

SIEMENS

SIMATIC STEP 7 V5.4

GMP-Engineering Manual

Guidelines for Implementing Automation Projects in a GMP Environment

Introduction

Contents

Prerequisites for Configuring Automated Systems in a GMP Environment	1
Requirements for Automated Systems in a GMP Environment	2
System Specification	3
Guidelines for Implementing SIMATIC STEP 7 in a GMP Environment	4
Additional Software Components	5
Supporting functions during qualification	6
Index	

04/2007

A5E01100600-01

Safety-Related Notices

Notices that you should observe to ensure your own personal safety and to avoid damage to property and equipment can be found in the relevant technical manuals. The safety of pharmaceutical products of prime importance to the pharmacist must be evaluated by the pharmaceutical company itself. This document provides information on this topic.

Qualified Personnel

Only **qualified personnel** should be allowed to install and work on this equipment. Qualified persons are defined as persons who are authorized to commission, to ground, and to tag circuits, equipment, and systems in accordance with established safety practices and standards.

Introduction

Scope of this manual

This manual describes what is required of the system, the software and the procedures for configuring SIMATIC STEP 7 from a Good Manufacturing Practice (GMP) perspective. The relationship between the requirements and implementation is explained based on practical examples.

Intended Audience

The manual is intended for all plant operators, persons responsible for branch-specific control system concepts, project leaders and programmers and maintenance or service personnel who use control systems in a GMP environment. It describes approaches for the implementing automation solutions with SIMATIC STEP 7 in situations where the principles of GMP are mandatory.

Basic knowledge required

Basic knowledge about SIMATIC STEP 7 / SIMATIC S7 is required to understand this manual. Knowledge of GMP as practiced in the pharmaceutical industry is also an advantage.

Disclaimer

This manual is a guideline for system users and programmers that will help them to integrate SIMATIC S7 programmable controllers (PLCs) and programming devices in a GMP environment with regard to validation while giving consideration to special aspects such as 21 CFR Part 11 (CFR – Code of Federal Regulations).

We have checked the contents of this manual for agreement with the hardware and software described. Since deviations cannot be precluded entirely, we cannot guarantee full agreement. The information in this document is checked regularly for system changes or changes to the regulations of the various organizations and necessary corrections will be included in subsequent issues. We welcome any suggestions for improvement and ask that they be sent to the Competence Center Pharma in Karlsruhe (Germany).

Validity of the manual

The information in this manual is valid for the SIMATIC S7-300/400 hardware and the SIMATIC STEP 7 V5.4 + SP1 engineering software. The components investigated are SIMATIC STEP 7 in combination with the options SIMATIC Logon, SIMATIC Version Trail, and SIMATIC Version Cross Checker. Information on the compatibility of individual components with SIMATIC STEP 7 V5.4 can be found in the STEP 7 compatibility list that is written to the folder x:\...Files\Siemens\Information\English when STEP 7 is installed or at <http://support.automation.siemens.com/> under entry number ID 18734363. The CD-ROM catalog can be ordered over the Internet at www.siemens.com/automation/ca01. The online catalog is available on the Internet at <https://mall.automation.siemens.com/>.

Where this documentation fits in

The system documentation of the programmable controllers and the SIMATIC STEP 7 engineering software are an integral part of the SIMATIC STEP 7 system software. This is available as online help (HTML help system) or as electronic documentation in Acrobat Reader format (PDF):

- SIMATIC S7/STEP 7 V5.4 electronic manuals

The electronic manuals for SIMATIC are available on the Internet at <https://support.automation.siemens.com/> or in the “SIMATIC MANUAL COLLECTION” on DVD order number 6ES7998-8XC01-8YE0.

Structure of the manual

This manual supplements the existing SIMATIC S7/STEP 7 documentation. It is useful not only during the configuration, but is also intended to provide an overview of the requirements for configuration and what is expected of automation systems in a GMP environment.

The rules and guidelines, recommendations and mandatory specifications are explained, that represent the basis for configuration of automation systems.

All the necessary functions and requirements for hardware and software components are also described and this should make the selection of components easier.

Based on examples, the use of the hardware and software is explained briefly and how they are configured or programmed to meet the requirements. More detailed explanations can be found in the standard documentation.

The appendix to this manual contains an index table.

Conventions

The following conventions are used in this manual.

Activities involving several steps are numbered in the order in which the activities should be performed.

Procedures involving few tasks are presented with a bullet (•).

References to other manuals are shown in bold italic.

Menu commands are shown in **bold** face.

Additional support

Please talk to your Siemens contact at one of our agencies or local offices if you have any questions about the products described here and do not find the answers in this manual.

You will find your contact person at:

<http://www.siemens.com/automation/partner>

You will find the guidelines to the range of technical documentation available for individual SIMATIC products and systems as follows:

<http://www.siemens.de/simatic-tech-doku-portal>

You will find the online catalog and the online ordering system at:

<http://mall.automation.siemens.com/>

If you have questions on the manual, please contact the Competence Center Pharma:

Guidelines for Implementing automation projects in a GMP environment

A5E01100600-01

Email: pharma.aud@siemens.com

Fax: +49 721 595 6390

Further information about the products, systems and services from Siemens for the pharmaceutical industry can be found at:

<http://www.siemens.com/pharma>

Training center

Siemens offers a number of training courses to familiarize you with the SIMATIC S7 and SIMATIC STEP 7 operator control and monitoring system. Please contact your regional Training Center, or the central Training Center in D 90327 Nuremberg.

Phone: +49 (911) 895-3200.

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

Technical Support

You can reach technical support for all A&D projects

- With the Support Request form on the Web:
<http://www.siemens.de/automation/support-request>
- Phone: + 49 180 5050 222
- Fax: + 49 180 5050 223

Further information about our technical support is available on the Internet at

<http://www.siemens.de/automation/service>

Service & support on the Internet

In addition to our pool of documentation, we offer you a comprehensive knowledge base on the Internet.

<http://www.siemens.com/automation/service&support>

There you will find:

- Our newsletter, providing you with the latest information about your products.
- The right documents for you, using our Service & Support search engine.
- A forum where users and experts from all over the world exchange ideas
- Your local Automation & Drives representative.
- Information about on-site service, repairs and spare parts. And lots more under "Services".

Table of contents

Introduction	iii
Table of contents	vii
1 Prerequisites for Configuring Automated Systems in a GMP Environment	1-1
1.1 Life Cycle Model	1-2
1.2 Regulations and Guidelines	1-7
1.3 Responsibilities	1-9
1.4 Approval and Change Procedure	1-9
1.5 Software Categorization of Control Systems	1-10
2 Requirements for Automated Systems in a GMP Environment	2-1
2.1 Hardware Categorization	2-2
2.2 Software Categorization	2-2
2.3 Configuration Management	2-4
2.3.1 Configuration Identification	2-4
2.3.2 Configuration Control	2-4
2.3.2.1 Version Control	2-4
2.3.2.2 Change Control	2-5
2.4 Software Creation	2-6
2.4.1 Use of Typical for Programming	2-6
2.4.2 Identification of Software Modules / Typical	2-6
2.4.3 Changing Software Modules / Typical	2-6
2.5 Access Protection and User Management	2-7
2.5.1 Using Access Protection in a System	2-7
2.5.2 Requirements for the User ID and Password	2-7
2.5.3 Smart Cards and Biometric Systems	2-7
2.6 Electronic Signatures	2-8
2.6.1 Conventional Electronic Signatures	2-8
2.6.2 Electronic Signatures Based on Biometrics	2-8
2.6.3 Security Measures for User IDs/Passwords	2-9
2.7 Audit Trail	2-10
2.8 Time Synchronization	2-10
2.9 Data Backup	2-11
2.9.1 Application Software	2-11
Process Data	2-12
2.10 Retrieving Archived Data	2-13
2.11 Use of Third-Party Components	2-13
3 System specification	3-1
3.1 Dimensioning of the hardware	3-2
3.2 Selection criteria for hardware	3-3
3.3 Required/optional software packages	3-5
3.3.1 Basic engineering software	3-5
3.3.2 Additional engineering software	3-5
3.3.3 Additional software packages for support of GMP compliance	3-8
3.3.3.1 Access Control	3-8
3.3.3.2 Versioning, Change control , Audit trail	3-8

4	Guidelines for implementing SIMATIC STEP 7 in a GMP environment	4-1
4.1	Introduction	4-1
4.2	Software categorization of STEP 7	4-2
4.3	Software installation	4-3
4.4	Basic engineering principles	4-4
4.4.1	Software creation	4-4
4.4.1.1	Procedure for programming	4-4
4.4.1.2	Rules and conventions	4-8
4.4.1.3	Software interlocks/safety	4-10
4.4.2	Integrated HMI system	4-11
4.4.3	Software documentation	4-13
4.5	Configuration management	4-18
4.5.1	Changing the system software	4-18
4.5.2	Replacing/changing the hardware/firmware	4-21
4.5.3	Versioning of the application software	4-22
4.5.4	Change control	4-25
4.6	Access protection	4-29
4.6.1	Access protection to the CPU	4-29
4.6.2	Access protection for a STEP 7 project	4-30
4.6.3	Protective Measures in the software	4-32
4.7	Audit Trail	4-35
4.8	Time synchronization	4-36
4.9	Time stamping	4-38
4.10	CPU storage	4-39
4.10.1	Memory concept of S7-300 CPUs	4-39
4.10.2	Memory concept of S7-400 CPUs	4-40
4.11	Backup / restoring system/application software	4-41
5	Additional software components	5-1
5.1	Diagnostic tools	5-1
5.2	Simulation tools	5-5
5.3	SIMIT simulation software	5-6
5.4	Rewiring S7 programs	5-7
6	Supporting functions during qualification	6-1
6.1	Introduction	6-1
6.2	Qualification of automation hardware	6-2
6.3	Qualification of automation software	6-4
6.3.1	Qualification of standard software	6-4
6.3.2	Installed SIMATIC software STEP 7	6-4
6.3.3	Installed licenses of SIMATIC STEP 7	6-5
6.3.4	Qualification of the application software	6-7
Index		Index-1

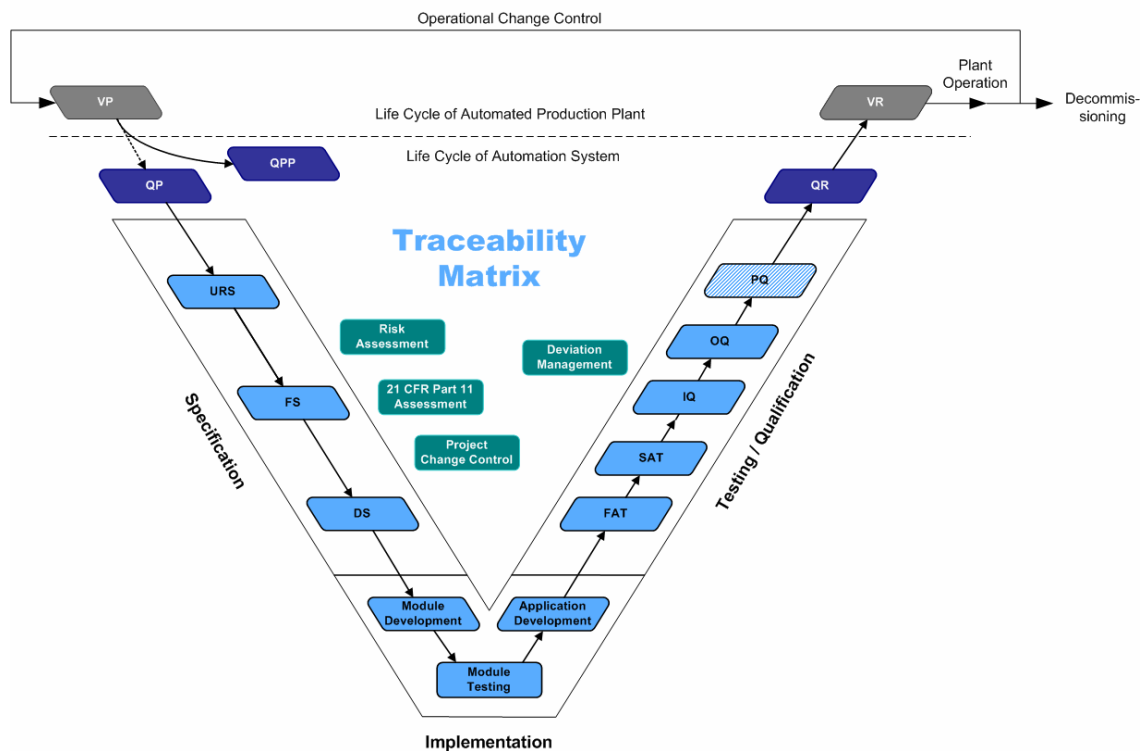
1 Prerequisites for Configuring Automated Systems in a GMP Environment

Before automated systems can be configured in a GMP Environment, approved specifications such as the user requirements and functional specification must be available. When creating these specifications, requirements stipulated in standards, recommendations and guidelines must be taken into account. This chapter lists the most important of these regulations as well as various specifications (URS, FS, DS).

1.1 Life Cycle Model

Good Engineering Practice (GEP) is defined as the application of acknowledged engineering methods within the framework of a defined life cycle. The aim is to provide a suitable and cost-effective solution in keeping with the requirements.

The following graphic shows the life cycle model used in this manual. It is oriented on the recommendations of the current GAMP® guideline for validation of computerized systems. It begins with the planning phase of a project and ends with the start of pharmaceutical production on completion of qualification and validation.



Key to the life cycle model

Abbreviation/Acronym	Description
VP	Validation Plan
QP	Qualification Plan
QPP	Quality and Project Plan
URS	User Requirement Specification;
FS	Functional Specification; also: Application specification
DS	Design Specification
FAT	Factory Acceptance Test
SAT	Site Acceptance Test
IQ	Installation Qualification
OQ	Operational Qualification
PQ	Performance Qualification
VR	Validation Report
QR	Qualification Report

Validation plan

The validation plan (VP) specifies the overall strategy and specifies the parties responsible for the validation of a system in its operational environment [PDA, GAMP®4].

- In the case of complex plants (for example a production line with several process cells and automation systems), there may also be a master validation plan (MVP) as well as VPs valid only for specific process cells and systems.
- See also GAMP®4, Appendix M1, "Guideline for Validation Planning".

Quality and project plan

The quality and project plan (QPP) defines the framework and procedures for project and quality management. It specifies, for example, procedures for managing documents and procedures for change control etc. The QPP defines the life cycle in such a way that it integrates not only the product phases relevant for validation but also other organizational relationships (for example, the various time plans of different trades).

Due to their similar structures and contents, a combination of the QPP and QP is possible.

- See also GAMP®4, Appendix M6, "Guideline for Quality and Project Planning".

Qualification plan

In contrast to the validation plan, a qualification plan (QP) describes the qualification activities in detail. It defines the tests to be performed and indicates the dependencies.

The qualification plan follows a validation plan. Due to the similar contents of both documents, it is possible to combine the QP and the QPP.

Specification

The specification phase begins with the creation of the user requirements specification. The user requirements specification is normally created by the user and describes the requirements that the system should meet. On completion of the user requirements specification, the functional specification is created, usually by the supplier. The functional specification (FS) provides detailed information on the requirements defined in the URS at the functional level. This is followed by the detailed specification for implementation in the design specification (DS).

The functional specification and design specification also form the basis for subsequent tests within the framework of the qualification or validation. The following points must also be specified in the functional and design specification phases:

- Software structure
- Programming standards
- Naming convention
- File naming convention

User requirements specification (URS)

The user requirements specification describes the requirements to be met by the system from the perspective of the user. The user requirements specification is generally created by the system user possibly with the support of the system supplier.

It is the basis of all subsequent specifications.

- See also GAMP®4, Appendix D1 "Example Procedure for the Production of a User Requirement Specification".

Functional specification (FS)

The functional specification is generally created by the system supplier with some cooperation from the end customer. Based on the user requirements specification, it describes the functions of the system in detail. The approved functional specification is the basis for creating detailed specifications.

- See also GAMP®4, Appendix D2 "Example Procedure for the Production of a Functional Specification".

Design specification (DS)

The design specification (DS) is generally created by the system supplier. It is based on the functional specification and expands this with detailed descriptions, for example, of the hardware and software to be used, process tag lists, etc.

- See also GAMP®4, Appendix D3 "Example Procedure for the Production of a Hardware Design Specification" and GAMP®4, Appendix D4 "Example Procedure for the Production of Software Design Specifications and Software Module Design Specifications".

Implementation

During the implementation phase, the system is implemented according to the design specification. Apart from the procedures and additional guidelines defined in the QPP (for example, coding standards, naming conventions, data backup), change management plays an important role making changes and deviations from the original specification traceable.

See also GAMP®4, Appendix M8 "Guideline for Project Change Control"; GAMP®4, Appendix M10 "Guideline for Document Management".

FAT

On completion of the implementation, a factory acceptance test (FAT) is often performed at the supplier's site. The purpose of this is to find and eliminate any errors in the programming prior to delivery.

The aim of the FAT is the acceptance by the customer to allow the system to be delivered in the tested status.

SAT

The site acceptance test (SAT) shows that an automated system works within its target operating environment with interfaces to the instrumentation and plant sections according to the specification. Depending on the project, the SAT can be combined with commissioning (and therefore with the IQ or OQ).

Test phase / qualification

The FAT is followed by the technical commissioning (commissioning phase). In this phase, the system along with the user program that has been created is installed at the system user's site, the technology is commissioned, tested and qualified.

The commissioning phase and qualification phases can run sequentially or can be combined. It is advisable to synchronize the activities of commissioning and qualification to save both time and costs.

The test planning should therefore be created in good time so that it is possible to check whether or not tests made beforehand during FAT or SAT need to be repeated during qualification. In this case, the documented FAT / SAT tests become part of the qualification documentation.

When creating the test documentation, tests and acceptance criteria must be described so that they are easy to understand.

Qualification report

Based on the qualification plan, the qualification report (QR) sums up the test results of the tests performed and confirms the successful completion of the qualification phases.

Validation report

The validation report (VR) sums up the results of the individual validation steps and confirms the validated status of the system. The creation of both the validation plan and the validation report is the responsibility of the customer.

Operation

- Following successful qualification and subsequent operation (start of production) of the system, the plant must be serviced and maintained by the user. The maintenance and servicing cycles and the procedure for operational change control must be defined and adhered to.

1.2 Regulations and Guidelines

When configuring automated systems requiring validation in a GMP environment, the recommendations and guidelines of various organizations should be adhered to. These are usually based on general guidelines such as Title 21 Code of Federal Regulations (21 CFR) of the American Food and Drug Administration (FDA) or the EU GMP Guideline Annex 11.

Regulation / Guideline	Issued by / Organization	Title	Regulation / Recommendation	Where Applicable
21 CFR Part 11	US FDA	Electronic records, electronic signature	Regulation	Manufacturers and importers of medicines for the US market
21 CFR Part 210	US FDA	Current good manufacturing practice in manufacturing, processing, packing, or holding of drugs; General		
21 CFR Part 211	US FDA	Current good manufacturing practice for finished pharmaceuticals		
Annex 11 of the EU GMP Guideline	European Commission Directorate General III	Computer-aided Systems	Guideline	Europe
Annex 18 of the EU GMP Guideline	European Commission Directorate General III	Good Manufacturing Practice for Active Pharmaceutical Ingredients	Guideline	Europe
GAMP ® 4	ISPE	GAMP ® 4 Guide for Validation of Automated Systems	Guideline	Worldwide
NAMUR Recommendation NE 58	NAMUR	Execution of Process Control Projects Subject to Validation	Recommendation	Europe
NAMUR Recommendation NE 71	NAMUR	Operation and Maintenance of Validated Systems	Recommendation	Europe
NAMUR Recommendation NE 72	NAMUR	Validation Support by Use of Control Systems	Recommendation	Europe

Note

This manual is based on the requirements of GAMP ® 4 and US 21 CFR Part 11.

Code of Federal Regulations Title 21 (21 CFR), Food and Drugs

The Code of Federal Regulations, Title 21 includes parts such as Parts 11, 210 and 211. For computerized systems, Part 11 (known as: 21 CFR Part 11) is particularly important. It deals with electronic records and electronic signatures.

Annex 11 of the EU GMP Guideline

Annex 11 of the EU GMP guideline is divided into 19 points and covers topics ranging from requirements for configuration, operation and change control for computerized systems in a GMP environment. An interpretation of Annex 11 can be found in the GAMP ® 4 Guide in the form of an APV guideline for the validation of automated systems.

Annex 18 of the EU GMP Guideline

Annex 18 of the EU GMP guideline deals with good manufacturing practice for active pharmaceutical ingredients. This is intended as a GMP manual for the manufacture of active pharmaceutical ingredients within the framework of a suitable quality management system. Chapter 5 of Annex 18 deals with the process equipment and its use.

GAMP ® Guide for Validation of Automated Systems "GAMP ® 4"

The GAMP ® (Good Automated Manufacturing Practice) Guide for Validation of Automated Systems was compiled as a recommendation for suppliers and as a manual for users of automated systems in the manufacturing pharmaceutical industry. The current version "GAMP ® 4" was published in December 2001.

NAMUR Recommendations

NAMUR Recommendations are reports of the experience that were produced by the "Process Control Systems Special Interest Group of the Chemical and Pharmaceutical Industry" for optional use by their members. They do not have the status of standards or directives. The following NAMUR recommendations are of particular interest with regard to configuration and the use of automated systems in a GMP Environment:

- NE58 "Execution of Process Control Projects Subject to Validation"
- NE71 "Operation and Maintenance of Validated Systems"
- NE72 "Validation Support by Use of Control Systems"

1.3 Responsibilities

When configuring automated systems in a GMP environment and creating the corresponding specifications, the responsibilities for the activities in the individual life cycle phases must be specified. Since this is normally decided with a specific customer and for a specific project and must be contractually agreed, it is advisable to specify these responsibilities in the quality and project plan. See also GAMP®4, Appendix M2.

1.4 Approval and Change Procedure

When setting up new systems that require validation or when changing systems already in operation and subject to validation, the main priority is to achieve and maintain the validated status.

Setting up new systems

When a new system is being set up, the approval of documents and the transition between life cycle phases is specified before the project is started. This is usually done along with the definition of responsibilities in the quality and project plan. A life cycle as described in section 1.1 "Life Cycle Model" is used.

Changing validated systems

Changes to an existing system with validated status are governed by the operational change control. Changes must be described before they are implemented, potential effects must be identified and accompanying measures (for example running tests, updating the as-built documentation) must be defined. Following subsequent approval, both the planned change and the defined measures are implemented.

If the changes are extensive, a life cycle like the one shown in this manual can be used if necessary.

