SIEMENS

SITRAIN Training for Automation and Drives

SIMATIC HMI

Operating and Monitoring with WinCC

Course ST-BWINCCS

AL: N ECCN: N

Export Regulations

AL Number of European or German export list.
Goods with labels not equal to "N" are subject to export authorization.

ECCN Number of US export list (Export Control Classification Number). Goods with labels not equal to "N" are subject to re-export authorization for export to certain countries.

Indication Goods labeled with "AL not equal to N" (here: technical documentation) are subject to European or German export authorization when being exported out of the EU.
Goods labeled with "ECCN equal to N" (here: technical documentation) are subject to US re-export authorization.
Even without a label, or with label "AL:N" or "ECCN:N", authorization may be required due to the final whereabouts and purpose for which the goods are to be used.
Decisive is the export labels stated on order acknowledgements, delivery notes and invoices.

Name: _____

Course: from _____ to _____

Instructor:

Infoline Tel: 01805 23 56 11
Fax: 01805 23 56 12
Internet: http://www.siemens.com/sitrain

IID-No.:
Release A6.0 (for WinCC Version 6.0)

System Overview

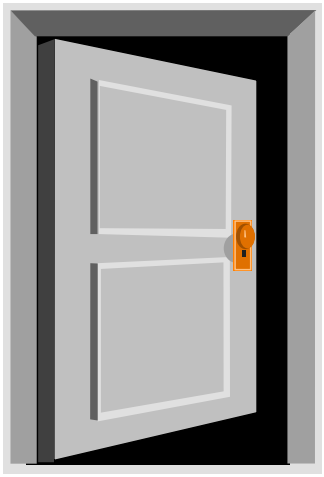
SIMATIC HMI

Date: 27.10.2003
File: swincc01e.1

SITRAIN Training for Automation and Drives

Contents **Page**

Objectives



- Application Possibilities
- Forms of Delivery
- Core Functions
- Architecture

SIMATIC HMI

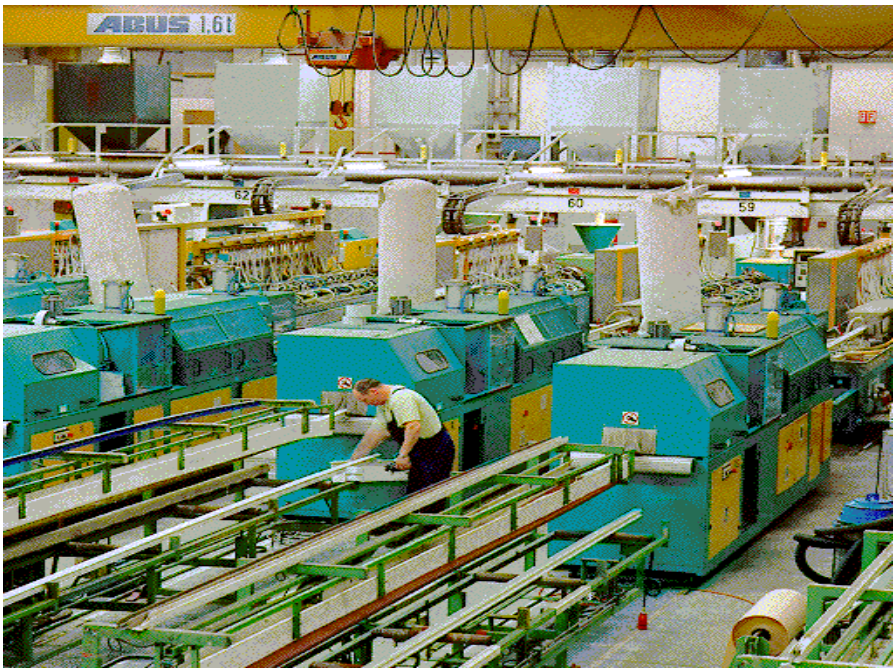
Date: 27.10.2003
File: swincc01e.2

SITRAIN Training for Automation and Drives

Upon completion of this chapter, the participant will:

- know the customer‘s benefits through using WinCC
- be able to evaluate the application possibilities of WinCC for his task
- be able to determine the possible system configurations
- know how to use the key disk
- know the forms of delivery of the system software
- know the core functions of the system software
- know the architecture of the WinCC software
- know the openness structure
- know the programmable logic controller (PLC) drivers and Microsoft interfaces
- be able to classify the option packages
- know the core blocks

