	project	5-248
Security through password		STRING tags	5-88
protection	5-223	Storage	5-88
Selecting the display classes	5-138	Structure of an archive for messages	
Service work	4-19	(example)	5-181
Functions	4-19	Structure of an archive for tags	
Set field grid	5-16	(example)	5-87
Set screen partitioning on		Structure of an archive for tags	
operating unit	4-14	(examples)	5-183
Set up user interface	5-16	Styles	5-113
Set zoom factor	5-16	Sub (example)	5-234
Setting date and time on the TP 170A		Subroutine (example)	5-234
(example)	5-55	Substituting a project and project parts	4-30
Setting date/time	5-17	Substituting project parts	4-30
Setting fonts	5-15	Substituting projects	4-30
Setting the column width	5-221	Switch buffer	5-99
Setting the column width and order		Switches	5-40
of screen objects	5-221	Symbol Library	5-12
Setting the column width of		Synchronization	5-62, 5-198
screen objects	5-221	Synchronization with the PLC	5-198
Setting the message procedure	5-138	Synchronizing the date and time	
Setting the order	5-221	with the PLC (example)	5-62
Settings	5-55	Synchronizing the date and time	
Settings for message classes	5-117	with the PLC on TP170A (example)	5-62
Shared database	5-141	System administrator	5-223
Importing message text	5-141	System messages	5-118
Showing and hiding objects	5-216	Languages	5-243
Showing objects	5-216	System messages (example)	5-154
SIMATIC HMI device family	1-5	System requirements	5-240
SIMATIC HMI Symbol Library	5-12	For foreign languages	5-240
SIMATIC Manager	3-6, 7-3		
SIMATIC Manger	3-7	<b>T</b>	
SIMATIC STEP 7	3-1	Tab order	5-219
Starting ProTool integrated	3-3	Tab order of operating unit	5-219
Simulating projects	6-6	Tag	5-76, 5-81, 5-91
Simulation	6-6, 6-7	Address	5-91
Project	6-7	Change value	5-81
Simulation of the project with the		Convert	5-81
simulator	6-7	Functions	5-81
Simulation options	6-6		

Status/Force	6-12		
Update	5-76		
Tag scaling	5-84		
Tags	5-68, 5-71, 5-75, 5-81, 5-89		
Array tags	5-73		
Create	5-71		
Decimal places	5-89		
Define	5-71		
Internal script tags	5-230		
Limit values	5-75		
Recipe	5-194		
Scale Linear	5-75		
Start value	5-75		
Update	5-71		
Tags for integrated operation	5-71		
Tags in scripts	5-82		
Technological alarms	5-133		
Testing projects	6-1		
Testing projects integrated in STEP 7	6-1		
Testing projects with PLC link	6-1		
Testing projects without PLC link	6-1		
Text	5-93		
Export	5-250		
Import	5-250		
Static	5-97		
Text displays	1-5		
Text lists	5-92		
Text or graphic list	5-194		
Recipe	5-194		
Text or graphic lists	5-92		
Text-based displays	1-5		
The components of a message	5-113		
Time	5-17, 5-22, 5-26, 5-46		
Display	5-22		
Enter (overview)	5-26		
Time display	5-46		
Time window	5-178		
Touch panel	1-5		
Touch-operated devices	4-17		
Configuration notes	4-17		
Transfer			
Data records:	5-199		
Translating projects	5-250		
Translating texts for multilingual projects	5-250		
Trend view	5-30		
Overview	5-30		
Trends	5-99		
Triggering functions	5-159		
Triggering trends	5-30		
Type	5-91		
Types of tags	5-68, 5-91		
<b>U</b>			
Undo	4-33, 4-35		
Undo actions	4-33		
Undo buffer	4-36		
Undoing and redoing actions	4-33		
Undoing the last action	4-35		
Update symbol table	3-1		
Updating tags	5-76		
Updating tags (Performance)	5-76		
Updating the operating unit	5-142		
Upload	6-10		
Use of resources	5-144		
Useful functions when editing projects	4-20		
User interface language	5-241		
User-defined functions	5-164		
Using area pointers	4-11		
Using functions	5-157		
Using ProTool objects in VBScript	5-229		
Using tags	5-68		
Using tags in several fields	5-68		
<b>V</b>			
VBScript	5-227		
VBScript in ProTool	5-227		
Vector graphic elements	5-98		
Vector graphics	5-93		
Version	4-37		
Virtual key	5-34		
Visual Basic Script	5-227		
<b>W</b>			
Windows CE Control Panel	6-2		
Windows-based systems	1-5		
Working with a SIMATIC HMI Symbol Library (Example)	5-19		
Working with SIMATIC Manager	3-7		
Working with Windows CE Control Panel	6-2		