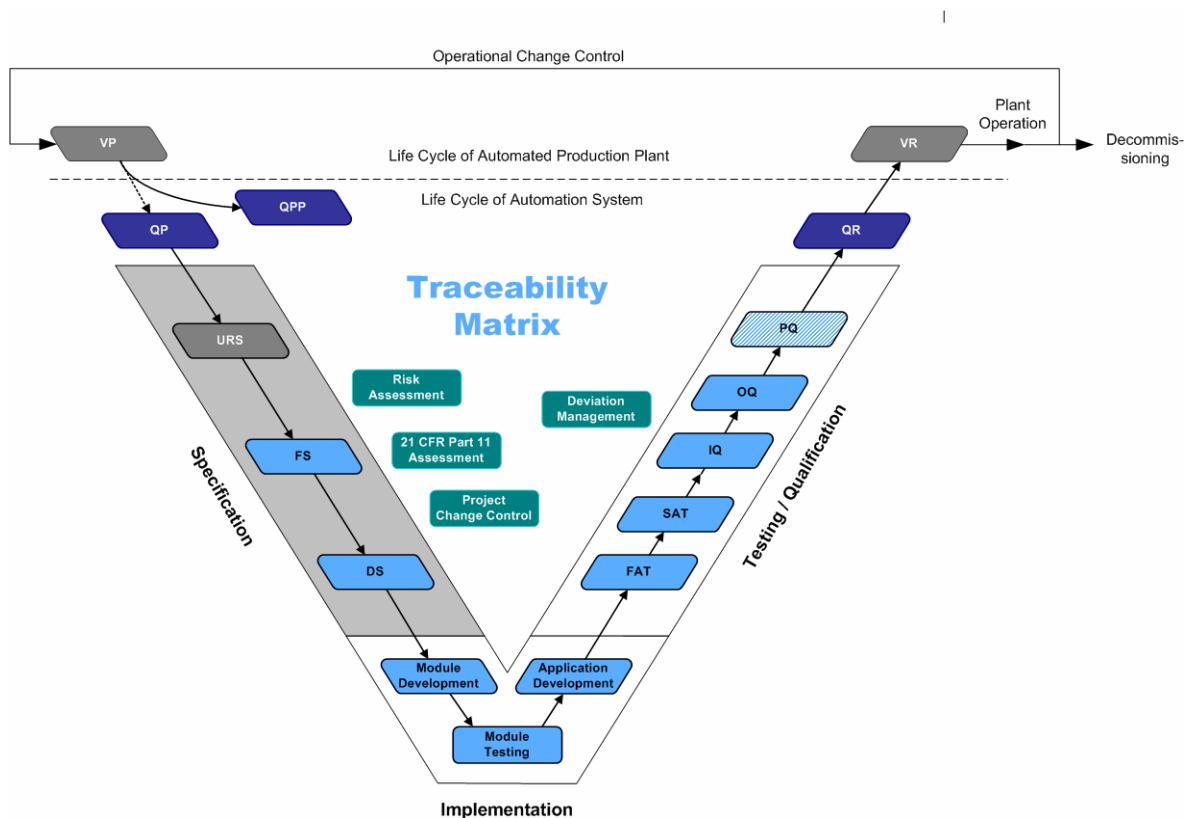
1.5 Software Categorization of Control Systems

As described in section 2.2 "Software Categorization" and section 4.2 "Software categorization of STEP 7", the software of a system can be divided into five software categories according to the GAMP ® Guide for Validation of Automated Systems. The software categories have a major influence on the effort involved during the test and qualification phase and should be defined during the specification phase for the software to be used.

2 Requirements for Automated Systems in a GMP Environment

In the context of GMP, automated systems must meet certain requirements. Section 2 "Requirements for Automated Systems in a GMP Environment" lists the main requirements that an automated system must meet in a GMP environment. These requirements must be stipulated in the specification and implemented during configuration. In general, it must always be ensured that proof of all changes (who did what, when, to change what) is recorded at all times ("why" is optional). The requirements involved in this task are implemented by various functions and are described in the following sections.

The graphic below shows the life cycle model. The requirements focused on in this section can be assigned to the *Specification* area in the graphic.



2.1 Hardware Categorization

According to the GAMP ® 4 Guide, Appendix M4, hardware components of a system are divided into two categories. The hardware categories are listed below:

Category 1, Standard Hardware

Category 1, standard hardware, covers established commercially available hardware components. This hardware must also be subjected to relevant quality and test mechanisms.

The hardware is accepted and documented by the IQ test.

Category 2, Custom Built (Bespoke) Hardware

The functionality must be specified and tested and documented in detail in suitable documented tests.

2.2 Software Categorization

According to the GAMP ® Guide for Validation of Automated Systems, the software components of a system can be divided into five software categories. The five GAMP ® software categories are listed below:

Category 1, Operating Systems

Category 1, operating systems, covers established commercially available operating systems. These are not subject to validation themselves, the name and version of the operating system must, however, be documented and verified during installation qualification (IQ).

Category 2, Firmware

Category 2 includes the firmware, for example in field instruments or compact controllers, whose configuration was adapted to the on-site conditions. Once again the name and version of the firmware and its configuration must be documented and verified during an installation qualification (IQ). The functionality of the device must be verified in an operational qualification (OQ).

Category 3, Standard Software Packages

Category 3 covers commercially available, standard software packages and "off-the-shelf" solutions for certain processes. The configuration of these software packages should be limited to adaptation to the runtime environment (for example network and printer connections) and the configuration of the process parameters. The name and version of the standard software package should be documented and verified in an installation qualification (IQ). User requirements, such as security, alarms, messages, or calculations must be documented and verified within the framework of the operational qualification (OQ).

Category 4, Configurable Software Packages

Category 4 covers configurable software packages that allow special business and manufacturing processes. This involves configuring predefined software modules. These software packages should only be considered as category 4 if they are well known and fully developed, otherwise category 5 is more suitable. In the case of critical and / or complex applications, a supplier audit is usually required

The name, version, and configuration must be documented and verified in an installation qualification (IQ). The functions of the software packages should be verified in terms of the user requirements in an operational qualification (OQ). The validation plan should take into account the life cycle model and an assessment of suppliers and software packages.

Category 5 Custom (Bespoke) Software

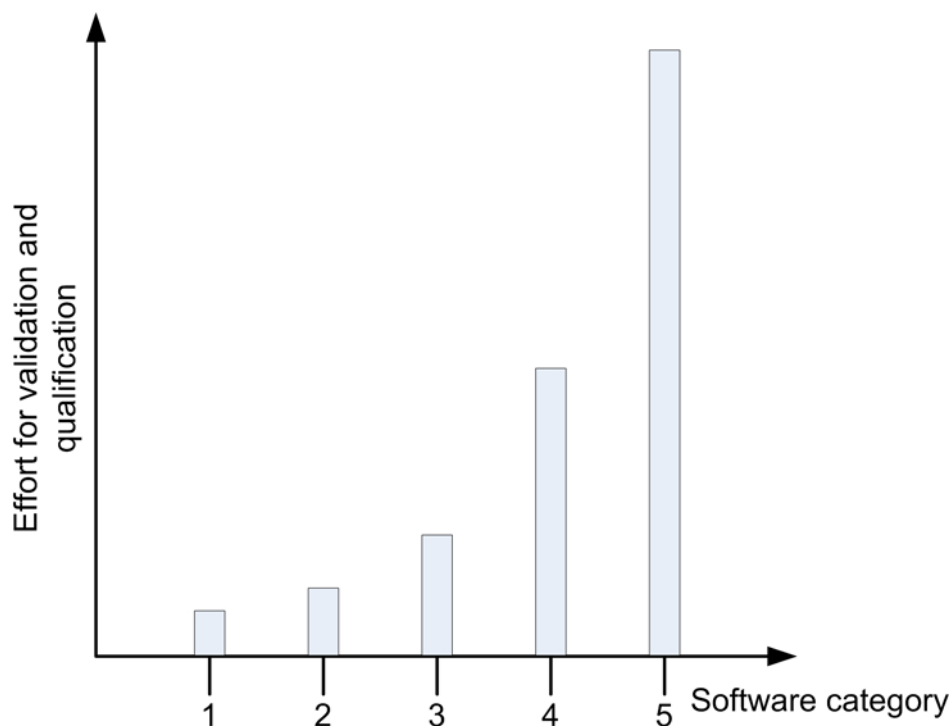
Category 5 covers custom software developed specifically to meet the needs of the user company.

A supplier audit is normally required to confirm the quality systems to control development and subsequent maintenance. Otherwise, suppliers should use the GAMP ® 4 guide as the basis for their own development life cycle.

The name, version, and configuration should once again be documented and verified in an installation qualification (IQ). A detailed software specification must be created and the function of the software verified in an operational qualification (OQ). The validation plan should specify a full life-cycle approach to validation.

The testing of software in higher categories involves far more effort than the software in lower categories.

The effort for validation and testing can be reduced by using standardized software whenever possible.



2.3 Configuration Management

According to the GAMP ® 4 Guide, configuration management is defined as the activity necessary to define an automated system precisely at every point in its life cycle from the first steps in development to its retirement.

Configuration management consists of the application of administrative and technical procedures through the life cycle of a system to:

- Identify and define basic system components and to specify them in general
- Control modifications and releases of items
- Record and report the status of the items and modifications to them
- Ensure the completeness, consistency, and correctness of the items
- Control storage, handling, and delivery of items.

Configuration management consists of the following activities:

- Configuration identification (WHAT is to be kept under control)
- Configuration control (how the control will be implemented)
- Configuration status accounting (how the control will be documented)
- Configuration evaluation (how the control will be verified).

This chapter covers the activities of configuration identification and configuration control.

2.3.1 Configuration Identification

Version and change management is only practicable with a suitable configuration environment. Each software and hardware package is therefore identified by a unique product identifier (MLFB number) and a version number. For the user software, the parts of an automated system that are subject to configuration management must be clearly specified. The system should therefore be organized according to configuration items. These should be defined at an early phase of development so that a complete list of configuration items can be created and maintained. The application-specific elements should have a unique identifier (name or identification number). The degree of detail when specifying the elements is determined by the needs of the system and the supplier developing the system.

2.3.2 Configuration Control

The upkeep of the configuration items should be checked at regular intervals, for example in reviews. Here, particular attention must be paid to the change control and the related version control. Archiving and release of individual configuration items should also be taken into account.

2.3.2.1 Version Control

To ensure correct change management, the configuration elements must be versioned. The version must be updated with every change.

2.3.2.2 Change Control

During configuration, there must be suitable control mechanisms that achieve transparency by documenting any changes. The control mechanisms are described by SOPs and should include the following points.

- Software versioning
- Information such as programming guidelines, naming conventions, etc.
- Guarantees of the traceability of program code changes
- Unequivocal identification of software and all the components it contains

2.4 Software Creation

When creating software, guidelines documented in the quality and project plan must be adhered to (Good Engineering Practice – GEP - awareness). Guidelines on software creation can be found in the GAMP ® 4 Guide for Validation of Automated Systems and in the relevant standards and recommendations.

2.4.1 Use of Typicals for Programming

As seen in section 2.2 "Software Categorization", the validation effort increases considerably from GAMP ® software category to category. While the validation effort for software of category 1 simply involves checking software names and versions, the effort for validation of software in category 5 involves verification of the entire range of functions and a supplier audit.

To keep validation work to a minimum, standardized function blocks should therefore be used during configuration (products, standard company components, standard project components). User-tailored typicals are created from standard function blocks and tested according to design specifications.

2.4.2 Identification of Software Modules / Typicals

During software creation, individual software modules should be given a unique name, version number, and a brief description of the corresponding block. Changes to software modules should be reflected in the identification.

2.4.3 Changing Software Modules / Typicals

Changes to software modules should be indicated in the identification of the relevant module. Apart from the incremented version ID, the date and name of the person making the change should also be included in the software module identification. The software modules to be changed, must, where necessary, be made known by comments with a reference to the corresponding change request. See also section 4.5.4 "Change control".

2.5 Access Protection and User Management

To guarantee the security of automated systems in the context of GMP, these systems should be provided with an access control system. In addition to physical access control (locked rooms etc.), access control systems also provide the option of protecting systems from unauthorized access. Users should be assembled in user groups, which are used to manage the user permissions. The access rights of individual users can be established in different ways:

- A combination of unique user ID and password - a description of the configuration can be found in Chapter 4.6 "Access protection".
- Smart cards in conjunction with a password
- Biometric systems

To ensure security, the assignment and management of the access permissions should be controlled by the plant owner or by an administrator named by the user.

2.5.1 Using Access Protection in a System

Actions that can be performed on an automated system should always be protected. Depending on the task, the user can be assigned various permissions. Access to user administration should only be possible for the plant owner or an employee named by the system owner. Access by unauthorized persons to the recording of electronic data must be prevented.

An automatic logout function should be installed in the system. The logout time should be defined in consultation with the user and stipulated in the functional specification.



Note

It is important to make sure that only authorized persons can access PCs. This can be achieved by suitable mechanisms such as remote kits.

2.5.2 Requirements for the User ID and Password

User ID:

The user ID of a system should have a minimum length agreed with the customer and should be unique within the system.

Password:

A password should always consist of a combination of numeric and alphanumeric characters. When setting up passwords, the number of characters and a period after which a password expires should be stipulated. The structure of the password is normally selected to suit the specific customer. The configuration is described in the section 4.6 "Access protection".

Criteria for the form of a password are as follows:

- Minimum length of the password
- Use of numeric and alphanumeric characters
- Case sensitivity

2.5.3 Smart Cards and Biometric Systems

Apart from the traditional methods of identification with a user ID and password, users can also identify themselves with smart cards or with biometric systems, such as fingerprint scanners.

2.6 Electronic Signatures

Electronic signatures are computer-generated character strings that count as the legal equivalent of a handwritten signature.

The regulations for the use of electronic signatures are set out in 21 CFR Part 11 of the FDA.

Each electronic signature must be assigned uniquely to one person and must not be used by any other person.

It must be possible to confirm to the authorities that an electronic signature represents the legal equivalent of a handwritten signature.

Electronic signatures can be biometrically based or the system can be set up without biometric features.



Note

When exporting pharmaceuticals into the USA, the regulations according to 21 CFR Part 11 of the FDA must be adhered to.

2.6.1 Conventional Electronic Signatures

If electronic signatures are used that are not based on biometrics, they must be created so that persons executing signatures must identify themselves using at least two identifying components. This also applies in all cases in which a smart card replaces one of the two identification components.

These identifying components, can, for example, consist of a user identifier and a password. The identification components must be assigned uniquely and must only be used by the actual owner of the signature.

When owners of signatures want to use their electronic signatures, they must identify themselves with at least two identification components. The exception to this rule is when the owner executes several electronic signatures during one uninterrupted session. In this case, persons executing signatures need to identify themselves with both identification components only when applying the first signature. For the second and subsequent signatures, one unique identification component (password) is then adequate identification.

2.6.2 Electronic Signatures Based on Biometrics

An electronic signature based on biometrics must be created in such a way that it can only be used by one person. If the person making the signature does so using biometric methods, one identification component is adequate.

Possible biometric recognition systems include systems for scanning a fingerprint or the iris of the eye.

Note

The use of biometric systems is currently considered a secure identification method. Nevertheless, there are reservations about the use of biometric identification characteristics in the pharmaceutical industry (for example poor face recognition due to protective clothing covering the face, no fingerprint scans with gloves, the expense involved and the reaction times of retina scans).

2.6.3 Security Measures for User IDs/Passwords

To guarantee the security of electronic signatures when using a user ID and password, the following points are important:

- Uniqueness of the user ID and password
- Supervised issue of user IDs
- Cancellation of rights if a user ID or password is not secure or compromised
- Security measures to prevent unauthorized use of user IDs / passwords and to report misuse
- Training of personnel with documented proof of training courses

2.7 Audit Trail

The audit trail is a control mechanism of the system that allows all data entered or modified to be traced back to the original data. A reliable and secure audit trail is particularly important in conjunction with the creation, change or deletion of GMP-relevant electronic records.

In this case, the audit trail must archive and document all the changes or actions made along with the date and time. Typical contents of an audit trail must be recorded and describe the procedures who changed what (old value/new value) and when.

The archiving period must match the period stipulated in the specification.

There must be adequate hard disk space to allow the entire audit trail to be stored until the next transfer to an external data medium.

Systems must be used that ensure adequate data security (for example redundant systems, standby systems, RAID 5).

2.8 Time Synchronization

Within a system, a uniform time reference must be guaranteed to allow messages, alarms, etc. to be archived with unequivocal time stamps. Time synchronization to a standard time is desirable, however not absolutely necessary. Time synchronization when archiving data and analyzing problems in a plant is strongly recommended.

2.9 Data Backup

In contrast to the archiving of electronic data, data backups are used to create backup copies which allow the system to be restored if the original data or entire system is lost.¹

The backup procedure must cover the periodic backup of volatile information to avoid total loss of data due to defective system components or inadvertent deletion of data. Backup procedures must be checked to ensure the correct storage of data. Backup records should be labeled clearly and intelligibly and dated.²

Data backups are created on external data media. The data media used should comply with the recommendations of the device manufacturer.

When backing up electronic data, a distinction is made between software backups (for example application software, partition images) and archive data backups.

Here, particular attention is paid to the storage of data backup media (storage of the copy and original in different locations, protection from magnetic fields, and elementary damage).

2.9.1 Application Software

Software backups should be created following any software change to the system. They must document the last valid software version of a system. If changes are made to software components, it is sufficient to back up the modified components of the application software. A complete backup of the software should nevertheless be made at regular intervals. If software backups need to be created when changes are made to the software of an existing system or during the installation of a new system, they should be created after the installation. During the course of a project, the software version should be backed up and documented in conjunction with defined milestones, for example at the end of the FAT (in other words before the system is supplied), on completion of the installation qualification (IQ) as a basis for the tests for operational qualification (OQ) and, of course, on handover of the system to the user.

Software generations should also be recorded during the creation of new software versions at regular intervals in the form of software backups.

Software backups must be created for both the application software and the configuration parameters.

Labeling software backups

According to the GAMP ® 4 Guide for Validation of Automated Systems, software backups should be documented both on the label of the backup medium itself and in a separate report containing the following information:

- Date of creation
- System designation
- Software designation
- Software or version designation

¹ "Good Practice and Compliance for Electronic Records and Signatures. Part 1, Good electronic records management". ISPE/PDA 2001.

² "Electronic records and electronic signatures assessment". Chris Ride & Barbara Mullendore, PDA 2001.

- Current number of the backup
- Reason for software backup
- Date of first usage
- Date of backup
- Date and signature of the person responsible
- Identity of the operator

Retention of software backups

At least the last two software backups should be archived. For reasons of safety, these should be stored at a different location from the system (according to the recommendations of the BSI (German authority responsible for security in information technology), for example in a fire compartment separate from the system).

A suitable backup strategy must be defined depending on the frequency at which changes are made.

The storage life of the data medium should be defined (for example based on the manufacturer's information or on publications of the relevant national authorities for information technology) and before this expires, the backup should be migrated, for example by copying it to a new data medium.

2.9.2 Process Data

The data saved in the system, such as trends, measured values or alarms should be backed up on external data media at periodic intervals. This measure can minimize data loss if problems occur.

Labeling data backups

According to the GAMP ® 4 Guide for Validation of Automated Systems, data backups should be documented either on the label of the backup itself or in a separate report containing the following information:

- System designations
- Software / data designation
- Version and/or software/firmware build number, if available
- Date of creation
- Date of first usage
- Current number
- Date of the data backup
- Reason for the data backup
- Identity of the operator

Retention of data backups

The same guidelines apply as in the section with the same name in Chapter 2.9.1 "Application Software".

Since process data, in contrast to software, is not normally stored in "overlapping" versions, suitable measures must be taken to ensure data integrity.

2.10 Retrieving Archived Data

Archived data must be retrievable at all times. Following system updates, care must be taken that the data transferred to archive prior to the update remains compatible.

2.11 Use of Third-Party Components

When using predefined third-party components (hardware and software), a supplier audit should always be performed and the supplier's quality management system verified. The compatibility of the hardware components must be confirmed.

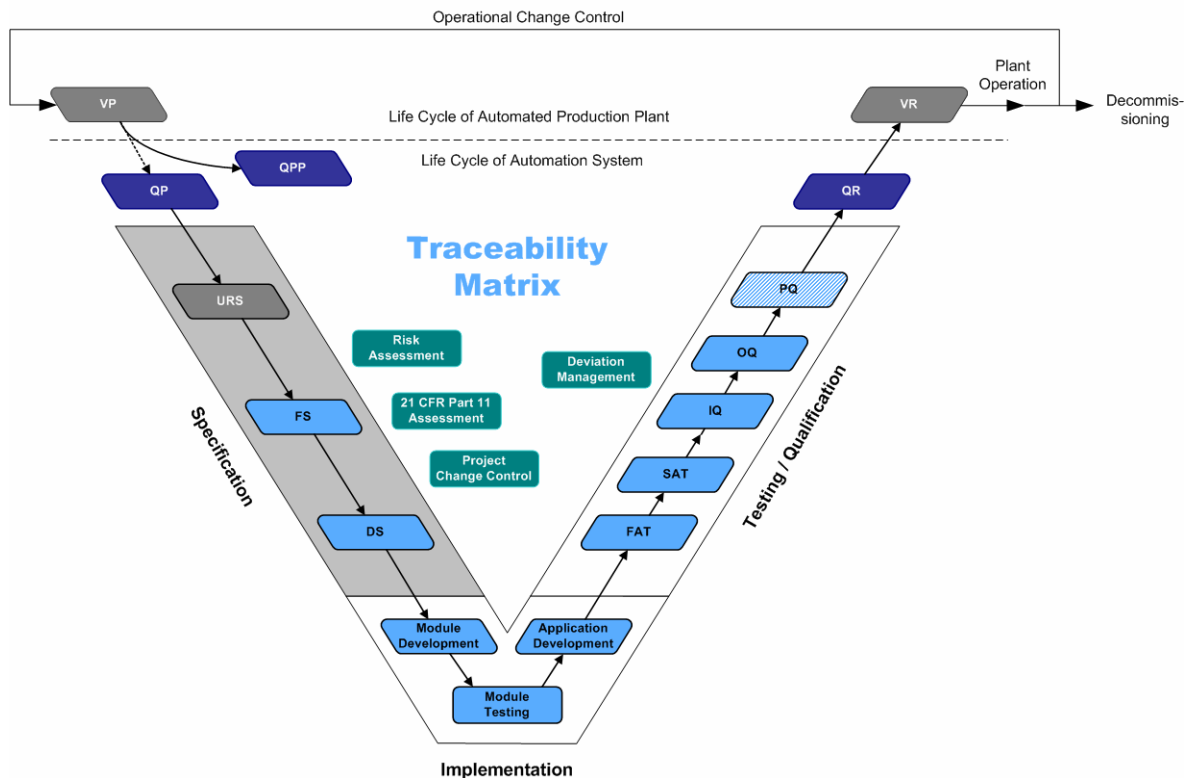
Even when using standard hardware and software components of other manufacturers, compatibility must be confirmed.

Note

The NAMUR Recommendation 72 contains a considerable amount of information on auditing a product supplier. Approaches to auditing a service provider or solution provider can also be found, for example, in the GAMP ® 4 Guide, Annex M2.

3 System specification

This chapter focuses on the selection criteria for the hardware and software. The activities for the selection of the products, product variants and system constellations are performed in the specification phase of an automation system. This is demonstrated in the following life-cycle model by the marking in the left-hand area.



Apart from the technical aspects, there are additional selection criteria for hardware and software in a GMP environment:

- Support of configuration management for hardware and software (version control, change control)
- Modularity and
- Scalability to minimize the validation effort.

The Totally Integrated Automation (TIA) concept provides the following:

- A common engineering environment
Software packages tested for compatibility (version compatibility of Windows operating systems, SIMATIC STEP 7, WinCC, WinCC flexible, utilities such as virus scanners, etc.)
- Standard software modules
corresponding to software category 3
- A common system-wide data management and database
Variable alignment between the controller and HMI with integrated operation by the system

- Service and support over all phases of the plant life cycle to maintain the validated status

3.1 Dimensioning of the hardware

When dimensioning hardware, both the work memory size and the software limits such as alarm resources and runtime must be taken into account.