Customer Benefits



WinCC because of the following features:

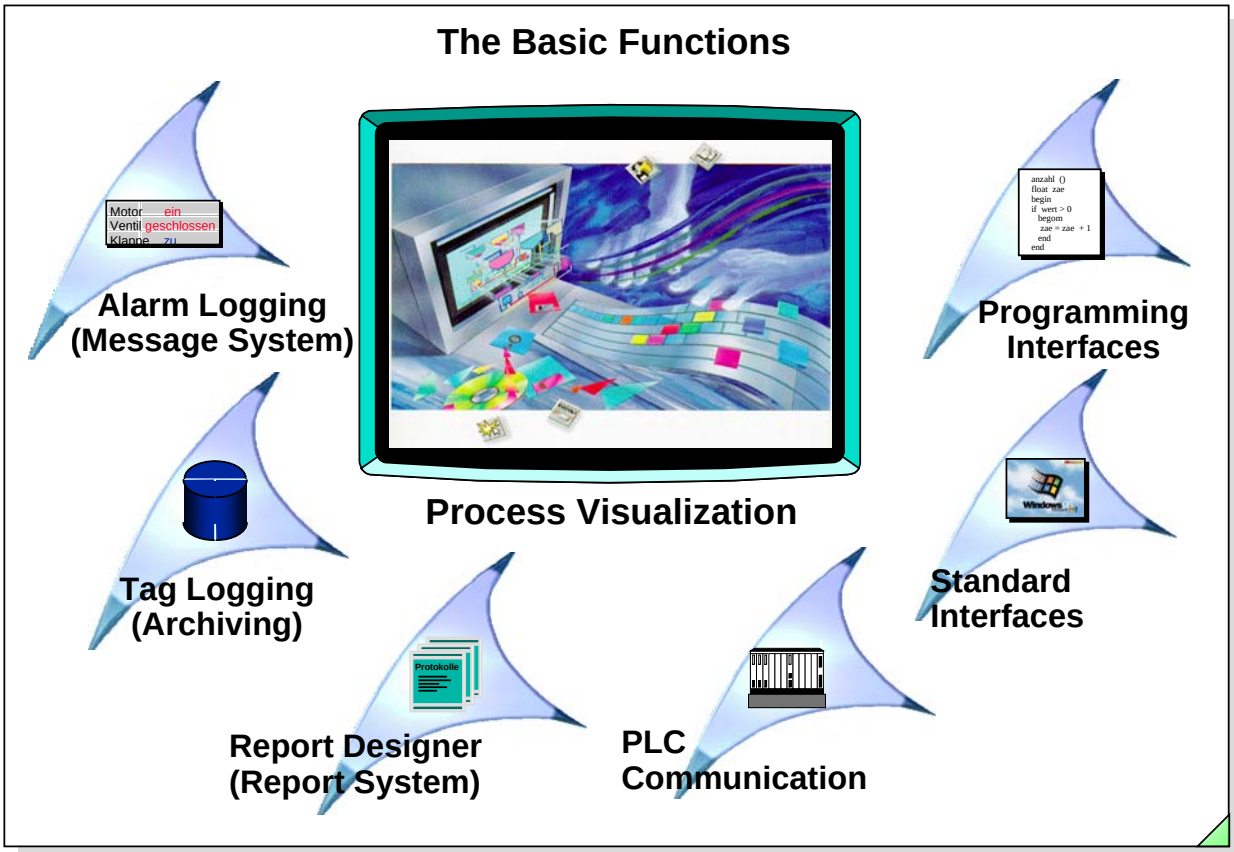
- standard operating system from Microsoft for Windows XP Professional and Windows 2000 Professional and Server
- modern 32 bit operating system
  - ⇨ multitasking
  - ⇨ safe for industrial applications
  - ⇨ fast
- uses Windows equipment (printer, driver, etc.)
- supports data exchange with other Windows applications using OLE, for example
- ODBC, SQL, OLE-DB, ADO database interfaces
- API programming interface
- integration of OCX and Active X Controls
- uses the OPC interface
- uses the database MS SQL Server 2000
- ANSI C, VBA, VBS programming interfaces
- ...