A reserve of at least 25% should be allowed for the working memory size after completion of the commissioning.

Reserves of at least 10% for the racks, for the digital and analog input/output modules and, if required, for further signal modules must be calculated for the hardware expansion.

The grouping of signals that belong together technologically leads to a clearer hardware allocation. Reserves should be planned based on practical considerations.

3.2 Selection criteria for hardware

SIMATIC S7 programmable controllers (PLC) with a graduated level of performance are available for automation in process and manufacturing branches. The choice of PLC depends on the requirements of the automation task and the environmental conditions.

SIMATIC S7-300 / S7-400 programmable controllers are suitable for connection to networks based on MPI, PROFINET DP, PROFINET or Industrial Ethernet. For this reason, only these two series are covered in the manual.

Automation solutions in the lower and mid performance range are implemented with the PLC SIMATIC S7-300 that allows modular expansion. The CPUs graduated according to performance range from the standard CPU to the high-performance CPU with an efficient processing speed. The required hardware can be selected from a wide range of digital, analog, function and communication modules. The compact design is an advantage particularly where little space is available. The S7-300 range also includes the C7 compact devices (for example the C7-636 = CPU315-2DP + OP270 / TP270) and the ET200 with IM-CPU assigned parameters as an S7-300 for the smallest tasks or as an I- slave with backup functionality. The SIPLUS S7-300 was developed specially for use under more extreme environmental conditions such as low ambient temperatures (-25°C), aggressive vapors (chlorine / sulfur).

When the automation solutions are more exacting and require greater computing power in the mid to high performance range, the SIMATIC S7-400 that allows modular expansion and provides extremely high performance and very short command times is the version of choice. In addition to the performance-graduated CPUs, digital and analog modules as well as function and communication modules for special technological tasks are also available. For programming, high-level languages such as SCL and graphic engineering tools can be used.

Further criteria influencing the selection of the S7-300 or S7-400 automation system include functionality, such as:

- Alarm capability (see section 4.9 "Time stamping")
- Inserting / removing modules during operation (PROFINET IO with S7-300/400 and PROFIBUS DP only with S7-400)
- Redundant I/O modules for duplication of critical signals, only possible with an H CPU.
- CiR (Configuration in Run), expansion of the hardware configuration during runtime with an S7-400
- Redundant power supply (only with S7-400)
- Isochronous I/O buses
ensure acquisition, processing and data output at the same point in time to allow controllers (for example motors) to function precisely (S7-300 with CPU319-3 PN/DP, CPU317T, S7-400 without CPU41x-4H, PC-based Automation, WinAC Slot 412 / 416 and WinAC RTX. Detailed information is available in the SIMATIC Isochrone Mode manual.)
- Redundant CPUs
- I/O for hazardous areas (gas and dust)
- Fail-safe modules (CPUs and I/O)
- Display and upkeep of I&M data (Identification and Maintenance), see section 4.5.2 "Replacing/changing the hardware/firmware"
- Time synchronization / setting the time

- If the automation task is expanded to include additional aspects such as data processing, communication and visualization, the use of a SIMATIC PC is recommended. Whether for direct installation on-site or for installation in a cabinet, the SIMATIC PCs meet the high constructional demands for use in industry. SIMATIC PCs are the ideal platform for PC-based Automation in conjunction with PC technology. One particularly rugged platform for visualization on the plant floor is the Multipanel SIMATIC MP370 in conjunction with SIMATIC Embedded Control (an expansion of SIMATIC PC-based Control).

Note

You will find technical details on the SIMATIC S7-300 / S7-400 programmable controller systems, the range of I/O products, and the SIMATIC PCs in the current SIMATIC catalog ST 70.

3.3 Required/optional software packages

For the SIMATIC S7-300 / S7-400 programmable controller systems the engineering software is the STEP 7 basic software; for PC-based controllers, the SIMATIC WinAC software is used.

3.3.1 Basic engineering software

The STEP 7 basic software is used for the entire automation solution starting with the hardware configuration, followed by the communication and network connections and ultimately creation of the user program. The user program is created in one of the programming languages LAD (Ladder Diagram), FBD (Function Block Diagram) or STL (Statement List). The integrated syntax check validates the program code immediately as it is entered during programming. Standard functions in the form of standard function blocks / functions (SFB / SFC) ship with the CPU and simplify and speed up program creation. In STL it is possible to enter the program or parts of it as STL source files and then compile them into blocks. A source file can contain the code for several blocks that can then be compiled in compiler run. An STL source file is necessary to assign KNOW-HOW protection.

During testing and commissioning, the STEP 7 basic software provides support with user-friendly monitoring, control, and diagnostic functions.

The STEP 7 basic software is supplied in the user interface languages German, English, French, Italian and Spanish.

3.3.2 Additional engineering software

Various options are available to expand the STEP 7 basic software. Each of the following options requires a license.

- **S7-GRAPH**
For configuring and programming sequential processes using sequencers. If the ProAgent option (see below) is used for visualization, manual input does not generate messages that can be recorded in an audit trail. The sequencer picture should only be used to eliminate problems in S7-GRAPH sequencers.
- **S7-SCL**
For programming in a high-level language similar to PASCAL
- **S7-PLCSIM**
For the functional testing of the compiled user blocks regardless of the available target hardware on the engineering system

STEP 7 Professional is a combination of the individual packages STEP 7, S7-GRAPH, S7-SCL and S7-PLCSIM.

- **S7-HiGraph**
For the graphic description of asynchronous processes and status graphs

- **CFC**
Extensive library of ready-made blocks
The program is created by drawing a technological chart.
CFC (Continuous Function Chart) is a graphic editor that is used in conjunction with STEP 7 basic software. It is used to create an overall software structure for a CPU from the ready-made blocks. For this purpose, blocks are positioned on function charts, parameterized and interconnected. Interconnecting means that the connections for communication between blocks are established so that values from an output can be transferred to one or more inputs.
How it works in principle
A block library is part of the CFC package. Users can then add their own specific blocks. The CFC Editor works with graphic tools. A block is selected from the block pool and inserted in the chart that serves as a sort of "drawing board". Technology functions need only be parameterized by linking function blocks (AND, OR, PID controllers). Time-consuming programming is no longer necessary. Details such as algorithms or the allocation of machine resources recede into the background and the technological aspects of configuration come to the fore.
When the configuration data is transferred to the target system, the charts are first compiled and then downloaded. The CFC option guarantees the consistency of the configuration data when changes are downloaded. If there is already data on the automation system, it is possible to download only the changes (with a CPU 400). Once again, the consistency of the project data is guaranteed. By comparison, the STEP 7 basic software transfers changes to the target system block-oriented and these become effective immediately in runtime. This usually results in inconsistencies between an FB and its instances that can lead to the CPU changing to STOP due to the download in runtime.
- **S7-PDIAG**
For configuring process diagnostics and increasing the availability of machines and production facilities
- **SIMATIC ProAgent**
The SIMATIC ProAgent optional package makes the display and operator control pictures available on pages of SIMATIC HMI (WinCC Flex and WinCC) based on a standardized user interface (uniform for S7-PDIAG and S7-GRAPH).

Note

SIMATIC ProAgent cannot be used with redundant WinCC servers.

- **DOCPRO**
DOCPRO is a tool for creating and managing plant documentation. DOCPRO enables the structuring of project data, the editing in form of circuit manuals and the printout in a uniform print layout.
- **Distributed Safety**
For creating safety-oriented automation applications with SIMATIC S7 in F-LAD or F-FBD with CPU31xF and CPU416F

-
- F Systems (F/FH)
The S7 F Systems engineering tool, which is integrated in the SIMATIC Manager, can be used to configure an S7 F system (F/FH). This tool can be used to assign parameters for the CPU and F signal modules and to create applications in CFC. Predefined, TÜV-approved blocks are available for this purpose. The failsafe blocks relieve the user of having to perform the diverse programming tasks for the detection of errors and reaction to errors. (Can be used with CPU41x-4H)
 - SIMATIC Safety Matrix
The Safety Matrix is based on the proven principle of a causes & effects matrix with which precisely defined reactions (effects) can be assigned to events (causes) occurring during a process. This is part of a plant's risk analysis. The specification of the safety program also corresponds to the input parameters for the Safety Matrix. Based on these parameters, Safety Matrix automatically generates complex, fail-safe CFC programs.
Compared to conventional programming, the safety logic can thus be configured significantly faster and with a greatly reduced overhead. Special programming knowledge is not required, and the programmers can concentrate fully on the safety requirements of their plant. If necessary, several matrices can be linked together.
 - SIMATIC Safety Matrix Tool
For configuring safety functions as a CFC program
 - SIMATIC Safety Matrix Editor
For the creation and testing of the Safety Matrix logic in an external computer, independent of the engineering system; (can be used as an option in addition to the SIMATIC Safety Matrix Tool)
 - SIMATIC Safety Matrix Viewer
For operator control and monitoring of Safety Matrix via WinCC
 - Parameter assignment software for FM / CP modules (license required in some cases)
 - Runtime software
 - Standard PID Control
The Standard PID Control software package allows the integration of continuous PID controllers, pulse controllers, and step-action controllers into the user program. CFC **cannot** be used for the interconnections.
 - Modular PID Control
The Modular PID Control software package is the preferred tool for mid-range and high-end control applications and process engineering. Modular PID Control is always suitable when minimum memory usage and the fastest execution times along with optimized adaptation to the control task are required. CFC can be used for the interconnections.
 - PID Self-Tuner
The "PID Self-Tuner" software package turns existing PID controllers into self-setting PI or PID controllers. PID Self-Tuner can be flexibly combined with PID Control (integrated in STEP 7), Standard PID Control, Modular PID Control, FM 355, FM 455, and any PID algorithm
 - Software Redundancy
The "Software Redundancy" software package allows fault-tolerant controllers to be set up cost-effectively with standard hardware components of the S7-300 and S7-400. **ALARM_x messages cannot be used.**

- **SIMATIC iMap**
The SIMATIC iMap application (license required) is used to configure communication for individual components (Component based Automation CBA) in distributed automation solutions. Each bus node is displayed graphically along with its communication data, such as its IP address. The application is based on the PROFINET standard and all Ethernet nodes require the PROFINET communications mechanisms. More detailed information can be found in the SIMATIC ST 70 catalog.

3.3.3 Additional software packages for support of GMP compliance

3.3.3.1 Access Control

SIMATIC logon

For user management in STEP 7, the licensed software SIMATIC Logon is required. The user logs on via the SIMATIC Logon Service dialog box with user ID and password. Only authorized users have access to the STEP 7 project (see section 4.6 "Access protection").

3.3.3.2 Versioning, Change control , Audit trail

Version trail

The licensed Version Trail software is used to version STEP 7 projects. Major versions and minor versions can be specified by the user for the versioning. The criteria that decide whether the STEP 7 project is to be versioned as a major version or a minor version must be specified in the configuration management as well as the configuration elements that are to be versioned. To better understand the versioning, the individual versions should be assigned a version name and an informative comment.

Note

Versioning using Version Trail relates to complete STEP 7 projects. If an HMI system such as SIMATIC WinCC or SIMATIC WinCC flexible is integrated in the STEP 7 project, the HMI system is also included in the versioning.

Online / offline comparison (STEP 7)

In the STEP 7 basic software, a block comparison function is integrated for LAD/FBD/STL. This block comparison can be used both to compare different projects as well as for an online / offline comparison between target system and engineering system (see also section 4.5.4 "Change control")

Note

SCL source files that are not integrated in block cannot be compared using STEP 7. To compare two SCL source files, operating system tools can, for example, be used.

Version Cross Checker

Different versions of user programs created with the optional software CFC (see also section 4.5 "Configuration management") in the form of a technological chart can be checked and compared with the licensed application Version Cross Checker (VXC). The result is displayed as a combination of tree structure and table and can be stored in a text file.

Change log for online actions on the CPU

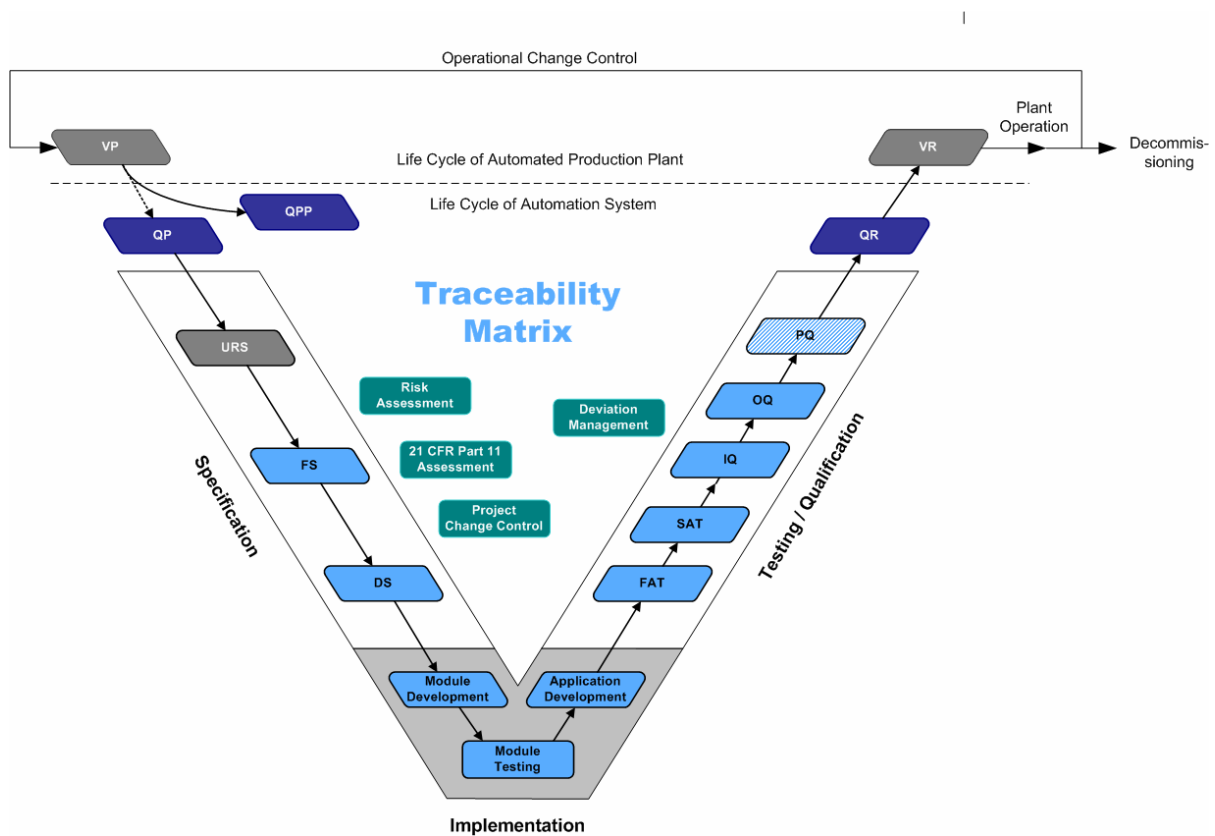
In conjunction with the SIMATIC Logon software, a change log can be activated in the STEP 7 basic software V5.4. Online changes with action, time stamp and user ID of the logged-on user are recorded in this change log.

4 Guidelines for implementing SIMATIC STEP 7 in a GMP environment

4.1 Introduction

This chapter explains the configuration of automation systems with STEP 7 in a GMP environment based on examples. The configuration of HMI systems in a GMP environment is not described in this chapter. Further information can be found in the SIMATIC WinCC and in the SIMATIC WinCC flexible GMP Engineering manuals.

The following graphic displays the life-cycle model. The focus of this chapter, the implementation, is indicated by the highlighting in the lower part of the graphic.



4.2 Software categorization of STEP 7

According to the GAMP ® 4 Guide for Validation of Automated Systems, the software components of a system can be assigned to five software categories. Below you will find examples illustrating how this categorization relates to STEP 7.

Category 1: Operating systems

Category 2: Firmware

Category 3: Standard software

Category 4: Configurable software packages

Category 5: User-specific software

<p>Category 1</p> <p>Operating system such as :</p> <ul style="list-style-type: none"> - WINDOWS 2000 Professional - WINDOWS XP Professional - WINDOWS Server 2003
<p>Category 2</p> <p>Firmware :</p> <ul style="list-style-type: none"> - Firmware in CPU - Firmware in Communication processors
<p>Category 3</p> <p>STEP 7 Standard software / Standard libraries</p> <ul style="list-style-type: none"> - SIMATIC Manager, KOP/FUP/AWL-Editor, etc. - Standard functions (SFB, SFC)
<p>Category 4</p> <ul style="list-style-type: none"> - Parameterizing of standard functions (SFB) - Instances of user-defined moduls - CFC plans
<p>Category 5</p> <ul style="list-style-type: none"> - Free programmed user defined software - Project specific blocks, functions, etc.

4.3 Software installation

The STEP 7 basic software is preinstalled on a programming device such as the Field PG. To configure the automation system on a standard PC, the STEP 7 basic software is installed on the PC. Hardware requirements and approved operating systems are documented in the readme file in **Start > SIMATIC > Product Notes** on the product CD.

The connection to the automation system is performed either via the MPI interface, which is already integrated in the programming devices, via PROFIBUS DP or via Industrial Ethernet. Suitable plug-in cards are available for installation in standard PCs.

For detailed information on hardware, refer to the current SIMATIC ST70 catalog.

4.4 Basic engineering principles

The automation task is described in detail in the User Requirements Specification (URS), Functional Specification (FS) and in the Design Specification (DS). The hardware is planned for the automation system and the user-specific software developed on the basis of this documentation.

The causes & effects matrix during risk analysis is supported ideally by the SIMATIC Safety Matrix option.

4.4.1 Software creation

The central user interface of the STEP 7 basic software is the SIMATIC Manager. The STEP 7 project is created and managed in the SIMATIC Manager. All objects of the project are displayed in a clear tree structure. During configuration and creation of the program, various editors, for example HW Config, Symbol Editor, LAD/STL/FBD Editor etc., are opened.

Using the STEP 7 basic software, a modular program is created. Different block types provide support for the structuring of the software.

A STEP 7 project or multiproject includes the hardware configuration of one or more automation systems, the symbol table, the application software for the individual automation systems, the configuration of the network connections and the documentation. If HMI systems are integrated in the project, their project data is also included in the STEP 7 project.

The folder for all projects and libraries of an automation solution is known as the multiproject and contains one or more STEP 7 projects and optionally also libraries. The projects within the multiproject can contain objects with cross-project relationships (e.g. cross-project S7 connections).

Benefits of Multiproject

- If projects are part of a multiproject, they can be created with a smaller size and clearer structure.
- Using multiprojects, you can, for example, create one project for each operator and divide the stations among the operators for distributed execution.
- Cross-project functions allow you to handle a multiproject almost like a single project.

For more detailed information, refer to the **STEP 7 Help > Working with projects in a multiproject** and the manual **Configuring Hardware and Communication Connections STEP 7 > Chapter 16**.

4.4.1.1 Procedure for programming

Before beginning the programming, it is recommended that the automation task be split up into smaller function areas. Individual units of equipment such as valves, motors, etc., are compiled and described according to their function. The operating philosophy of the production facility is specified in the manual/automatic/local operating modes. A message concept and a security concept are worked out.

Proposals for procedures are described in the **STEP 7 Help > Concept of the automation solution**.

Note

Once all the functionalities have been identified, the next step is to check the extent to which standard components (SFB /SFC / FB / FC) can be used. The use of standard components greatly reduces validation effort since no software is being created and the only activities involved are configuring calls and setting parameters.

The standard components are listed and documented in the STEP 7 Help in **Calling Reference Helps > Language Descriptions, Help on Blocks, System Attributes**.

To create the software, the STEP 7 basic software provides the programming languages Ladder Diagram (LAD), Function Block Diagram (FBD) and Statement List (STL) complying with the standard DIN EN 61131-3 / IEC 1131-3. Users also have the option of using other programming languages (see also section 3.3.2 "Additional engineering software").

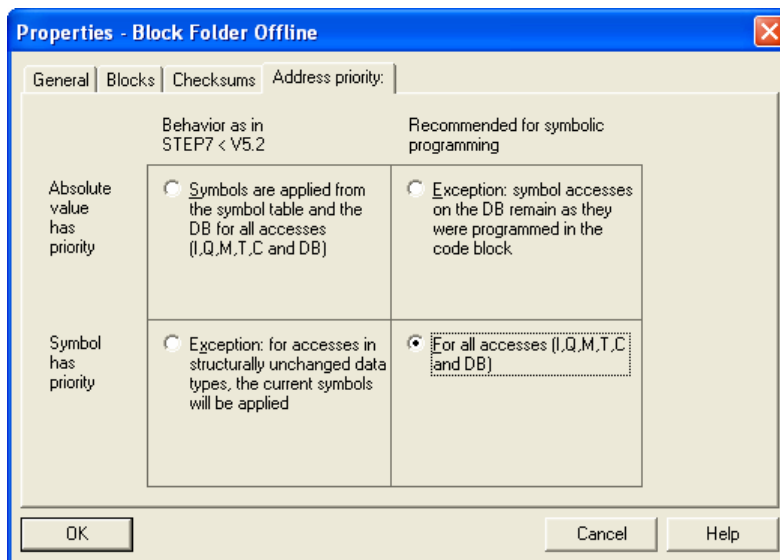
The software is created in graphical form not only in LAD, but also in FBD. LAD is read as a circuit diagram, FBD uses the graphic symbols of Boolean algebra. The syntax is checked as entries are made. It is possible to switch over directly from LAD to FBD and vice versa. This type of representation provides a fast overview of the programmed functions and makes these programming languages the language of choice when creating programs.

The Statement List programming language (STL) lists the program code line by line. LAD and FBD can always be represented in STL. Since, however, not all instructions, for example mathematical calculations, can be implemented with LAD or FBD, the STL programming language is used when these are required.

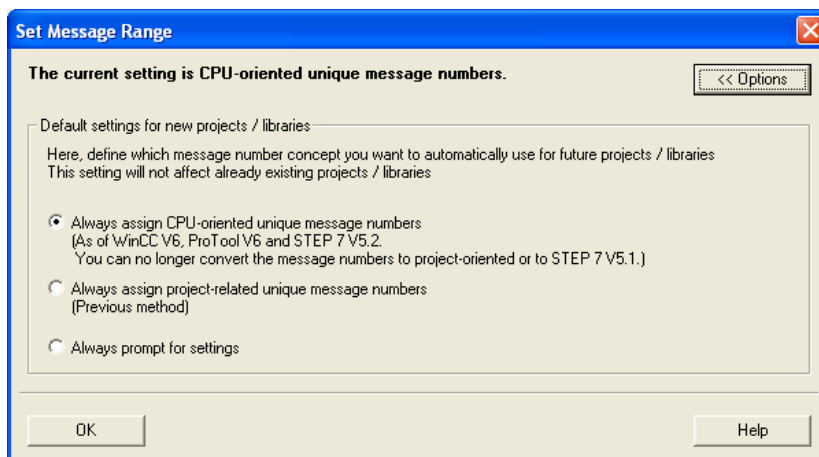
The program is created in the LAD/STL/FBD Editor in the form of blocks. Functions that are required several times in the program are programmed separately in user function blocks (FBs) or functions (FCs), tested as user-specific standard components and called where required. Specifying a variable interface for FBs allows the function to be called repeatedly with different parameter settings (for example a motor block). An instance DB is assigned to an FB.