Application Possibilities of WinCC

Date: 27.10.2003
File: swincc01e.4

From energy management control systems for buildings to industrial systems

- Energy management control systems for buildings (ventilation, air conditioning)
- Materials handling technology
- Mechanical engineering
- Water resources and waste water management
- Steel industry
- Iron producing industry
- Foundries
- Cold rolling mills
- Steel casting industry
- Chemical industry
- Pulp, paper, and corregated cardboard production
- Breweries, malt factories, and soft drink production
- Foodstuffs and luxury food industry
- Milk processing industry
- Sugar industry
- etc.



Date: 27.10.2003
File: swincc01e.5

Basic Functions

This slide illustrates the functionality of a modern process visualization with

- full graphics display.
- alarm display, alarm archiving and alarm logging.
- measured value acquisition (archiving functions, data compression, minimum and maximum values etc.).
- Report Designer for documenting machine sequences and process sequences with individual reporting.
- process communication to various PLC systems.
- standard interfaces for database and data processing, for example, in Microsoft programs.
- programming interfaces (API) are available in all WinCC application modules and allow access to data and functions.

Complex visualization tasks can be solved with the ANSI-C programming language and/or with Visual Basic Script.

Options

For useful extensions of the WinCC Basic Software, for example, Server expansion, redundancy, etc.

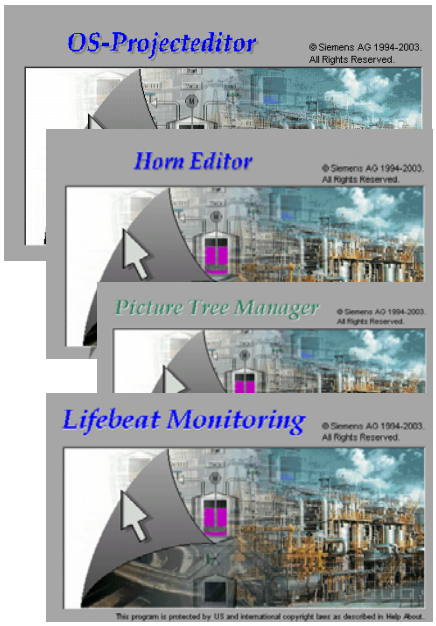
Note

You will find the currently offered options on the Internet http://www.siemens.com/wincc.

Basic Process Control 1/2

- OS-Projecteditor
- Group Display
- Picture Tree Manager
- Time Synchronization
- Horn Editor
- Lifebeat Monitoring
- Graphic Object Update Wizard*
- Block List Editor*

- Chipcard (Option)



As of Version 6.0, there is no longer a license check for the Basic Process Control package. * Function can only be used in PCS7.

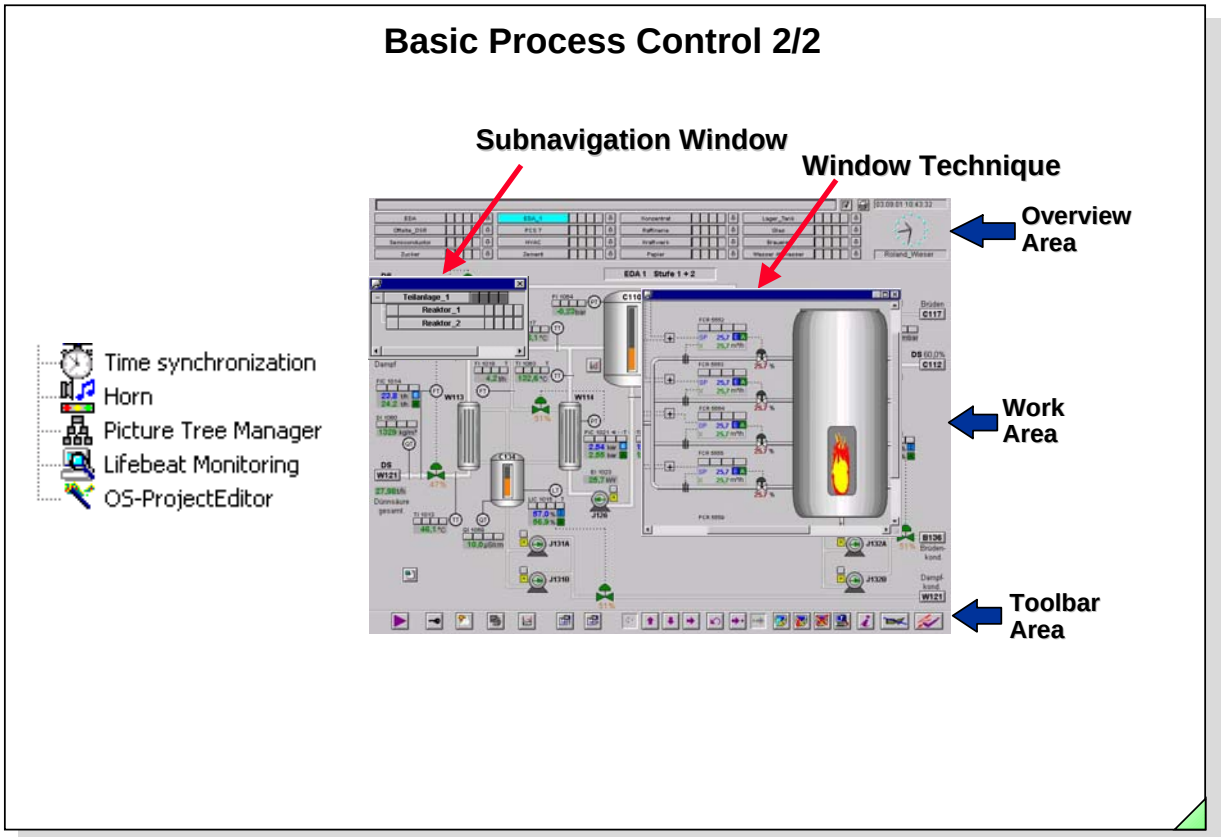
General Information The "Basic Process Control" package comprises, among others, the components:

OS-Projecteditor This Wizard is used to configure and initialize the monitor settings and screen settings for the current WinCC project.
You should do this initialization immediately after you create a project, since other applications (Runtime, Group Display etc.) also access this data.
The Wizard defines the newly created picture (@Screen.PDL) as the Start screen.
As well, there is a simple way of configuring the message system with fixed default settings (configuration of the message window, messages, message classes and creation of the process control messages).

Picture Tree Manager The Picture Tree Manager is used to manage a hierarchy of application pictures, parts of application pictures, function names and pictures from the Graphics Designer.

Group Display The Group Display is an additional element of the Graphics Editor (Smart Objects) that is used to collect messages and events in individual pictures and / or in the picture hierarchy and to display them in the application (overview) picture.
With the help of the Group Display, a malfunction or an alarm can be displayed in the overview picture. This allows the operator to switch directly to the malfunctioned system section by simply selecting it.

Lifebeat Monitoring The Lifebeat Monitoring takes over the permanent monitoring of the individual systems (OS and AS) and generates the process control messages.
After startup, the monitoring begins automatically for the configured (selected) components and is executed cyclically. A message is generated when one of the participants repeatedly fails to answer (after three successive cycles).



SIMATIC HMI

Date: 27.10.2003
File: swincc01e.7

SITRAIN Training for Automation and Drives

Basic Process Control Functions

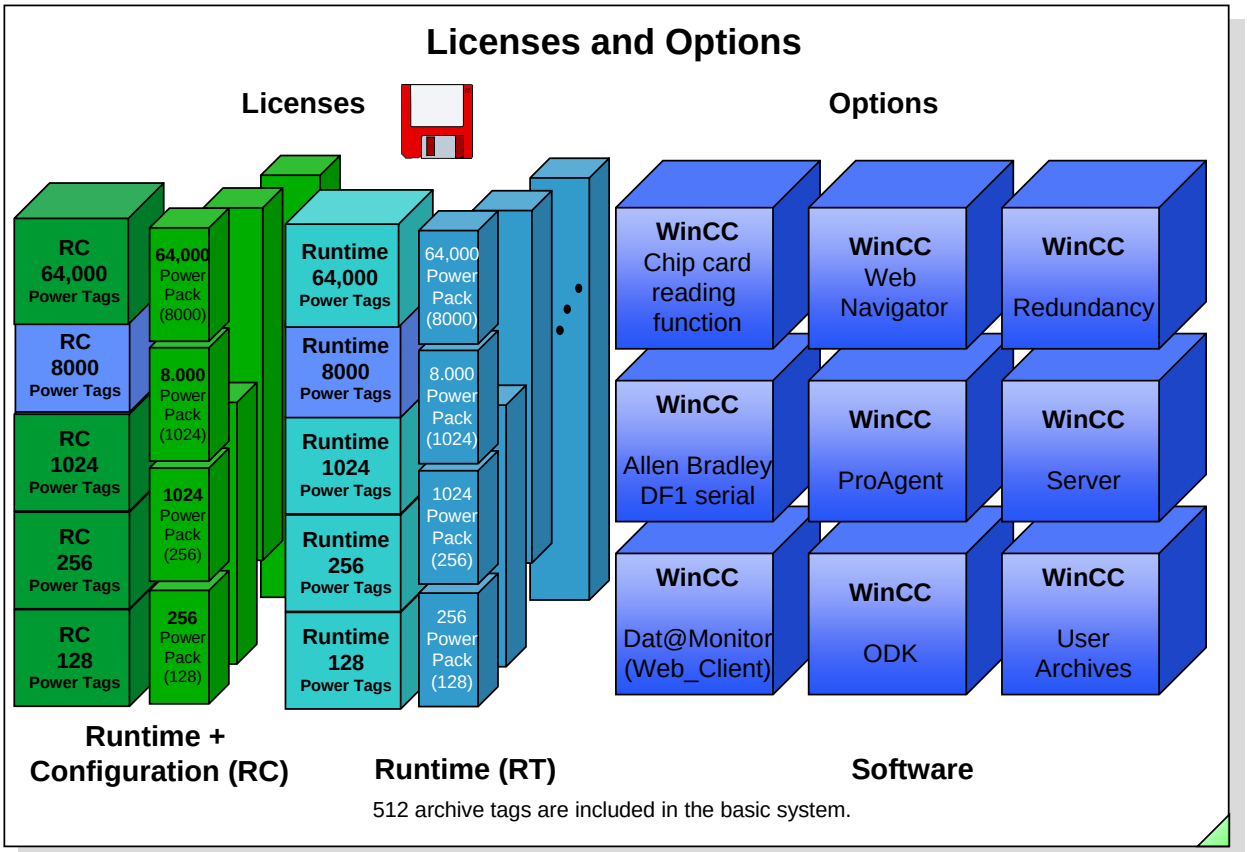
- Basic data with screen subdivisions of overview, desktop and button areas
- Process mimic hierarchy with graphic configuration of the picture hierarchy. (Picture Tree Manager).
- Scrolling through the picture hierarchy in process control.
- Storing and recalling the screen layout.