Properties of a block container

General settings for programming are made in the object properties of the block container. In **Object Properties > Address priority**, the recommendation is to set the symbolic representation for I, Q, M, T, C, and DB. The assignment of the absolute parameters to the symbolic parameters is made in the symbol table (see also section 4.4.2 "Integrated HMI system"). Symbolic parameters convey much more meaning than absolute parameters.

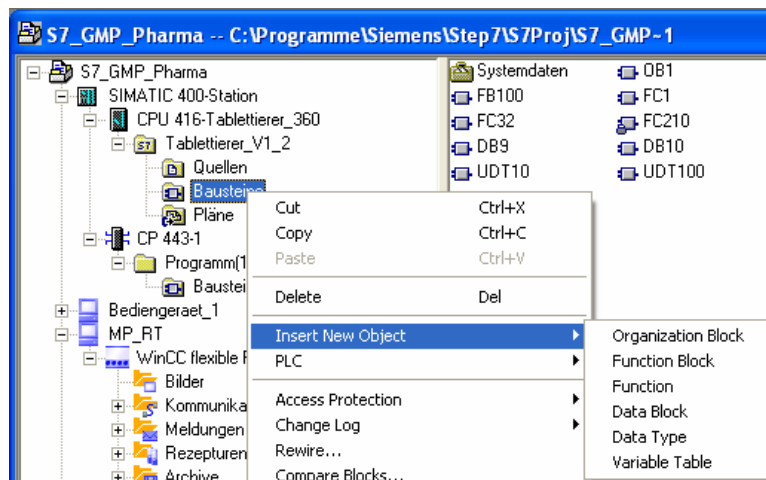


The way in which message numbers are assigned is set in the block container in **Special Object Properties**.

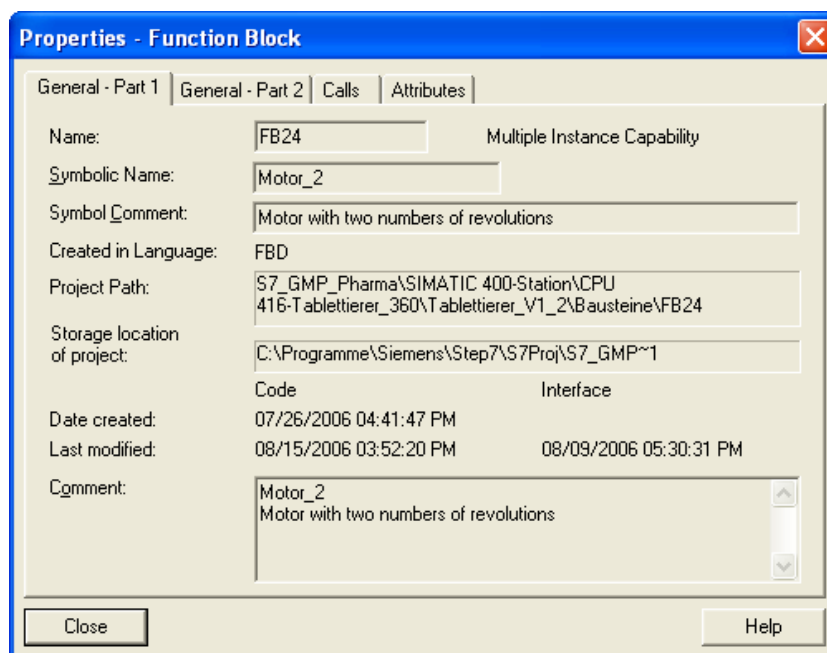


Block properties

When a new block is created in the SIMATIC Manager, the block properties are configured first. When a block is created in the LAD/STL/FBD Editor, the block properties should be edited later in the SIMATIC Manager.



The dialog for configuring the properties opens automatically after the block type is selected.



The block number, name, creation language etc. are defined in the properties. The time stamps for **Date created:** and **Last modified:** are assigned automatically by the STEP 7 basic software according to the local computer time. The Comment field shows the comment on the block that was entered before the first network in the block.

The **General - Part 2** tab shows the additional information block header, version number, author, etc.

Properties - Function Block

General - Part 1 | **General - Part 2** | Calls | Attributes

Name (Header): Motor_E Version (Header): 0.1
 Family: Motor Author: LP

Lengths

Local Data:	0 bytes
MC7:	2 bytes
Load Memory Requirement:	90 bytes
Work Memory Requirement:	38 bytes

☐ DB is write-protected in the PLC ☐ Standard block
☐ Know-how protection ☐ Unlinked
☐ Non Retain ☐ Block read-only

OK Cancel Help

To keep track of different block versions, a version number can be entered manually here.

The **Calls** tab lists which blocks this block itself calls. Special block attributes are assigned in the **Attributes** tab, for example, the S7_m_c attribute required for mapping the interface to the SIMATIC WinCC operator control and monitoring system for integrated operation. (See also section 4.4.2 "Integrated HMI system")

Additional block properties such as KNOW HOW protection are only displayed here.

Once the block properties are completed, the block is created as an object in the Blocks folder. Double-clicking on the block opens the LAD/STL/FBD Editor for program creation.

4.4.1.2 Rules and conventions

- The software should be well structured and created in as simple a form as possible. This makes it easier to understand the programmed functions.
- Functions that are used often are created once in an FB (can be assigned an instance DB) or a typical and tested. The block / typical is instantiated in the program.
- It should be possible to display the programmed networks in LAD / FBD / STL on one screen page so that the programmed function can be seen all at once.
- Outputs should be accessed only once in the user program.
- Adequate care should be taken when calculating the execution times in the PLC program. Example: When calculating the time for a cleaning procedure, the expired time should not be calculated simply by deducting the current absolute time from the starting time. This can lead to serious problems if the time is changed during the cleaning procedure, for example due to synchronization or due to a standard/daylight saving time change. This could lead to a shortened cleaning time. -> This means a deviation from the specification. -> This might mean the loss of an entire batch, etc.

-
- When creating the program, standard components already on the CPU should be used. (For example, for reading in analog values (FC105, FC106), for alarm processing (Alarm_S etc.)
 - Routines for cyclic processing, warm restarts, hot restarts and error handling and messages should be included in FBs created by the user. For processing, the same FB is called in the appropriate OBs.
 - The user-specific program should be created with diagnostic messages, such as runtime, limit position and limit value monitoring. The diagnostic messages should be displayed on the connected operator control and monitoring system.
 - Indirect addressing should be avoided as this has a significant adverse effect on the readability of the program. If indirect addressing cannot be avoided because of the function to be implemented, a plausibility check of the pointer calculation should be incorporated. For indirect addressing, SCL is the recommended programming language. The software must be commented on in detail.
 - The standard function blocks (for example ALARM_S/D, ALARM_8/8P) should be used for message processing. The standard function blocks for alarm processing are described in the **STEP 7 Help > Calling Reference Helps > Block Descriptions, Help on Blocks... > Help on SFBs / SFCs > section *Creating Block-related Messages***. The selection of the block is based on the automation system (S7-300 only ALARM_S and S7-400 ALARM_S/D and ALARM_8/8P) and the connected operator control and monitoring system (SIMATIC WinCC flexible only with ALARM_S/D).
 - Organization blocks called if an error occurs should be inserted in the program to allow diagnostics. At least one message should be generated in the OBs.
 - The following activities must be performed for a code review:
 - Check whether the previously defined "Device identifier name convention" has been used consistently in the software
 - Check whether the previously defined versioning (SOP configuration management) has been performed
 - Check whether a unique symbol name has been assigned for all operands
 - Check whether each operand has been used in the user program
 - Check whether all inputs/outputs are clearly contained in the symbol table
 - Check whether the software has been adequately commented
 - Check for blocks that are not called, networks without program (dead code)
 - Check of the program structure
 - Check whether all operands used in the software are set/reset/assigned only once
 - Auxiliary development code must be identifiable

4.4.1.3 Software interlocks/safety

It is the responsibility of the software engineer to ensure that the user program functions in a safe way under all conditions. Before creating the software, events must be considered that can lead to a dangerous reaction and the appropriate interlocks must be created.

Such events are, for example:

- Restart after line voltage failure
- Opening/closing of valves or similar components without an operator action and acoustic signal
- Starting of motors without an operator action and acoustic signal
- Faulty input from external operator panels may not lead to any incorrect reactions
- Reaction to an unexpected absence of an analog input signal
- Output values of the analog output signals when the CPU unexpectedly goes into STOP
- For legal reasons, emergency stop functions may not be implemented with a standard PLC. The fail-safe systems are available for this. The SIMATIC Safety Matrix is recommended. Safety can also be guaranteed without a PLC with an emergency stop.

Note

The selection of the right hardware is also important to guarantee the correct logic decisions in a production plant (DI = 0 means 'Sensor open' or 'Wire breakage'). Wire breakage monitoring can also be performed, for example, via signal modules with NAMUR technology.

4.4.2 Integrated HMI system

As part of the Totally Integrated Automation (TIA) concept, the WinCC flexible project or the WinCC project can be integrated in a STEP 7 project.

Advantages of the integration:

- Central overview in the tree topology of the SIMATIC Manager
- Central variable management in the symbol table of the SIMATIC Manager
- Central connection overview of all participating components via NetPro

SIMATIC WinCC flexible

Due to the order of installation (refer to the information on the installation CD), the WinCC flexible system software can be integrated directly in the SIMATIC Manager to configure a new project directly related to the STEP 7 project. The subsequent integration of an already existing project is also supported by the SIMATIC Manager.

After the integration of the WinCC flexible project, the symbol editor of the SIMATIC Manager is accessed to allow the variables to be linked into WinCC flexible.

- The integration of WinCC flexible is described in greater detail in the SIMATIC WinCC flexible GMP manual.

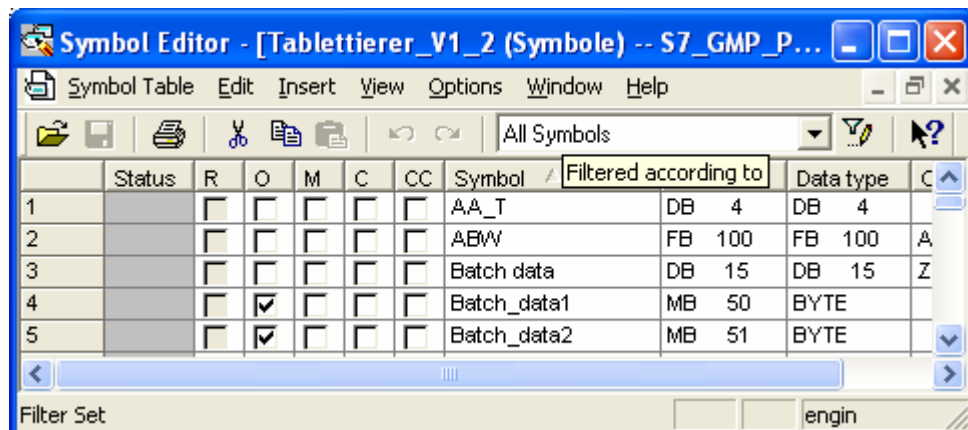
SIMATIC WinCC

To allow integration of SIMATIC WinCC, an OS object is created for every operator and monitoring station in the tree structure of the SIMATIC Manager. All variables (tags) necessary for operator control and monitoring in WinCC are assigned the OCM attribute S7_m_c.

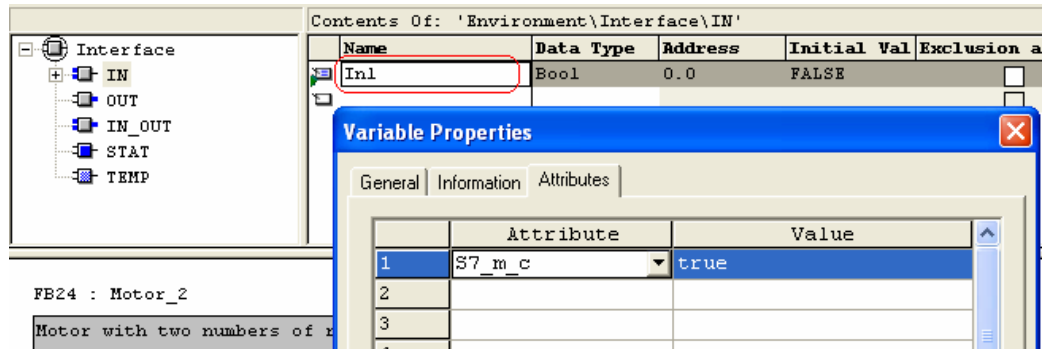
The OCM attribute can be configured for the following variables:

- Input / output and in-out parameters of function blocks registered for mapping with S7_m_c.
- Bit memory and I/O signals only with S7-400

The attribute is assigned to function blocks / data blocks in the object properties, to individual parameters in the function blocks, or to the symbols in the symbol table.



The graphic above shows the symbol table with the columns for setting attributes. The attribute for operator control and monitoring is enabled in column B.



The graphic above shows the assignment of attributes for a block parameter.

Naming convention

To allow the configuration data for WinCC to be saved and transferred, they are stored under a unique name assigned automatically by STEP 7. The names of the variables that can be controlled and monitored by an operator, CFC charts and the S7 programs form part of this name and are therefore subject to certain conventions:

- The names of the S7 programs in an S7 project must be unique.
- The names of the variables, S7 programs and CFC charts should not contain any blanks or special characters.

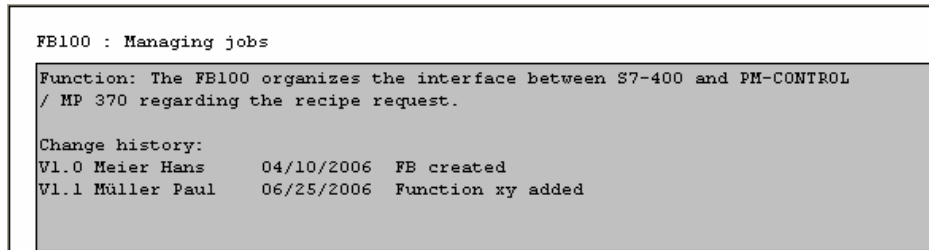
AS-OS Engineering

The data prepared for display in WinCC is transferred to the data management of WinCC using the AS-OS Engineering transfer system. The AS-OS Engineering application ships with the WinCC system software and must be installed separately. During installation, the S7 project of an automation system is assigned to an OS object. The variants whereby several automation systems are visualized on one OS or several operator stations are used to visualize one automation system can also be configured.

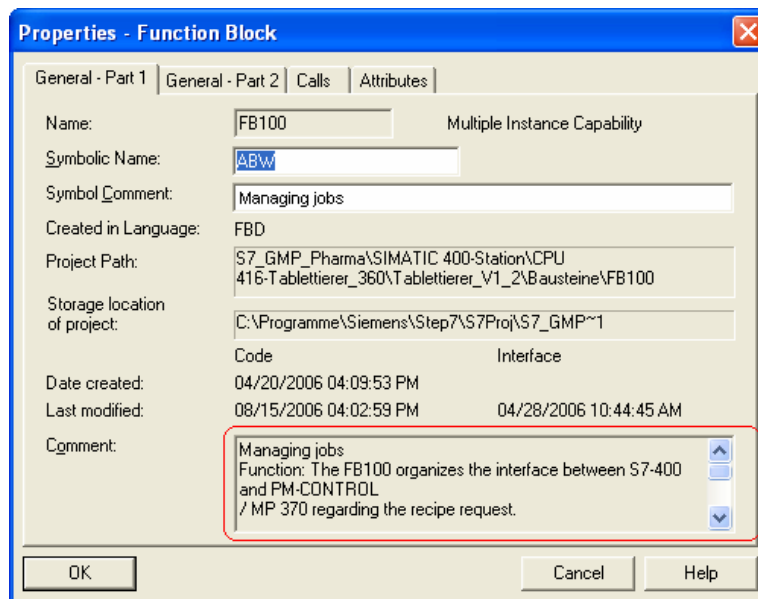
4.4.3 Software documentation

Detailed commenting is required for good readability of the user program. The following comments are available in the STEP 7 basic software:

Block title and block comment

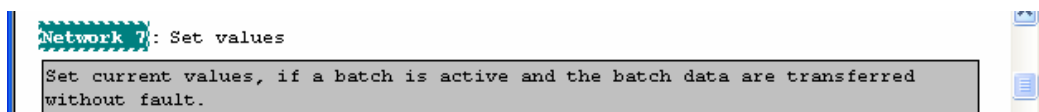


Normally the block comment describes the function of the block. The block comment is also suitable for manual upkeep of the change history. The block comment is also displayed in the *Properties* dialog for the block opened with the **Object Properties** context menu of the block in the SIMATIC Manager.



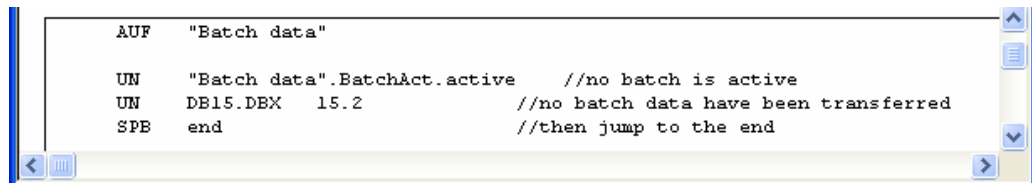
Network title and network comment

Each network has an informative title. The network comment box is available for a detailed description of the programmed function.



Line comment

The user program also includes comments for each line if STL or the optional language SCL was used for programming.



```

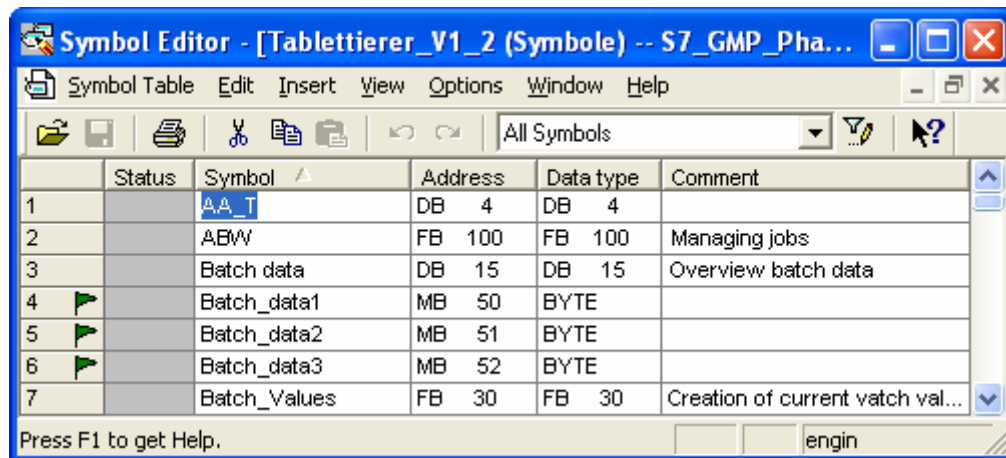
AUF  "Batch data"

UN   "Batch data".BatchAct.active    //no batch is active
UN   DB15.DBX 15.2                   //no batch data have been transferred
SPB  end                             //then jump to the end

```

Symbolic name

A symbolic name is entered in the symbol table for all addresses such as digital and analog inputs and outputs, counters, timers, bit memory depending on its use with bit, byte, word or double word and the programmed blocks (FB, FC, UDT, DB). This symbolic name should be selected so that, for example, the relationship to a unit or a function can be recognized. A maximum of 24 characters are permitted for a symbol name. The name is not case-sensitive. A detailed comment can also be entered.



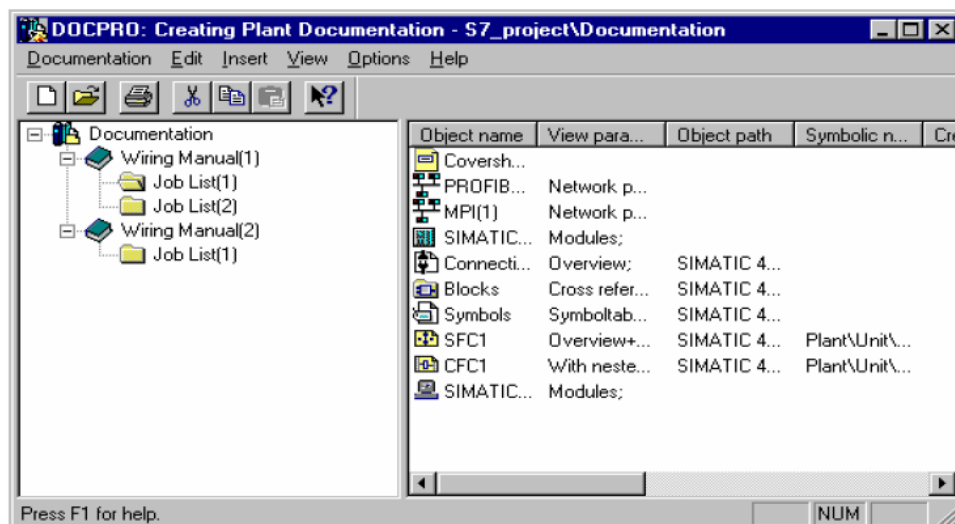
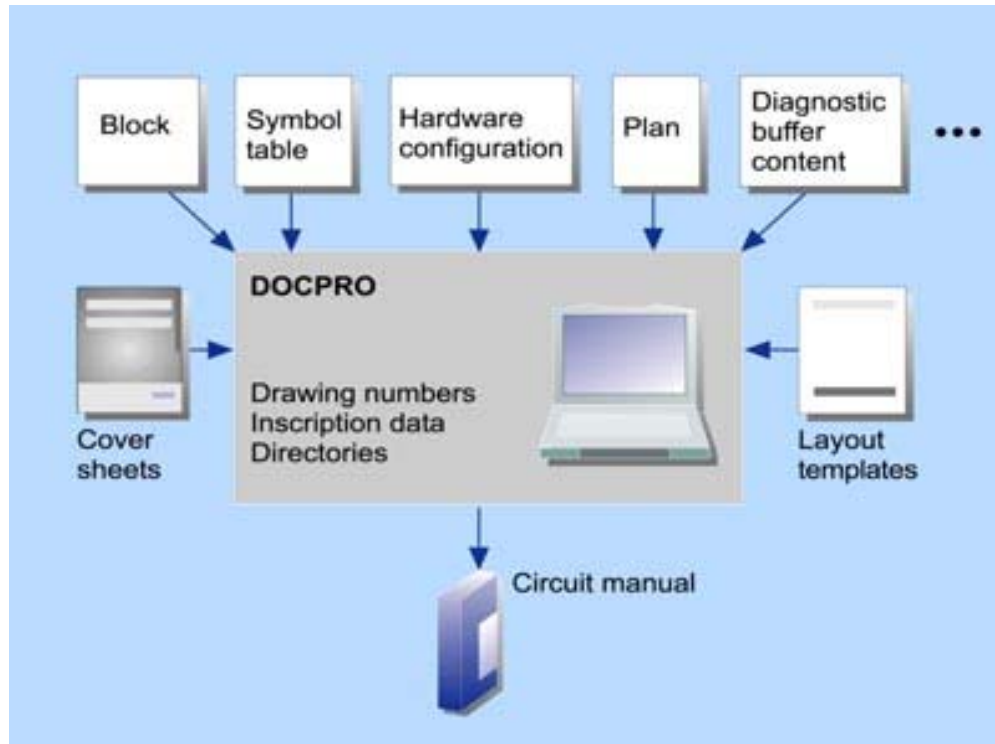
	Status	Symbol	Address	Data type	Comment
1		AA_T	DB 4	DB 4	
2		ABW	FB 100	FB 100	Managing jobs
3		Batch data	DB 15	DB 15	Overview batch data
4	▶	Batch_data1	MB 50	BYTE	
5	▶	Batch_data2	MB 51	BYTE	
6	▶	Batch_data3	MB 52	BYTE	
7		Batch_Values	FB 30	FB 30	Creation of current vatch val...

Press F1 to get Help. engin

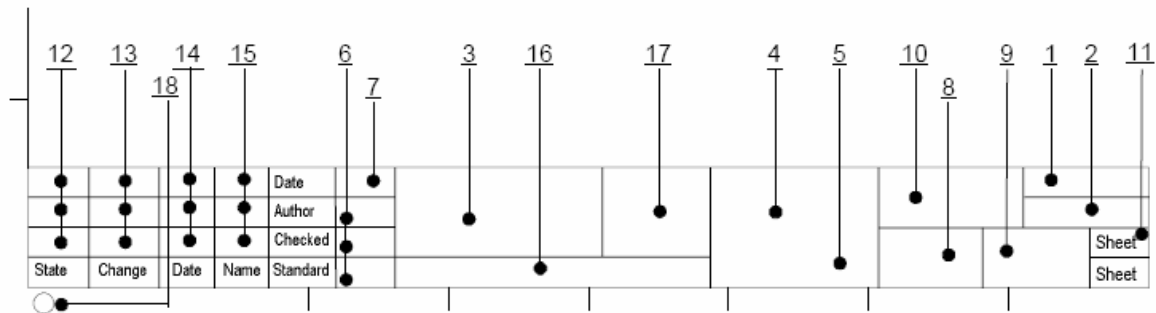
With STEP 7, it is possible to export, translate and re-import texts that exist in one language in a project and then display them in the language into which they were translated. These include titles, comments, and display texts. You will find more detailed information in the STEP 7 Help under **Setting Up and Editing the Project > Editing a Project > Managing Multilingual Texts**.

DOCPRO

The licensed DOCPRO option is a user-friendly tool for creating a consistent plant log including versioning.



As default, footers are available that comply with DIN 6771. The layout can be adapted widely to suit the user's requirements; the sketch shows the basic layout of a footer according to DIN 6771. The individual fields are not shown to size.



Meaning of the individual footer fields:

No.	Meaning	No.	Meaning
1	System identifier block	10	Customer's drawing number
2	Location identifier block	11	Sheet no.
3	Customer, Operator, System	12	Status: modification index
4	Designation	13	Comment on the modification
5	Document type	14	Date when modified
6	Editor, tested, standards	15	Modified by (name)
7	Date of creation	16	Number of original chart
8	Special Comment	17	Creating company
9	Document number	18	Company, Year

All the data created with a configuration tool can be inserted in DOCPRO documentation. The data is therefore available in a clearly structured form and can be used centrally for qualification. Documentation could, for example, contain the following data:

- Blocks (program code, created with LAD, FBD, STL etc.)
- Symbol tables with the symbolic names of absolute addresses
- Reference data such as cross-reference lists, assignment lists, program structure
- Hardware configuration tables with the arrangement of the modules in the automation system and information on the module parameters
- Variable tables with status formats along with status and control values
- Shared data tables
- Connection tables
- CFC Charts

Note

Entering an object into the documentation creates a print object. DOCPRO does not store the object in its own data management but requests it from the application responsible at the time it is printed. The printouts always contain the current data.

4.5 Configuration management

STEP 7 is the basic package for configuring and programming SIMATIC automation systems. The SIMATIC programming languages integrated in STEP 7 meet the requirements of the standard DIN EN 6.1131-3 / IEC 1131-3. Standards are part of the basic package in the form of libraries. These include, for example, function blocks for PID controllers, time stamping, interrupt processing, system functions, system function blocks, etc.

The configuration and program creation is user-specific, dependent on the requirements of the automation task. The software creation is difficult to trace without version and change management. Therefore, a professional configuration management should be used right from the start of the software creation.

The configuration management should be described in an SOP. Everyone involved in the project must be trained in the use of this SOP; this produces a common base for the configuration.

Note

The following section provides an example for the software versioning and for the change control. Changes made to a plant in operation should always be agreed with the plant operator.

4.5.1 Changing the system software

Updates, Service Packs and Hotfixes

Updates, Service Packs and Hotfixes are provided for the STEP 7 basic software for functional expansion or for fixing bugs.

- An update is an update within a STEP 7 version.
- A hardware catalog update contains files for the new hardware
- A service pack fixes bugs and includes several hotfixes.
- A hotfix is an interim bug fix. Hotfixes are available only in special situations.

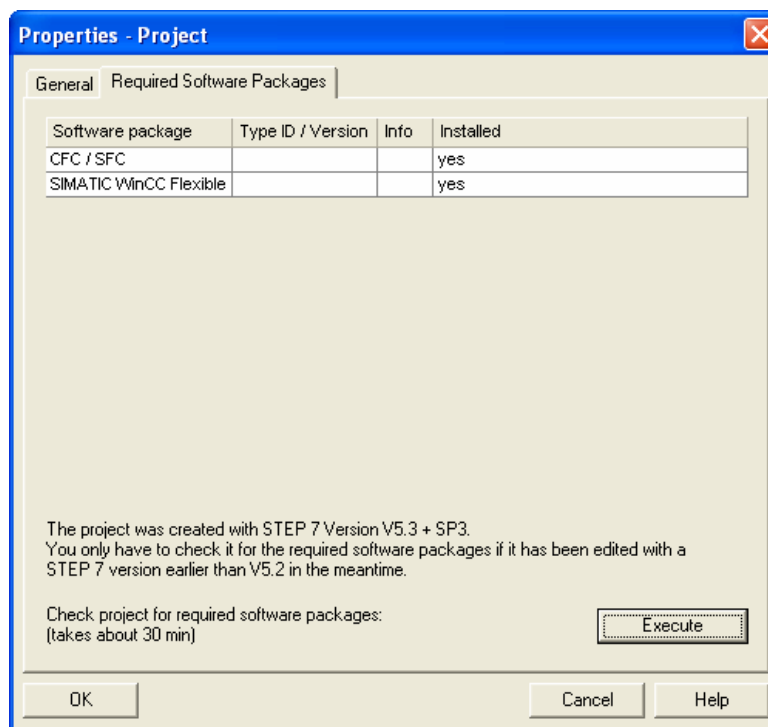
The validation work with respect to changes is specified within the framework of a risk evaluation.

Upgrades (migration)

An upgrade upgrades the STEP 7 basic software. In addition to the expansion of the range of functions, the upgrade can also mean a supplement to the integrated hardware catalog. Adaptations may be necessary in the **HW Config** editor or in the connection settings in the **NetPro** editor.

Appropriate procedures are documented in the STEP 7 help system under **Editing Projects with Different STEP 7 Versions**.

The precise software version with which the STEP 7 project was created and the optional packages used can be seen using the context menu **Object Properties > Required software packages** for the top node in the project structure.



The **Execute** button checks the current project or the current library for software packages that are required to process certain objects. This function is used when a project or a library is edited the first time with an older STEP 7 version.

Note

With updates and migrations, a validation check should be performed on the tools used. Standard tools can be used without any additional validation work.

Procedure when updating the system software

When the STEP 7 basic software is updated, certain measures are necessary to retain the validated status of the plant:

- Basis for a change is always the change request of the operator
- Description of the causes for the update of the system software
- Description of the new functions
- Data on the compatibility to the previous version
- Update of the technical documentation
- Installation according to manufacturer documentation (e.g. standard documentation, project migrator)
- It is recommended that a risk evaluation be carried out beforehand and the main testing points for the qualification specified
- Qualification

4.5.2 Replacing/changing the hardware/firmware

When upgrading the hardware/firmware, certain measures are required to retain the validated status of the plant:

- Basis for a change is always the change request of the operator
- Description of the causes for the upgrade of the hardware
- Description of the new, improved functions
- Data on the compatibility to the previous version
- Update of the technical documentation
- Installation according to manufacturer documentation (e.g. standard documentation)
- It is recommended that a risk evaluation be carried out beforehand and the main testing points for the qualification specified
- Qualification

Updating the firmware

To update the firmware for modules, follow the instructions in the STEP 7 Help under **Downloading and Uploading > Downloading from the PC/PG to the Programmable Controller > Updating Modules and Submodules** or under **Setting the Operating Response > Updating the Operating system on the PLC**.

Replacing modules

Before replacing modules, the module type must be checked. The Identification and Maintenance data (I&M) is helpful. I&M functions define information functions with information about devices can be called up, for example, vendor, version, ordering data etc. By using I&M functions, it is possible to evaluate information on the device in the phases configuration, commissioning, parameter assignment, diagnostics and repair. Identification data (I data) is information on the module, some of which is printed on the module housing. I data can be read during module diagnostics with STEP 7 ("General" tab and "Identification" tab of the module information). M data is plant-dependent information such as the HID (plant designation), LID (location identifier), installation date and comment that should be maintained in the object properties of the module in HW Config. M data can be written to the module using online access. If a new module type is available, the compatibility to the previous type must be checked. If it is compatible, the module can be replaced directly.

Note

When checking the compatibility of a CPU module, it must be taken into consideration whether faster runtimes have an effect on the user program.

If there is no compatibility, configuration measures are required. The STEP 7 project must be updated with the required hardware update. The appropriate adaptations are made in the **HW Config** editor.

The S7-400 automation system supports removing and inserting modules during operation. This is only possible if OB83 is programmed and executes both when a module is removed or inserted. IN conjunction with PROFINET IO, the S7-300 also supports removal and insertion of modules during operation (OB55, 56, 57). When replacing modules it should be make sure that a suitable module is inserted again. Inserting the wrong module is indicated by the EXTf LED lighting up on the CPU.

4.5.3 Versioning of the application software

General information about versioning

The software version provides information on the current version of the application software.

The following data are specified for the versioning of the application software:

- Name
- Date
- Version number
- Comment on the change
- Change history

Note

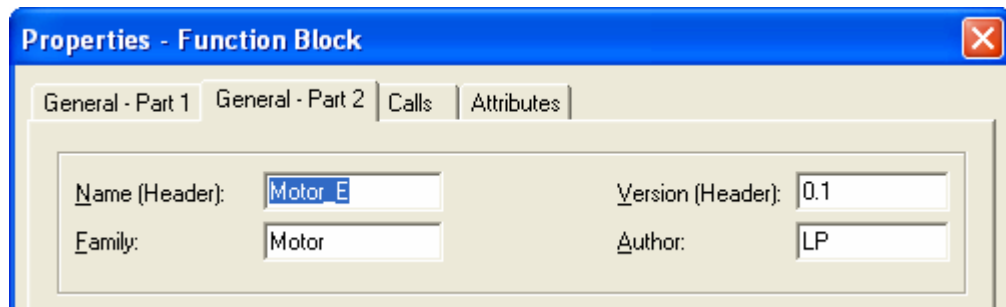
Versioning of the application software in STEP 7 is possible block-oriented. It can be maintained, for example, with the "Version" box in the block properties and / or with additional comments.

The procedure for the versioning is part of the configuration management and must be described in a SOP, which is binding for all persons participating in the project.

Below, you will see examples and options for versioning in STEP 7.

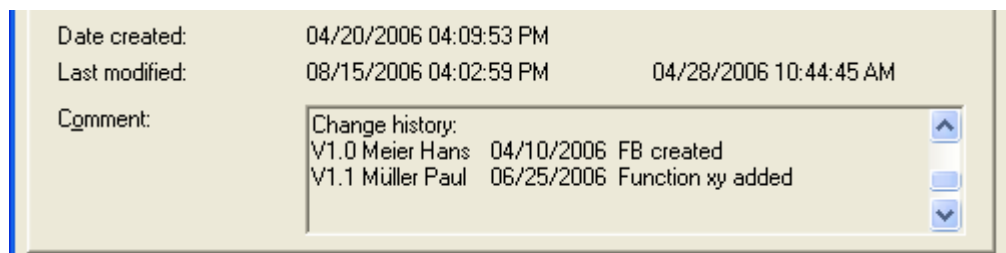
Versioning individual blocks

Object properties are created automatically in STEP 7 for every block (OB, FB, FC, DB, UDT). These can be viewed and edited manually with the context menu **Object Properties** of the block selected in the **Blocks** folder. The **General – Part 2** tab contains boxes for the version number and the author.



The version number is assigned manually. The criteria for the assignment of a major or minor version number must be set down in a SOP along with information on the configuration elements to be versioned.

The change history is entered in the block comment. The block comment and with it the change history can also be viewed in the object properties of the block in the **General - Part 1** tab.



The **Date created** is fixed automatically by STEP 7 when the block is created. The information in **Last modified**: is also recorded automatically by the system separately for code and interface changes.

Versioning the STEP 7 project

The STEP 7 basic software generates checksums for system data and user data that represent a certain version of the STEP 7 project (see also section 4.6.3 "Protective Measures in the software").

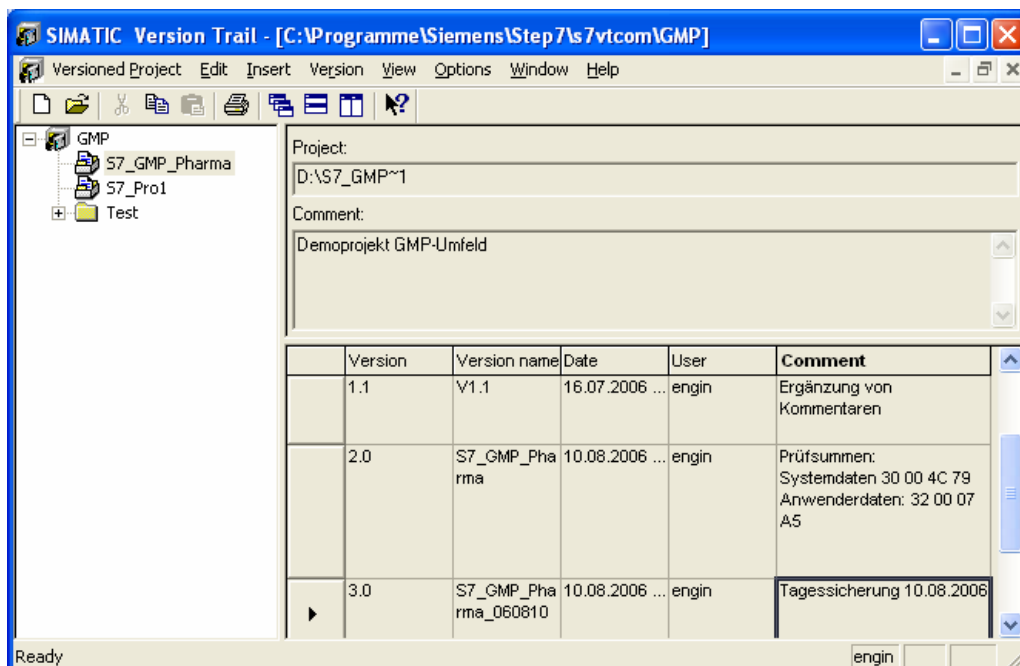
To manage the S7 projects (application software) from a GMP perspective through the life cycle of an automation system, the licensed software *Version Trail* is used (see also section 3.3.3.2 "Versioning, Change control").

Note

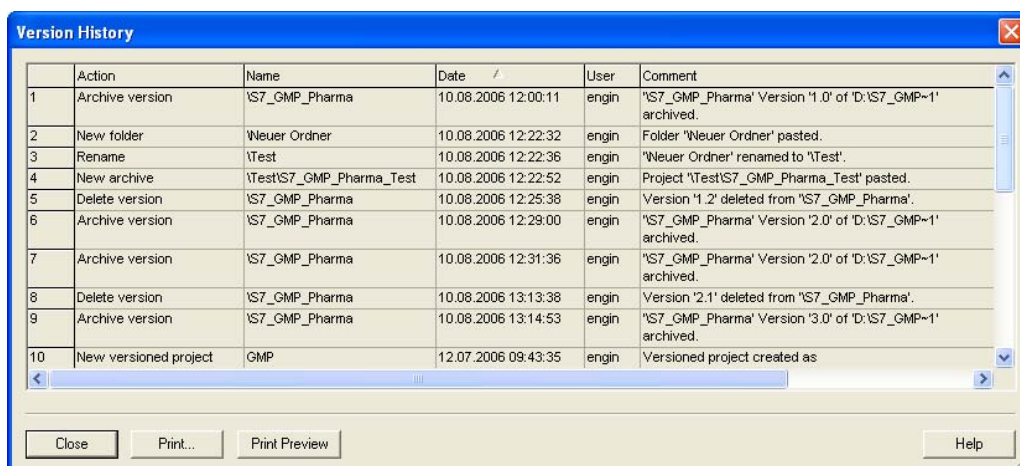
To use Version Trail, the licensed software SIMATIC Logon is required. A user must be logged on over the SIMATIC Logon Service or over the SIMATIC Manager under whose name all relevant actions are entered in the change history.

With *Version Trail*, it is possible to version several STEP 7 projects / libraries / multiprojects. Each STEP 7 project is archived under a major and a minor version number. The criteria for the assignment of a major or minor version number must be set down in a SOP along with information on the configuration elements to be versioned. Version Trail ensures a continuous incrementation of the version according to validation perspectives. A completed version can no longer be changed. Projects can also be read back (retrieved) from an archive of project data created with Version Trail (version project).

Version Trail is started with **Start > Programs > SIMATIC > STEP 7 > Version Trail**. Only closed projects can be archived.



Every archiving action in Version Trail is recorded with the action, project name, version number, time stamp, logged on user ID and comment. The history can be viewed using the **Options > Version History** menu.



The Version Trail software can be accessed directly in the SIMATIC Manager to archive / retrieve a STEP 7 project. The menu **File > Version > Archive (Retrieve)** is opened for this.

Note

If an HMI system such as WinCC or WinCC flexible is integrated in the STEP 7 project, the HMI project is also included in the versioning.

The help system of Version Trail has detailed information on creating and managing project versions.

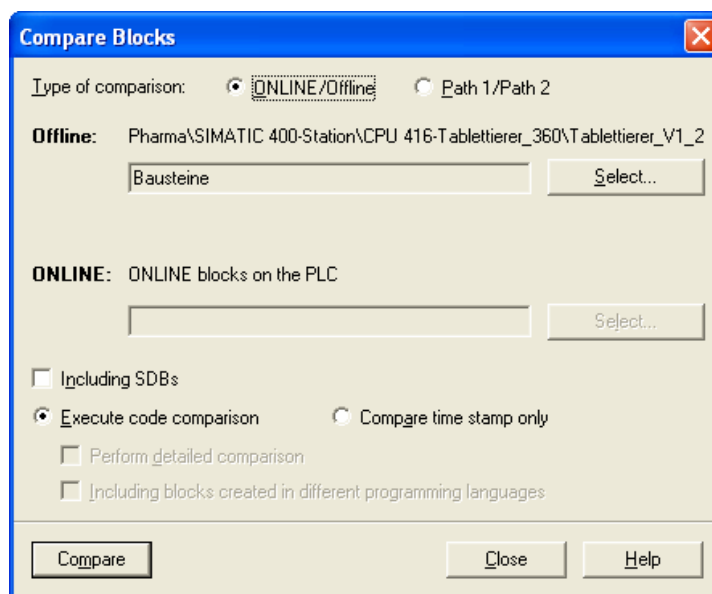
4.5.4 Change control

The change control provides information on changes made to the application software (**Who** changed **What** **When**).

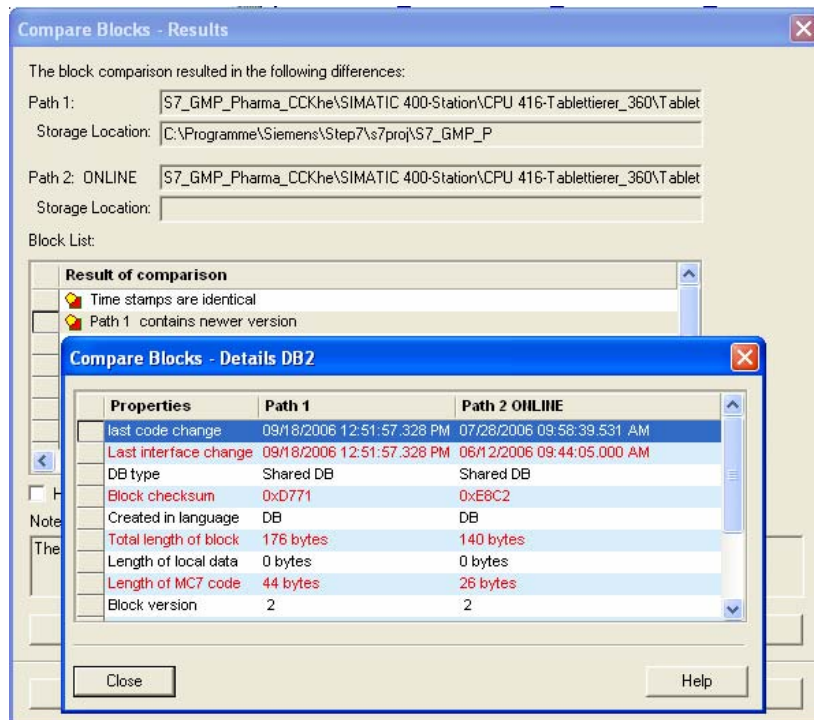
Online / offline comparison based on STEP 7

With the online / offline comparison, STEP 7 provides a tool that represents differences between two different projects/versions (offline/offline) or between the CPU and project version (online/offline). The programming languages LAD/STL/FBD are supported.

The online / offline comparison is opened in the SIMATIC Manager in the **Extras > Compare Blocks** menu.



Either block code or the time stamp can be compared. If the **SDB** check box is enabled, the hardware configuration is also compared. The same time stamp always means the same code. All block types such as OB, PB, FB, FC, DB, UDT etc. are included in the comparison. The result is shown in a dialog in which the block list and the result of the comparison are displayed. Detailed information on each block can also be called up. The result of the comparison can also be output on a printer as documentation.



Note

The Version Cross Checker option is necessary to compare CFC source files.