- Group displays for user guidance through the graphics hierarchy
- Selecting process mimics using names (in process control).
- Lifeboat monitoring with system configuration diagram.
- Message Configuration provides you with a simple way of configuring the message system with fixed default settings (configuration of the message window, messages, message classes and creation of the process control messages).
- Using external signalling device.
- Enhanced graphics objects: 3D Bar graphs,Group Display.
- OS-Projecteditor for adjusting screen resolution and multi-channnel operation with Multi-VGA cards and up to four monitors.

Time Synchronization With the time synchronization function, it is possible to synchronize all the clocks in the system using special clock masters such as Siclock or DCF77.

Horn Editor For use of a signal module or sound card. The signal module has relay outputs for a horn (acoustic) or signal lamp (optic).

Note You will require the Multi-VGA card from IGS for multi-channel operation. For activating (using) the signalling device, you will also require the signal module.

Chipcard Reader With the chipcard reader option, a WinCC User can log on with a chipcard using a reading device that is connected to a serial interface.



Date: 27.10.2003
File: swincc01e.8

Basic Software

The "WinCC" system software (core blocks with the basic functions), is scaled through various licenses.

Licenses

- 128 Power Tags for Complete Package (CS) or Runtime Software(RT).
- 256 Power Tags for Complete Package (CS) or Runtime Software(RT).
- 1024 Power Tags for Complete Package (CS) or Runtime Software(RT).
- 8000 Power Tags for Complete Package (CS) or Runtime Software(RT).
- 64000 Power Tags for Complete Package (CS) or Runtime Software(RT).

Archive Tags

512 archive tags are included in the basic system, others are optional.

Power Pack

The number of tags is upgraded with the power packs.
The cost is the price difference between the respective WinCC software packages for Configuration and Runtime software.