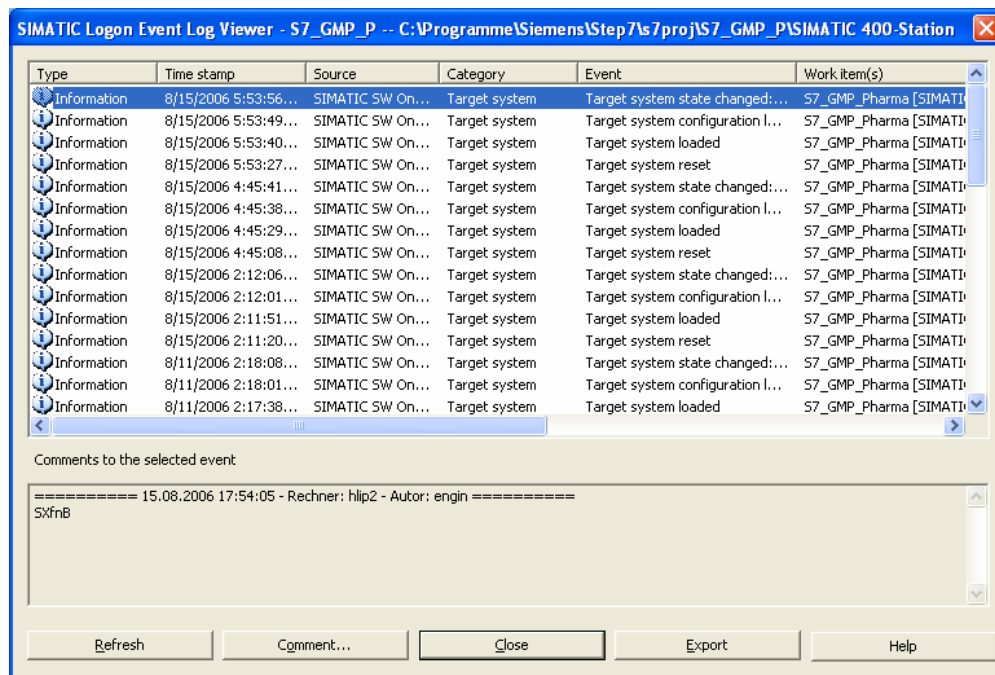
Enabling the change log

In conjunction with the licensed SIMATIC Logon software, a change log can be activated for the STEP 7 project. Online changes with time stamp and logged-on user ID are recorded in the change log. The input of a comment is mandatory.

The following online actions are recorded:

- Activation/deactivation/configuration of access protection and audit trail
- Opening / closing projects and libraries
- Download to target system (system data)
- Selected instructions for downloading and copying blocks
- Activities for changing the operating mode
- Memory reset

The change log can be displayed in the SIMATIC Manager by selecting, for example, the **Station** object in the tree structure and then selecting the **Change log Display** shortcut menu.



The contents of the change log depend on the selection in the structure tree. A change log can be displayed for each station and across the entire project. The displayed log shows the accesses to the target system. In the lower area, the entered mandatory comment, time stamp and user ID are displayed for the selected entry.

Online transfer of changes / changed blocks

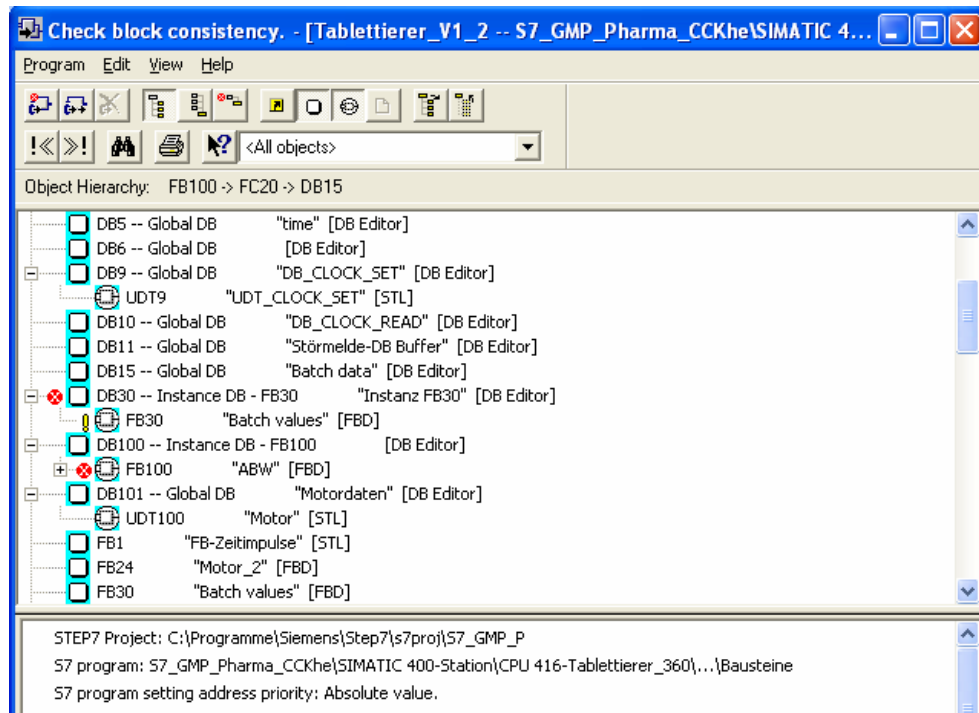
To transfer changed blocks, the CPU must be in a mode in which downloads are permitted (STOP or RUN-P). If the block already exists on the CPU, there is a prompt to ask whether or not the block should be overwritten. Access to the CPU should be protected by a password (see section 4.6.1 "Access protection to the CPU").

Note

Prior to downloading, the block consistency must be checked and, if necessary, reestablished with **SIMATIC Manager -> Edit -> Check Block Consistency**. When downloading in RUN-P, remember that the program is transferred block by block. This can lead to conflicts, for example when a block parameter has been changed. The CPU then changes to STOP during the processing of the cycle. It is therefore recommended to change to "STOP" mode before downloading.

Note

Prior to downloading, the local data requirements per priority must be obtained with the reference list and checked against the parameter settings in HW Config and, when necessary, adapted.



4.6 Access protection

Access protection can be set up both for access to the CPU as well as for the STEP 7 project.

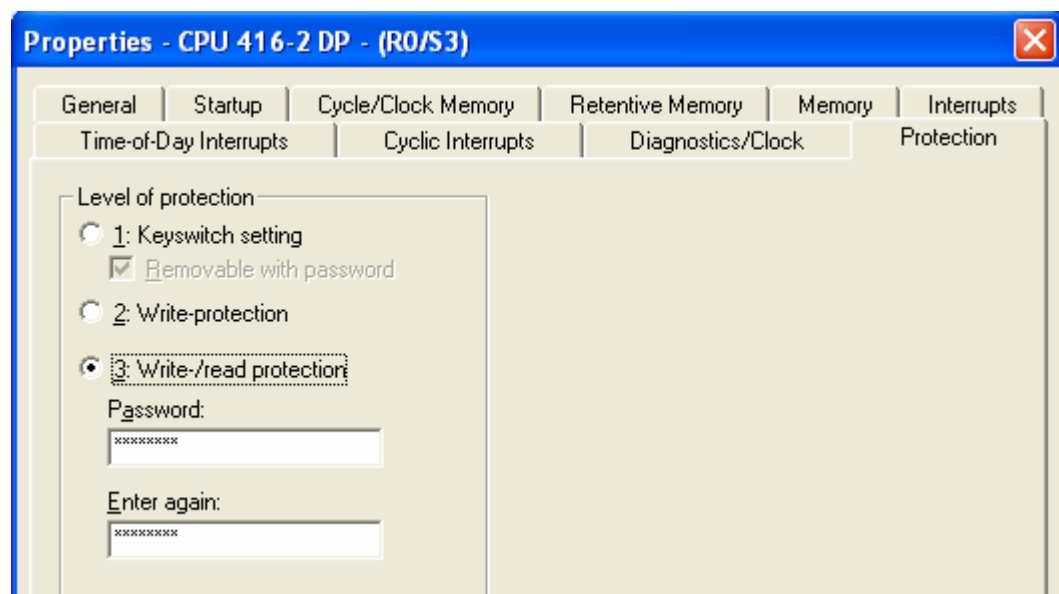


Note

A hardware memory reset on the CPU cannot be prevented either for S7-300 or for the S7-400. An overall reset is the same as removing the module in a de-energized state. Therefore, it is the responsibility of the operator to secure the physical access (e.g. locked cabinet) to the automation systems.

4.6.1 Access protection to the CPU

CPU modules that support this functionality can be protected from unauthorized access by means of a password. The password is set in the **HW Config** Editor. The CPU is selected in the hardware list and the **Object Properties** dialog is opened by double-clicking with the left mouse button. The password is entered in the **Protection** tab.



The dialog shown here depends on the version of the CPU module and is therefore only an example.

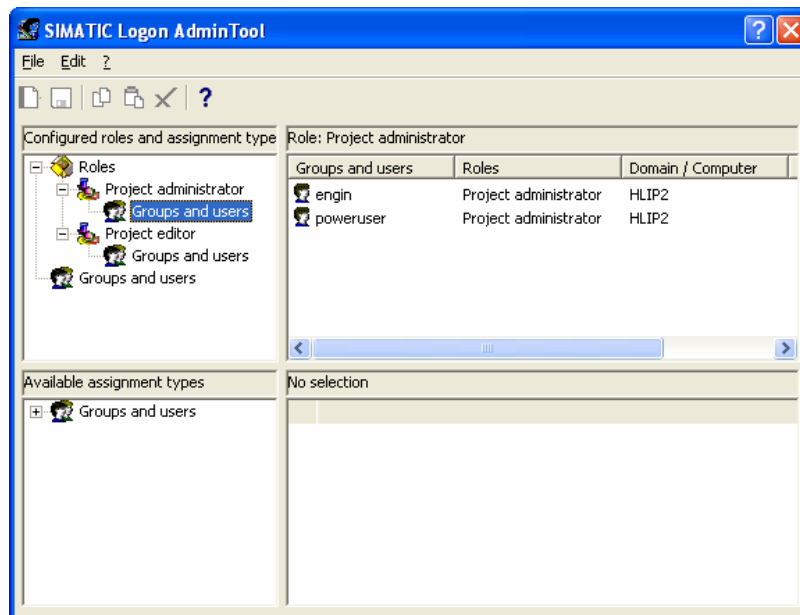
Three different protection levels can be configured for this CPU.

1. Whether PG functions are allowed depends on the keyswitch setting (default setting).
Example: Keyswitch setting RUN: Loading of objects from the CPU to the programming device is permitted, i.e., only read-accessing programming device functions are permitted. Functions for process control, process monitoring and process communication are allowed. All information functions are permitted.
2. Functions for process control, process monitoring and process communication are allowed. All information functions are permitted.
3. The password is queried with every operator activity or can be entered for an entire project session. The rights enabled by the password can be canceled again explicitly but always end when the SIMATIC Manager is exited.

For details of password assignment, refer to the **STEP 7 Help > Establishing an Online Connection and Making CPU Settings > Password Protection for Access to Programmable Controllers** or the help on the Object Properties dialog of the CPU in the HW Config Editor.

4.6.2 Access protection for a STEP 7 project

As of version V5.4 of the STEP 7 basic software, user management can be set up in the STEP 7 project in conjunction with the licensed SIMATIC Logon software. This functionality is available with the menu command **Options > Access Protection > Manage**.

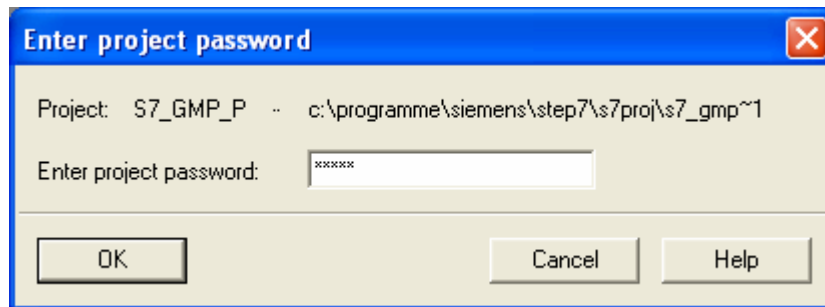


The users must be set up in Windows under **Control Panel > Management > Computer Management > Local Users and Groups**. The users and user groups set up in Windows are displayed in the bottom left-hand area.

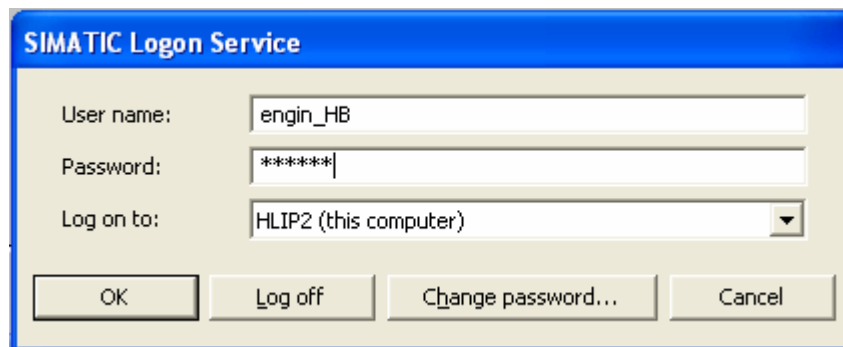
Two roles with different rights are available in the system for the access protection of the STEP 7 projects:

- Project administrator**
 The project administrator has rights for the administration of the users, for the activation/deactivation of the access protection, for the activation/deactivation of the change log and for the complete editing of the project.
- Project editor**
 The project editor has the right for the complete editing of the project, but cannot administrate users or change the settings for the access protection or the change log.

When the access protection is activated for the first time, a dialog box appears for the entry of a project password. For users that are not entered as project editors or do not have SIMATIC Logon installed, the STEP 7 project can only be opened with the project password.



After the access protection has been activated for a STEP 7 project, the SIMATIC Logon service for logging on the user is displayed when the project is opened. SIMATIC Logon controls the user ID and password.



Note

If a user who is not entered in the user administration of the STEP 7 project logs on with the SIMATIC Logon service, a query appears for the project password. The project password should therefore be treated as confidential. Access to and constraints regarding the use of the project password should be regulated (for example in a SOP).

Note

It is recommended that the access protection and the change log be activated after the first qualified version.

4.6.3 Protective Measures in the software

Block protection

Blocks of the types OB, FB, FC and DB can be protected. To assign the Know-How attribute to protect a block, the block is generated in LAD/STL/FBD Editor as an STL source. This function is selected with the **File > Generate Source** menu. A name is set for the block source and the block to be generated is selected. The generated source is stored in the Source folder in the SIMATIC Manager. Double-clicking on a source opens it.

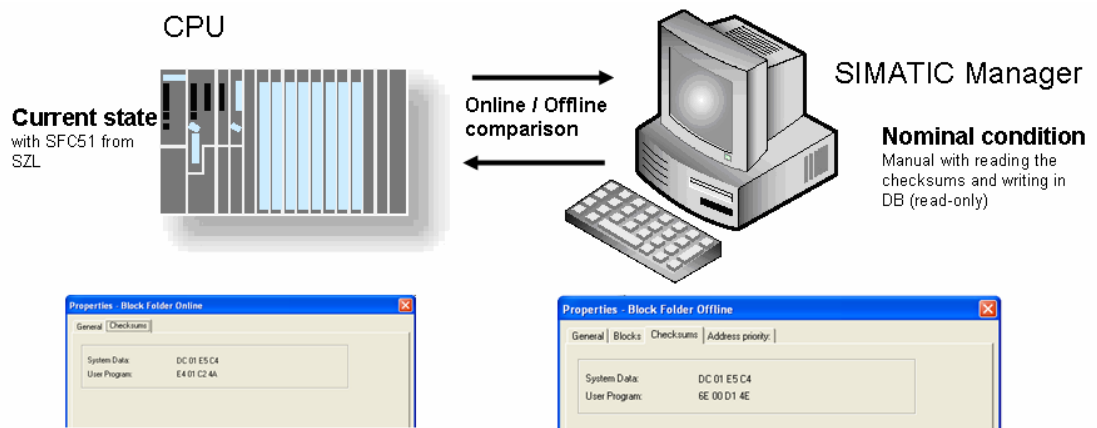
```
FUNCTION FC 20 : VOID
TITLE =Batch
//Title: Batch operating
//Version: V0.1
//Creation date: 04/28/2006
//Date of change:
//Creator: Ip
VERSION : 0.0
KNOW_HOW_PROTECT

VAR_INPUT
  PLCRequest : WORD ; //Recipe request
  JOBControl : WORD ; //Job Control Bits
  PLCControl : WORD ; //PLC Control Bits
```

The KNOW_HOW_PROTECT entry prevents the code being displayed in the LAD/STL/FBD Editor. The code can only be read and edited in the source. Since the blocks run on the automation system without a source, the source can be omitted from the shipped application software. This means that the block can only be changed if the source is available.

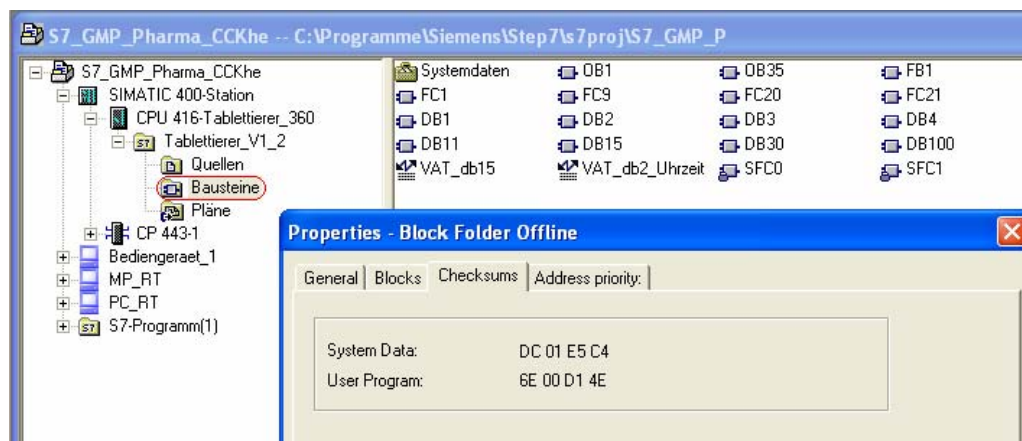
Checking the integrity of the S7 runtime system on a CPU

The STEP 7 basic software automatically creates checksums for system data and the user program. The **system data checksum** is used to check the hardware configuration (for example parameter assignment of the modules, configuration of the connections), the **user data checksum** is used to check the user program. The **user data checksum** does not include the runtime data of the data blocks. If changes are made, the checksum is recalculated. The current checksums can be displayed in the **Object Properties** dialog of the **Blocks** folder.



The following procedure is recommended for evaluating the checksum:

1. Versioning the tested and accepted STEP 7 project (for example following the SAT)
2. Reading out the expected checksums for **system data** and **user data** in the S7 offline project (SIMATIC Manager) in the object properties of the block container



3. Downloading the tested, versioned version to the S7 station
4. Reading out the online actual checksum
 - using the object properties of the online block container
 - from the system status list (SZL) using SFC51
5. Comparison of the expected checksums (offline) / actual checksums (online)
 - Manually using the engineering system
6. Differences in the comparison can, for example, trigger messages in the HMI system
7. Detection of unauthorized changes with the online / offline comparison of individual blocks (section 4.5.4 "Change control") or Version Cross Checker in conjunction with CFC (section 3.3.3.2 "Versioning, Change control")

A description of SFC51 RDSYSST for evaluation of the SZL partial system status list for diagnostic functions is available in the STEP 7 Help as follows:

1. **STEP 7 Help > Calling Reference Helps > Language Descriptions, Help on Blocks, System Attributes > Help on SFBs/SFCs**
2. Under the title **SFCs for Diagnostics > Reading a System Status List or Partial List with SFC 51 "RDSYSST"**, there is a general description of the use, parameter assignment and call of SFC51
3. Under the title **Communication Status Data > SZL_ID W#16#0232 Status data for one communication unit, CPU protection level and operator control settings**, there is a description of the partial list for communication.
4. Under **Data Record of the Partial List Extract with SZL-ID W#16#0232 Index W#16#0004**, there is a description of how checksums are stored.

Proposal for a solution:

After commissioning and acceptance of the user program by the customer, the current checksums for the system data and user data are read out in the SIMATIC Manager and stored manually in a data area (read-only). Each checksum occupies a double word. The version number and version date can also be stored. Calling SFC51 RDSYSST organizes the reading of the checksums from the system status list (SZL) of the CPU. To make the evaluation, the current checksums that have been read out are compared with the content of the manually created data area and corresponding messages are generated.

Detection of electromagnetic interference (S7-400 only)

The hardware of the automation system must be protected against electromagnetic interference according to the installation guidelines (EMC). Electromagnetic interference causes memory errors on the CPU that can lead to spontaneous, uncontrolled software malfunctions. The S7-400 can correct memory errors itself. When memory is corrected, OB84 is called and the cause of the error is displayed in the startup information. To detect memory errors, a message is generated in the HMI system via OB84.

4.7 Audit Trail

The user program (STEP 7 project) is a validated version. Therefore, changes to the automation systems are not made during the runtime.

Unauthorized intervention on the automation system can be detected during runtime. The measures necessary for this are described in section 4.6 "Access protection".

The audit trail for operator input to the process is generated by the operator control and monitoring system. More detailed information can be found in the SIMATIC PCS 7, SIMATIC WinCC or SIMATIC WinCC flexible GMP Engineering manual: "Guidelines for Implementing Automation Projects in the GMP Environment".

4.8 Time synchronization

Within a system, a uniform time reference must be guaranteed to allow messages, alarms etc. to be archived with uniform time stamps. Time synchronization to a standard time is desirable, but not mandatory. Time synchronization is recommended especially for the archiving of data and the analysis of faults (Sequence of Events, SOE).

The time synchronization is based on the standardized world time UTC (Universal Time Coordinated).

In an automated system, either the controller, the operator control and monitoring system or for example SICLOCK can be the time master. SICLOCK supports synchronization with based on GPS or DCF77.

For time synchronization (in the millisecond range) over Ethernet, the SIMATIC time synchronization protocol can be used. Setting the time (in the seconds range) is handled by the NTP protocol (Network Time Protocol).

In the new CPUs (CPU 319) of the S7-300 automation system, the time is synchronized, otherwise it is **set**. Setting the time does not, however, have the same accuracy as time synchronization because frame and script execution times are also involved.

The S7-400 automation systems can be operated time-synchronized on the Ethernet plant bus. One S7-400 is set as master, all others are slaves. These settings are made in the HW Config Editor in the **Object Properties** dialog for the **CPU, CP, DP master, DP slave**.

The procedure for setting the time in the automation system is the same for the Ethernet, Profibus and MPI bus systems. The Ethernet NTP is not available on Profibus and MPI.

Setting the time on the automation system

Setting the time of the SIMATIC WinCC server as time master in an S7 controller is described in detail in entry ID 7802886

(<http://support.automation.siemens.com/WW/view/en/7802886>).

Setting the time of a WinCC flexible operator device as the time master is documented in "SIMATIC WinCC flexible GMP Engineering manual: Guidelines for Implementing Automation Projects in a GMP Environment".

Setting the time of the AS in the Panel (WinCC flexible)

Creating a date/time data area in a data block

A data area is created on the controller consisting of a **"DATE_AND_TIME"** variable.

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Date_Time_OB35	DATE_AND_TIME	DT#90-1-1-0:0	
+8.0	Reserve_1	WORD	W#16#0	necessary for reading the time in WinCC flexible
+10.0	Reserve_2	WORD	W#16#0	necessary for reading the time in WinCC flexible
=12.0		END_STRUCT		

The system time of the automation system is available in this data area to be fetched by WinCC flexible.

The system time is read out with cyclic interrupt OB35. As default, OB35 executes every 100 ms. The time at which execution of OB35 was started is in the local data of the OB and simply transferred to the previously defined data area.

Interface

TEMP

OB35_EV_CLASS

OB35_STRT_INF

OB35_PRIORITY

OB35_OB_NUMBER

OB35_RESERVED_1

OB35_RESERVED_2

OB35_RESERVED_3

OB35_PHASE_OFFSET

OB35_RESERVED_3

OB35_EXC_FREQ

OB35_DATE_TIME

Contents Of: 'Environment\Interface\TEMP'

Name	Data Type	Address	Comment
OB35_EV_CLASS	Byte	0.0	Bits 0-3 = 1 (Coming event), Bit...
OB35_STRT_INF	Byte	1.0	16#36 (OB 35 has started)
OB35_PRIORITY	Byte	2.0	Priority of OB Execution
OB35_OB_NUMBER	Byte	3.0	35 (Organization block 35, OB35)
OB35_RESERVED_1	Byte	4.0	Reserved for system
OB35_RESERVED_2	Byte	5.0	Reserved for system
OB35_PHASE_OFFSET	Word	6.0	Phase offset (msec)
OB35_RESERVED_3	Int	8.0	Reserved for system
OB35_EXC_FREQ	Int	10.0	Frequency of execution (msec)
OB35_DATE_TIME	Date_And_Time	12.0	Date and time OB35 started

OB35 : "Cyclic Interrupt"

The OB35 reads the S7 system time in a cycle of 100ms and transfers the readed time to the data area reserved for WinCC flexible. This data area is polled by WinCC flexible in the configured acquisition cycle. The time is synchronized accordingly.

Network 1: Read time

Reading the system time and transferring in the data area

```

L    LD    12
T    DB11.DBD    0
L    LD    16
T    DB11.DBD    4

```

In WinCC flexible, the data area is specified for the area pointer **date/time controller** and the acquisition cycle for time synchronization is set. The operator device runs the synchronization automatically.

4.9 Time stamping

The S7-300 and S7-400 automation systems are equipped with real-time clock (software clock). This module time is used for time stamping. As of the STEP 7 V5.4 basic version, the CPU time that corresponds to UTC time is converted to the local time of the engineering system or PG. This means that time stamps, for example of entries in the diagnostic buffer are displayed in the local time of the PG/PC.

Time stamping of GMP-relevant data

For production plants in a GMP environment, both GMP-relevant process data and events are archived. Data is always archived with a time stamp. The process values are archived in the operator control and monitoring system (SIMATIC WinCC, SIMATIC WinCC flexible) that adds the time stamp to the process value to indicate the time of archiving.

Two signaling methods are available for the archiving of signaling events, the bit signaling method and the message number method.

- *Bit message procedure*
A message is detected by the controller and signaled to the operator panel via a bit change in a variable. The time stamp of the message is specified by the operator panel when evaluating the variable.
- *Message number procedure*
The controller transfers a message number (possibly also a message text) to display a message. To allow this, messages are configured in STEP 7 using the message blocks *ALARM_S/SQ/D/DQ* or *ALARM, ALARM_8/8P*. The time stamp relates to the time of the call of the alarm block and not directly to the occurrence of the event/alarm.

In the bit signaling method, the acquisition interval, bus runtime and processing time are contained in the time stamp. Messages present for a time shorter than the acquisition cycle are lost. The bit message procedure and limit value monitoring can be used in a WinCC single user system. **The bus load is high (operator system polls).**

In redundant systems or WinCC and WinCC flexible systems with several operator stations, chronological signaling is used for coordinated acknowledgment and transmission.

Time stamping via the message number method is much more precise. **The bus load is low (AS signals active).** If high accuracy of the time stamp for the occurrence of the alarm event is important (time stamping in the ET200M), the alarm sample ALRM7PBT can be purchased for the S7-400 automation system (Entry ID 20614217). This example records the original time stamp when the alarm event occurs and buffers the accumulating alarm events for transfer to the HMI system WinCC.

There are, however, restrictions relating to the message number procedure, as follows:

- SIMATIC WinCC flexible supports only the message number procedure with the alarm blocks *ALARM_S/SQ/D/DQ*. This means that in the most powerful current CPU3xx, a maximum of 60 messages can be generated and in the most powerful current CPU 4xx, a maximum of 200 messages.
- A CPU4xx can send up to 10000 instances to a SIMATIC WinCC system using the message number procedure, which means up to 80,000 messages via the alarm blocks *ALARM, ALARM_8/8P*.

Note

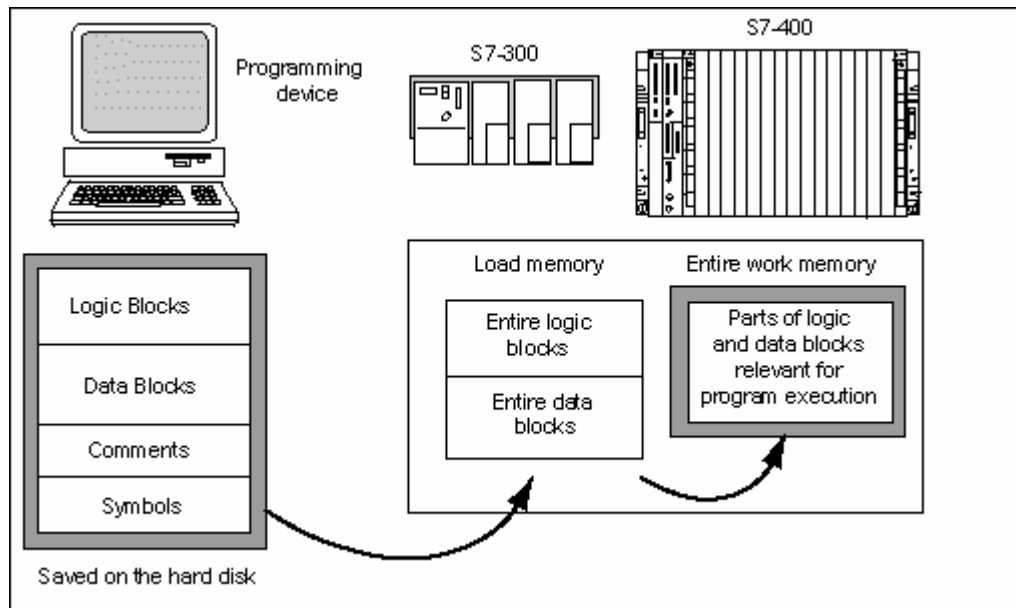
The use of the signaling method is a decisive criterion for the selection of the hardware components.

4.10 CPU storage

When the user program is downloaded from the programming device to the CPU, only the logic and data blocks are downloaded to the work memory of the CPU. The symbolic address assignment (symbol table) and the block comments remain on the PG.

The memory of a CPU is divided into load memory and work memory. To ensure fast processing of the user program and to put as little load on the restricted work memory, only the parts of the blocks relevant for program execution are loaded in work memory.

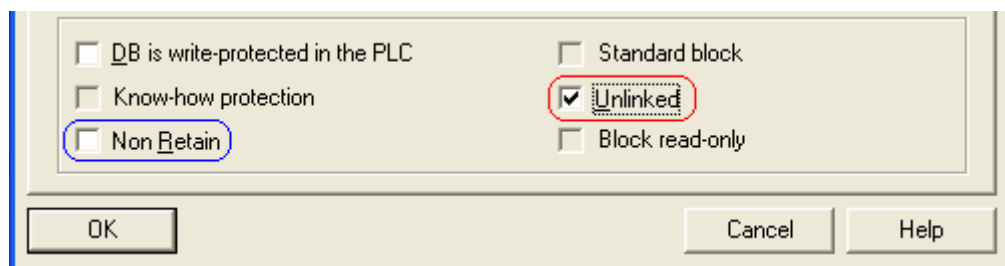
The following schematic illustrates how the program is loaded into CPU memory.



4.10.1 Memory concept of S7-300 CPUs

The overall size of the work memory (RAM) and the retentive allocation depend on the particular CPU. The CPU is equipped with a Micro Memory Card (MMC) as load memory to which the user program is loaded. If a suitably large MMC is selected, the project data of the complete STEP 7 project can also be loaded on it as an archived file.

Data blocks that are not relevant during runtime (for example those containing recipe data) can also be loaded to the MMC. To do this, the **unlinked** property is activated in the object properties of the data block.



Overwriting a data block stored on the MMC is only possible with system functions. This should not be done cyclically but only sporadically since the number of writes to an MMC is limited.

The advantage of the MMC is that data is not lost during a power down although the S7-300 does not have battery backup. Runtime data from the work memory is transferred to the MMC with the residual voltage. However, only the amount of runtime data, limited by the size of the retentive memory area of the CPU, is written back.

When data is written back, the content of data blocks marked as **Non-Retain** is backed up first.



Note

The size of the retentive memory area available for reading back runtime data if there is a power down depends on the selected CPU. Generally, the retentive memory area is only half the work memory. This can lead to a loss of runtime data.

Note

An MMC is designed for a limited number of write operations (approximately 100,000). Each loading operation, each power failure means a write access. The number of write accesses is usually not critical when it is ensured that there are no cyclic write accesses to the MMC.

When the maximum number of write accesses has been reached, the MMC must be replaced with a new MMC. However, this can only be performed with the power turned off, which means runtime data loss.

4.10.2 Memory concept of S7-400 CPUs

The memory configuration depends on the type of CPU. Load memory can be expanded with flash or RAM. If a suitably large memory size is selected, the project data of the complete STEP 7 project can be loaded to it as an archived file.

If there is a power down, the user program and runtime data of the S7-400 automation system can be battery backed. A lithium battery is used for backup. A second redundant battery can be inserted. The backup voltage is monitored by the system. If there is a fault in the power supply / backup, the system calls OB81. (See **STEP 7 Help > Calling Reference Helps > Language Descriptions, Help on Blocks ... > Help on OBs**)

The S7-400 can also be equipped with two redundant power supplies, for example one at 220V and one at 24V. Each can be backed up by two batteries.

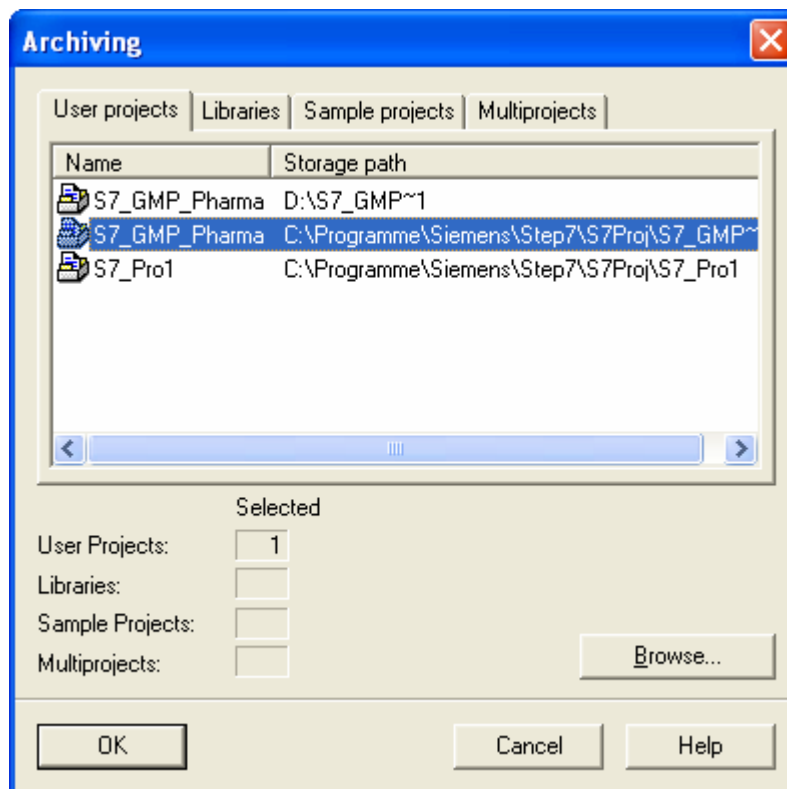
4.11 Backup / restoring system/application software

In order to be able to fall back on the created software in emergency situations, backup copies of the software versions must be made at regular intervals during the configuration phase. It is also advisable to make a backup of the system partition containing the engineering system with the operating system, SIMATIC STEP 7 basic system, etc.

Backing up the application software

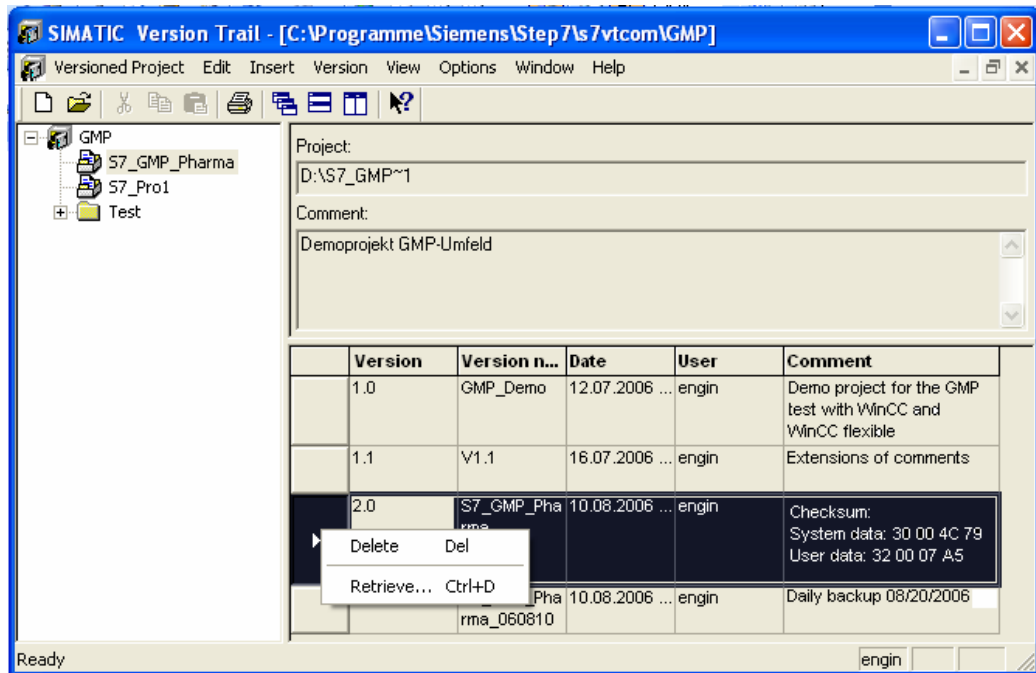
The STEP 7 application software can be backed up in different ways:

- The **File > Save As** menu command makes a copy of the STEP 7 project, for example a backup copy. Adequate storage space should be available. At regular intervals, the **With Reorganization** check box should be enabled to reduce the storage requirements for the project. A backup copy can be edited with the **File > Open** menu command.
- During the engineering phase, it is advisable to enable the check box **Archive automatically on opening project or library**.
- In SIMATIC Manager over the **File > Archive...** menu
With this function the STEP 7 project is archived and compressed. The tool used to archive can be configured in the **Options > Settings > Archive** menu. (The archiving program *PKZIP* is the default). Before editing the STEP 7 project it must be retrieved from the archive.



For information on archiving, refer to the STEP 7 Help **Printing and Archiving > Archiving Projects and Libraries > Requirements for Archiving**.

- In the licensed Version Trail software (see section 4.5.3 "Versioning of the application software"), the STEP 7 project is backed up with a major and minor version. Each version can be decompressed again with Version Trail.



•

Note

The Version Trail option is recommended for version management of the STEP 7 project.

5 Additional software components

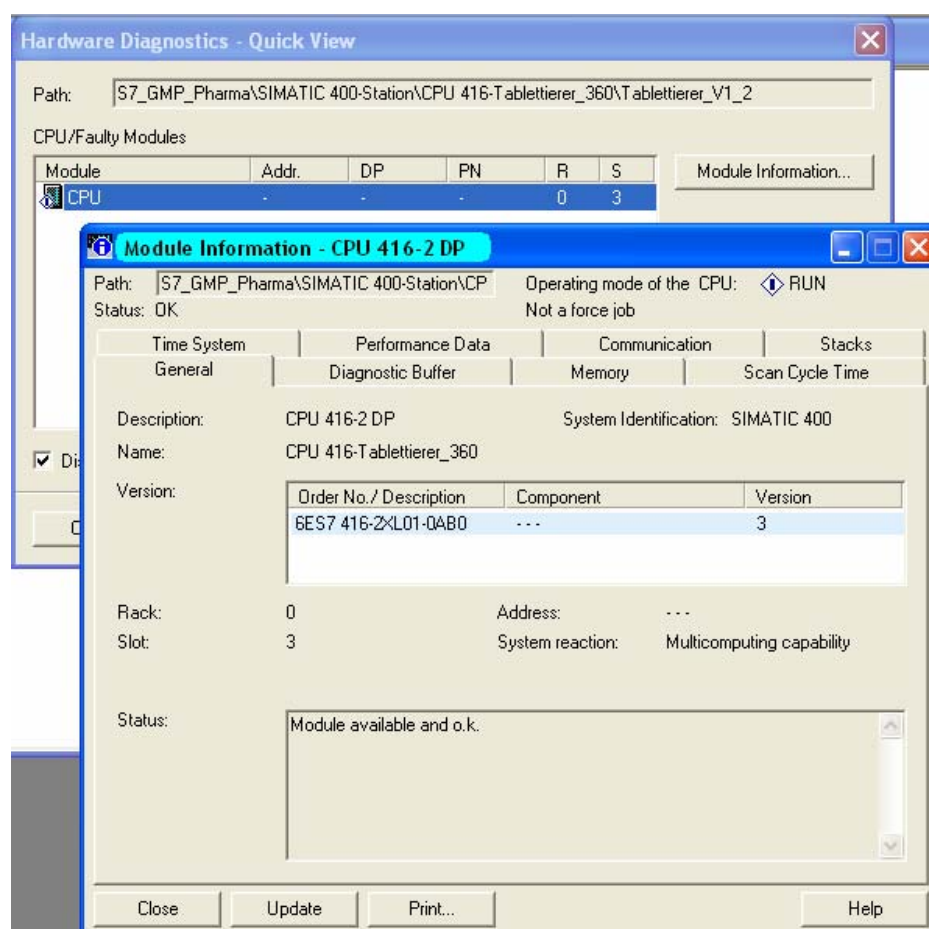
5.1 Diagnostic tools

User-friendly functions for hardware and software diagnostics are already integrated in the STEP 7 basic software. Primarily, these functions are used for commissioning the user program and diagnostics if errors occur.

Hardware diagnostics

Hardware diagnostics is performed in the STEP 7 basic system with the SIMATIC Manager or with the HW Config Editor. Various examples are listed in the following. The diagnostic options are described in detail in the STEP 7 help system under Diagnostics.

- The **View > Online** menu, for example, marks modules with an informative icon. The view changes to the online view of the project and access to the target system is possible.
- The **PLC > Diagnostics/Settings > Hardware Diagnostics** menu opens a dialog in which module information is displayed.



If there are problems on the module, the problems are listed in the **Status** box. The precise cause of the problem can be analyzed in the **Diagnostic Buffer** tab.

Software diagnostics

Variable tables and the status display of the individual blocks are used for the software diagnostics or software test. Various examples are listed in the following. The debugging options are described in detail in the STEP 7 help system under **Debugging**.

Variable tables

Variables, data areas, bit memory, timers and counters are put together in any combination in variable tables so that their current values can be read out. The values can be manipulated via the control function. The variable tables are saved under a name and can be called at any time.

	Address	Symbol	Display format	Status value	Modify value
1	DB15.DBW 2	"Batch data".PMQStatus	BIN	2#0000_0000_0000_0000	//2#0000_0000_0000_0000
2	DB15.DBW 10	"Batch data".PLCRequest	BIN	2#0000_0000_0000_0000	
3	DB15.DBW 12	"Batch data".PLCControlBit	BIN	2#0000_0000_0000_0000	
4	DB15.DBW 14	"Batch data".JobControlBit	BIN	2#0000_0000_0000_0000	//2#0000_0000_0000_0000
5	DB15.DBD 20	"Batch data".Ingredients1	FLOATING_...	9.0	//50.0
6	DB15.DBD 24	"Batch data".Ingredients2	FLOATING_...	6.9	//80.0

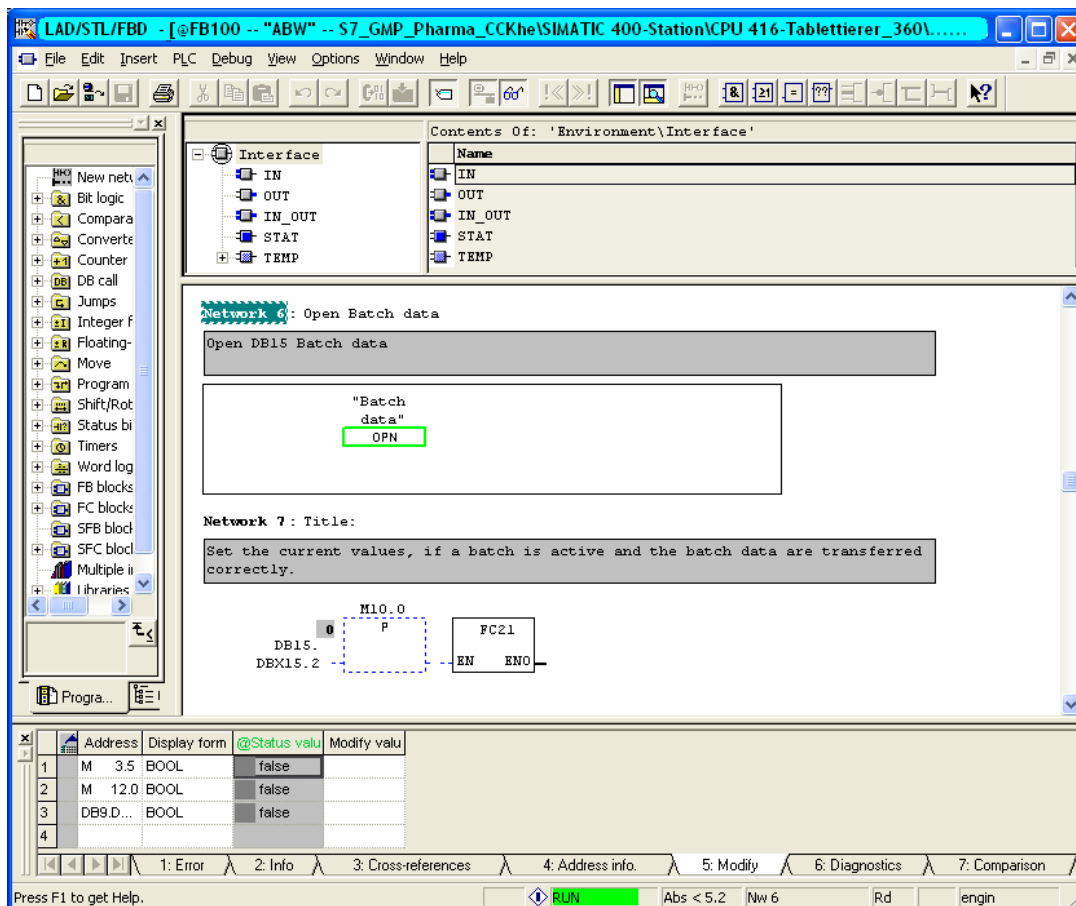


Note

The control function in the variable table is very dangerous and only permitted during the development phase of the user program. The modify function should be prevented in runtime with the **write-protection** or **Write/read protection** setting in the object properties of the CPU. (See also section 4.6.1 "Access protection to the CPU")

Monitoring the status of individual blocks

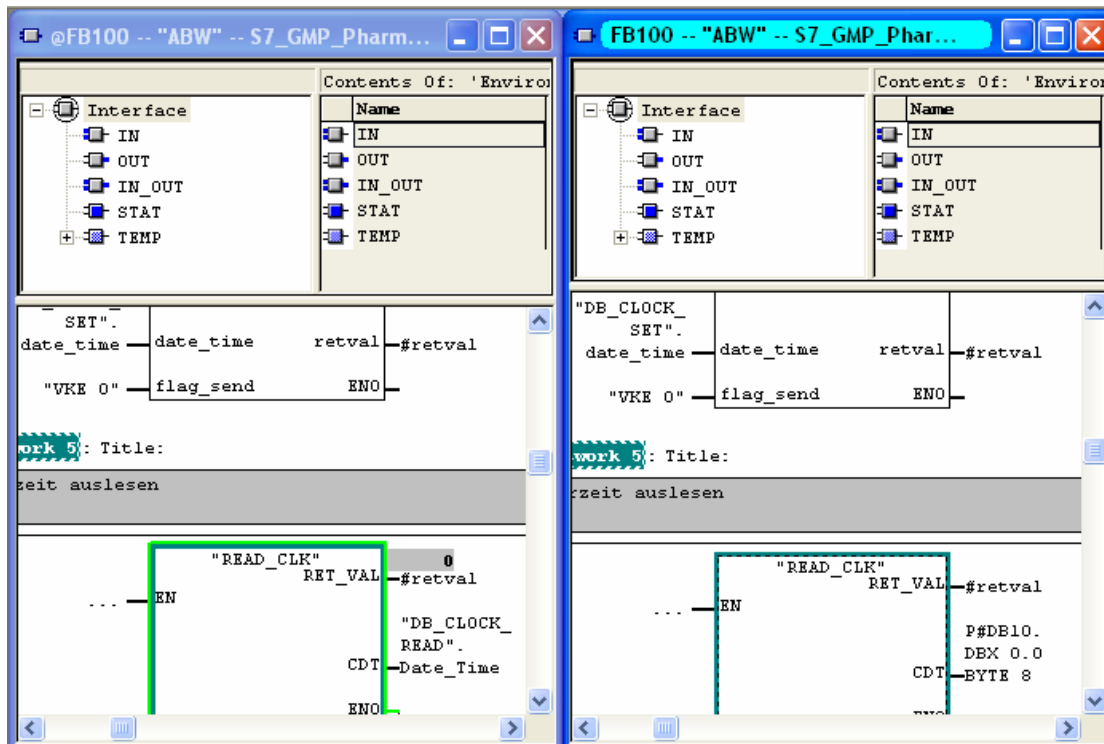
To monitor the program execution, individual blocks can be observed online. To use this function, the block is opened in the LAD/STL/FBD Editor. Monitoring of the block is enabled or disabled with the **Monitor on/off** button. Active connections are now displayed in color in the block display and inactive connections are shown as being interrupted.



Note

The modify function in the LAD/STL/FBD Editor is extremely dangerous and is only permitted during the development phase of the user program. The modify function should be prevented in runtime with the Write-protection setting in the object properties of the CPU see also section 4.6.1 "Access protection to the CPU").

In the online/offline comparison, the block on the target system is displayed beside the block in the project. Displaying them side-by-side makes it easier to recognize differences.



5.2 Simulation tools

The licensed S7-PLCSIM optional package simulates an automation system on the computer or programming device (field PG) on which the running program is tested. Since the simulation is implemented entirely within the STEP 7 software, no S7 hardware (CPU or signal modules) is required. Programs for S7-300 and S7-400 CPUs can be tested with the simulated S7-CPU and bugs eliminated.

The application has a simple user interface for monitoring and changing various parameters used in the program (for example for activating/deactivating inputs). The various applications from the STEP 7 software can also be used while the simulated CPU is executing the program. Variables can, for example, be monitored and modified using a variable table.

Note

It is possible that brand new SFBs/SFCs of the CPUs are not supported. The integration of updated or new functions takes time to be included in S7-PLCSIM.

5.3 SIMIT simulation software

The SIMIT simulation software allows a software test on a simulation platform without needing the actual field devices. SIMIT simulates field devices and allows not only simple signal tests at the touch of a button but also complex tests at the drive level. Along with the S7-PLCSIM programmable controller simulation software for simulating the CPU of an automation system, cost-effective software tests can be performed without automation systems (AS) and field devices. This means, for example, that a factory acceptance test (FAT) can be performed by the software provider. The factory acceptance test is used to detect and fix possible bugs prior to commissioning and brings about a reduction in the commissioning time.

5.4 Rewiring S7 programs

The STEP 7 basic software supports rewiring of inputs/outputs, bit memory, timers, counters, functions and function blocks. During the rewiring, an old address is replaced by a new one. The rewiring is documented in an information file.

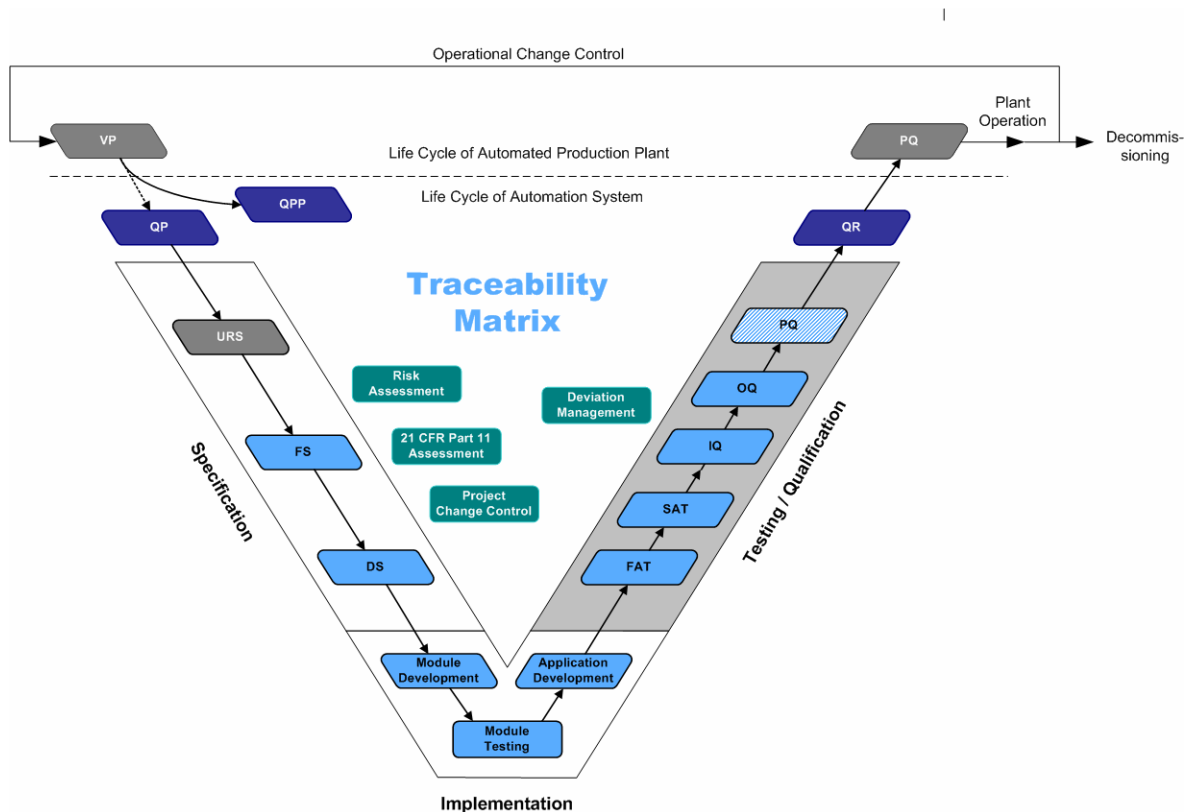
Note

If a *symbols leading* is set, the program must be recompiled with *Check Block Consistency*.

6 Supporting functions during qualification

6.1 Introduction

This chapter concentrates on system functions that support qualification activities. These activities take place in the "Test / Qualification" phase and they are highlighted in the life cycle model below in the right hand part of the graphic.



The aim of the qualification is to provide documented proof that the system was set up according to the specifications and that all specified requirements have been met. The qualification describes, executes and finally evaluates all the activities necessary for this. Various standard functionalities of SIMATIC STEP 7 can be used to support the qualification during IQ and OQ.

6.2 Qualification of automation hardware

The design specification of the installed hardware is used to set up the system according to detailed specifications and adherence to these specifications can and must be verified during the subsequent system tests. The design specification describes all hardware used with information such as order number, firmware/version, serial number, installation location, etc. The components such as the CPUs used, input and output cards, interfaces to third-party systems, etc. are listed below.

Qualification of field devices

In the qualification of field devices, checks are necessary to ensure that the requirements of the hardware design specification were implemented. This means verifying the following:

Vendor
Order number
Serial number
Function of the field device
Intended location
Process tag name
Type of connection electrical / bus type
Physical attachment
Address number
Unit of measure
Measuring range

Note

A visual inspection is made of the field device.

Qualification of the automation hardware

In the qualification of automation hardware, checks are necessary to ensure that the requirements of the hardware design specification were implemented. All hardware components must be configured in the hardware configuration of SIMATIC STEP 7 as stipulated in the design specification. These include:

- Number of racks
- Verifying the hardware components used (CPU, CP, etc.)
- Number of distributed I/O stations
- Interfaces to third-party systems
- Verifying the order numbers of the hardware used
- Address description
- Symbolic naming of inputs/outputs
- Etc.

Note

The hardware configuration can be printed out and used as qualification verification (IQ/OQ) of the installed hardware components. A visual inspection of the installed hardware can be made at the same time. The hardware used must match that in the control cabinet documentation.

Qualification of the network structure

In the qualification of network structure, checks are necessary to ensure that the specifications of the hardware design specification were implemented. All connections must be configured in the SIMATIC NetPro configuration of SIMATIC STEP 7. These include:

- Name of: Station, PC, AS, clients etc.
- Communications module, type of connection and communication partner (Ethernet, PROFIBUS, serial etc.)
- MAC address (when using the ISO protocol on the plant bus)
- TCP/IP address and subnet mask (when using clients)
- PROFIBUS addresses
- Etc.

Note

The network configuration can be printed out and used as qualification verification (IQ/OQ) of the configured network structure. A visual inspection of the configured network structure can be made at the same time.

6.3 Qualification of automation software

6.3.1 Qualification of standard software

In the qualification of standard software used, checks are necessary to ensure that the requirements of the software design specification were implemented. These include:

- Operating system of the engineering system
- SIMATIC STEP 7 standard basic packets (engineering system) and optional packages (CFC, SCL, SIMATIC Logon, Version Trail, DOCPRO, etc.)
- Standard libraries

Note (operating system)

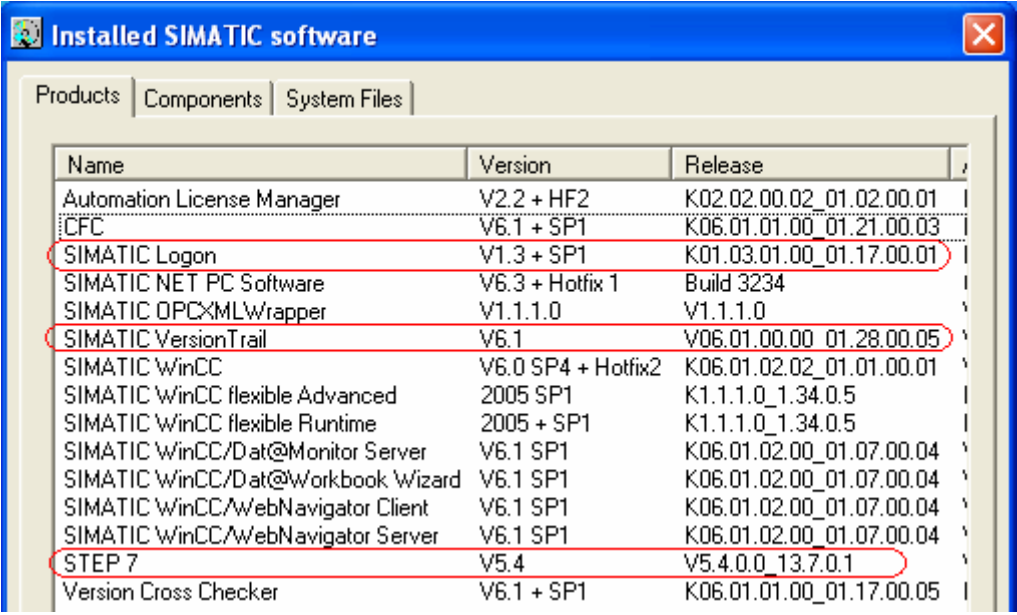
The installed software can be verified by operating system functions. The information can be found in the **Control panel > Software**. All installed software components are displayed here. A screenshot can be printed and used for the qualification (IQ/OQ).

Note (SIMATIC software)

The "Installed Software" tool can be used to check the installed SIMATIC Software. The tool provides information on the SIMATIC software currently installed on the computer. The installed components can be printed and used for the qualification (IQ/OQ).

6.3.2 Installed SIMATIC software STEP 7

A documentation of the installed software packages must be made for the engineering system used for the configuration of the automation systems. Detailed documentation of the installed SIMATIC software can be found under **Programs > SIMATIC > Product notes > Installed software**.

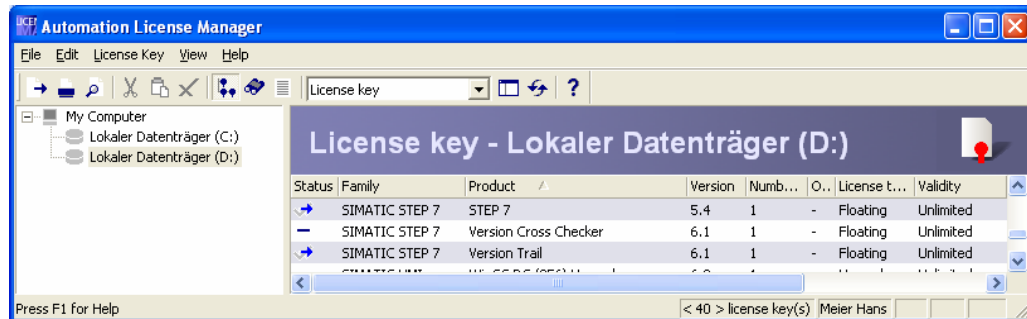


Name	Version	Release
Automation License Manager	V2.2 + HF2	K02.02.00.02_01.02.00.01
CFC	V6.1 + SP1	K06.01.01.00_01.21.00.03
SIMATIC Logon	V1.3 + SP1	K01.03.01.00_01.17.00.01
SIMATIC NET PC Software	V6.3 + Hotfix 1	Build 3234
SIMATIC OPCXMLWrapper	V1.1.1.0	V1.1.1.0
SIMATIC Version Trail	V6.1	V06.01.00.00_01.28.00.05
SIMATIC WinCC	V6.0 SP4 + Hotfix2	K06.01.02.02_01.01.00.01
SIMATIC WinCC flexible Advanced	2005 SP1	K1.1.1.0_1.34.0.5
SIMATIC WinCC flexible Runtime	2005 + SP1	K1.1.1.0_1.34.0.5
SIMATIC WinCC/Dat@Monitor Server	V6.1 SP1	K06.01.02.00_01.07.00.04
SIMATIC WinCC/Dat@Workbook Wizard	V6.1 SP1	K06.01.02.00_01.07.00.04
SIMATIC WinCC/WebNavigator Client	V6.1 SP1	K06.01.02.00_01.07.00.04
SIMATIC WinCC/WebNavigator Server	V6.1 SP1	K06.01.02.00_01.07.00.04
STEP 7	V5.4	V5.4.0.0_13.7.0.1
Version Cross Checker	V6.1 + SP1	K06.01.01.00_01.17.00.05

The list provides information on the installed software products, software components and DLLs on the local computer. This information can be used, for example, to include the installed software in the installation qualification.

6.3.3 Installed licenses of SIMATIC STEP 7

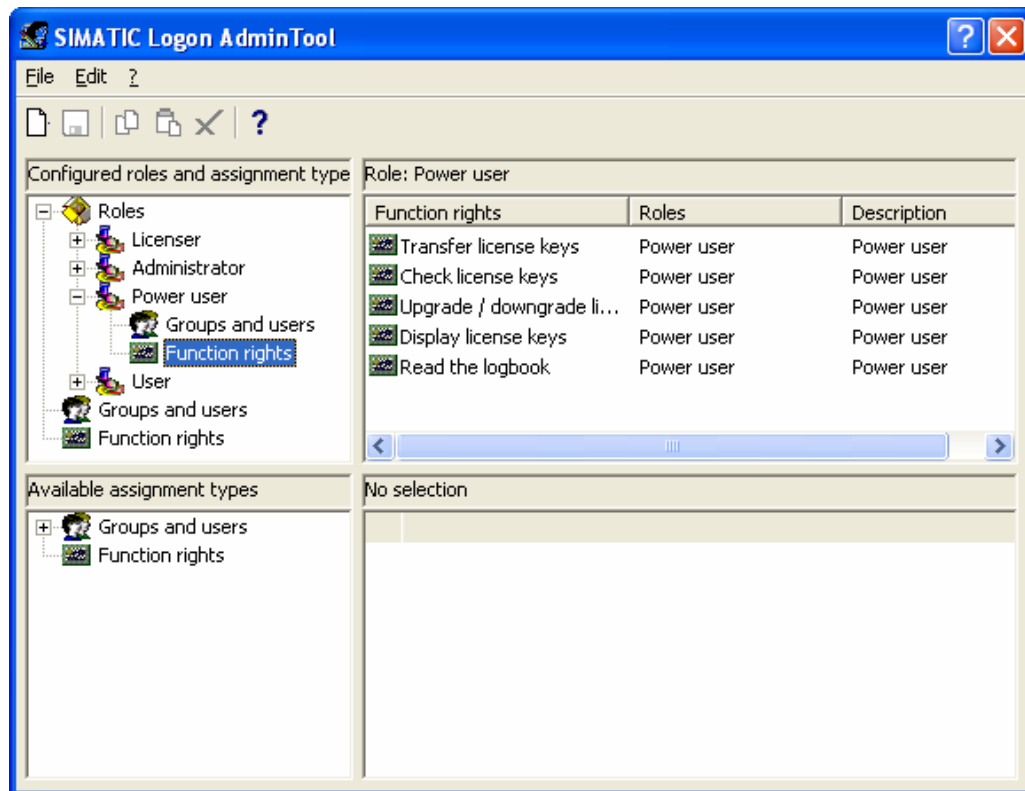
The *Automation License Manager* program provides information on the licenses currently installed on the engineering system. To view the licenses, open the Automation License Manager. Select the PC partition on which the licenses are installed on the left-hand side of the Explorer bar. The available SIMATIC licenses of the system are then shown on the right.



In conjunction with the SIMATIC Logon option, an access protection can be activated for the Automation License Manager application. To do this, select **File > Settings > Activate SIMATIC Logon access protection**.

The users are assigned roles in the **File > User Management** menu. Certain functions are associated with roles, such as Licensor or Power user, that can be performed by the assigned users in the application.

The users and user groups must first be created in the Windows User Management.



When the Automation License Manager is started, the SIMATIC Logon service opens for logging on a user. SIMATIC Logon controls the user ID and password.

Note

The installed licenses must correspond to the requirements defined in the specification.

The installed licenses can be printed and used for the qualification (IQ/OQ).

6.3.4 Qualification of the application software

In the qualification of application software, checks are necessary to ensure that the requirements of the software design specification were implemented. Test descriptions must be agreed with the user (e.g. for FAT/SAT) and generated. These test descriptions must be created individually to meet the software design specifications.

As a minimum, the following must be checked and tested and can be used as a reference for the qualification:

- Check of the name of the application software
- Check of the technological hierarchy (plant, plant section, technical equipment, individual control element, etc.)
- Software module test (typical test)
- Check of the communication with other subscribers (third-party controllers, MES systems, etc.)
- Check of all inputs and outputs
- Check of all control modules (individual control level)
- Check of all equipment phases and equipment operations (technical functions)
- Check of the relationships between modes (MANUAL/AUTOMATIC switchovers, interlocks, start, running, stopped, aborting, completed, etc.)
- Check of the measuring point names
- Check of the visualization structure (P&I representation)
- Check of the operating philosophy (access control, group rights, user rights)
- Check of the archiving concepts (short-term archives, long-term archives)
- Check of the message concept
- Check of the trends, curves
- Check of the time synchronization

Index

2

21 CFR Part 11..... 1-7

A

Access control 3-8
Access protection 2-7, 4-29
Access protection - CPU 4-29
Access protection - Step 7 project 4-30
Audit trail..... 2-10, 4-35

B

Backup 2-11
Backup process data 2-12
Backup user software 2-11
Backup, Restore - System /
 Application software 4-41
Biometric systems 2-7
Bloc container..... 4-6
Bloc properties..... 4-7
Block protection 4-32
Block title and block comment 4-13

C

Change control 2-5, 4-25
Change Control 3-8
Change log 4-26
Changing system software 4-18
Configuration control 2-4
Configuration identification 2-4
Configuration management 2-4, 4-18
CPU storage..... 4-39

D

Diagnostic tools 5-1
DOCPRO..... 4-15

E

Electronic signature 2-8
Engineering - Basic principles 4-4
EU GMP Guideline 1-7, 1-8

F

FAT..... 1-6
FDA 1-7
Firmware update..... 4-21
Functional specification 1-5

G

GAMP..... 1-7, 1-8
GMP requirements 2-1

H

Hardware - Dimensioning 3-2
Hardware - Selection criteria 3-3
Hardware categorization..... 2-2
Hardware diagnostics 5-1

I

Implementation..... 1-5
integrated HMI system..... 4-11

L

Life cycle model..... 1-2
Line comment..... 4-14

M

Module replacement..... 4-21

N

NAMUR 1-7, 1-8
Network title and network comment 4-13

O

Online / Offline comparison 3-8
Online/Offline comparison 4-25

P

Password..... 2-7, 2-9
Programming - Procedure 4-4

Q

Qualification..... 1-6, 6-1
Qualification application software 6-7
Qualification automation Hardware..... 6-2
Qualification Field devices..... 6-2
Qualification network structure 6-3
Qualification plan..... 1-4
Qualification report 1-6
Qualification standard software 6-4
Quality and project plan..... 1-4, 1-9

R

Replacing / changing the hardware / firmware...	4-21
Retrieving archived data	2-13
Rewiring S7 programs	5-7
Rules and Conventions	4-8

S

SAT	1-6
SIMATIC Logon	3-8
SIMIT	5-6
Simulation tools	5-5
Smart card	2-7
Software – additional engineering packages	3-5
Software - additional packages for GMP compliance	3-8
Software - Basic engineering	3-5
Software - required / optional packages	3-5
Software categorization	1-10, 2-2
Software Categorization	4-2
Software creation.....	4-4
Software diagnostics	5-2
Software installation	4-3
Software interlocks/safety.....	4-10
Specification	1-4
Specification - design specification.....	1-5
Symbolic name	4-14

System specification.....	3-1
---------------------------	-----

T

Third-party component	2-13
Time stamping	4-38
Time synchronization.....	2-10, 4-36
Typicals	2-6

U

Update, Service Pack, Hotfix.....	4-18
Upgrade (Migration)	4-19
User ID	2-7, 2-9
User management.....	2-7
User requirements specification	1-5

V

Validation plan.....	1-4
Validation report	1-6
Version control	2-4
Version Cross Checker.....	3-9
Version Trail	3-8
Versioning	3-8, 4-23, 4-33
Versioning application software.....	4-22
Versioning - blocs	4-23
Versioning - project	4-23