Options

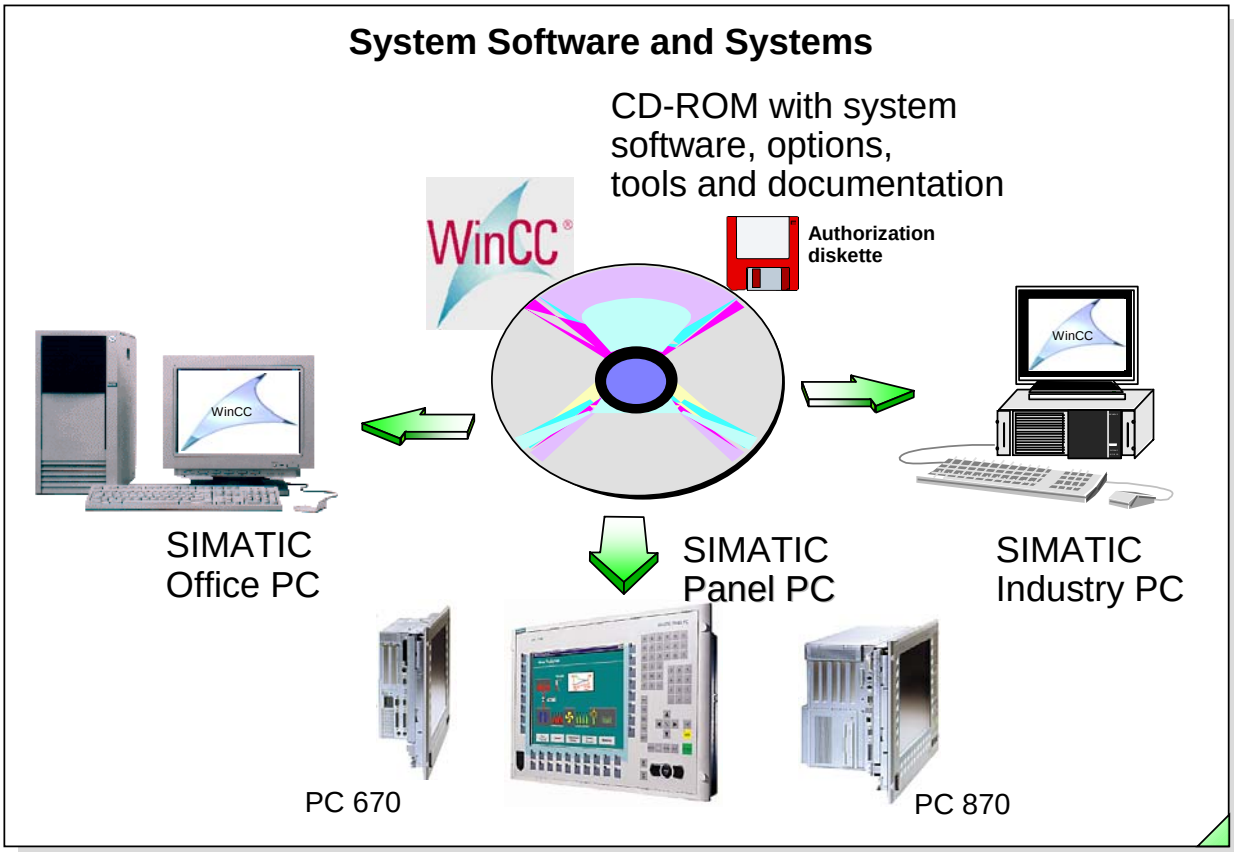
The basic software can be expanded through options.
- User archives (User Archives for batch recipes, for example).
- Additional PLC drivers.
- Server - Client (Multi-user system)
- ODK (Open Development Kit) for C programming.
- etc.

Add-ons

The basic software can be expanded through add-ons.
- Customer-specific or technological extensions that use the WinCC Basic software. For example, PM-MAINT maintenance program, PM-Aqua
- etc..

Note

You can find the currently offered add-ons and options on the Internet http://www.siemens.com/wincc.



Software Package Form of Delivery

The WinCC system software is delivered on a CD ROM with an accompanying Key Disk. WinCC runs under WindowsXP Professional and Windows 2000.

Systems Form of Delivery

In addition to being sold as a software package, the WinCC system software is also sold with current Siemens‘ PCs.
The PCs can be used for various requirements.

WinCC CD-ROM

The system software for Runtime systems, Complete systems, and selected options are found on the CD-ROM. In addition, tools, demonstration examples, and documentation are available.

Key Disk

The usage rights for each software package and expansion are stored on a red key disk.

SIMATIC PC and Panel PC

SIMATIC PC and Panel PC differ in their different hardware platforms and interfaces. They are designed as powerful industrial PCs with a robust construction and have a high degree of protection factor IP65. Because of their small depth, the PCs can be used anywhere in the plant, for example in an operator desk.

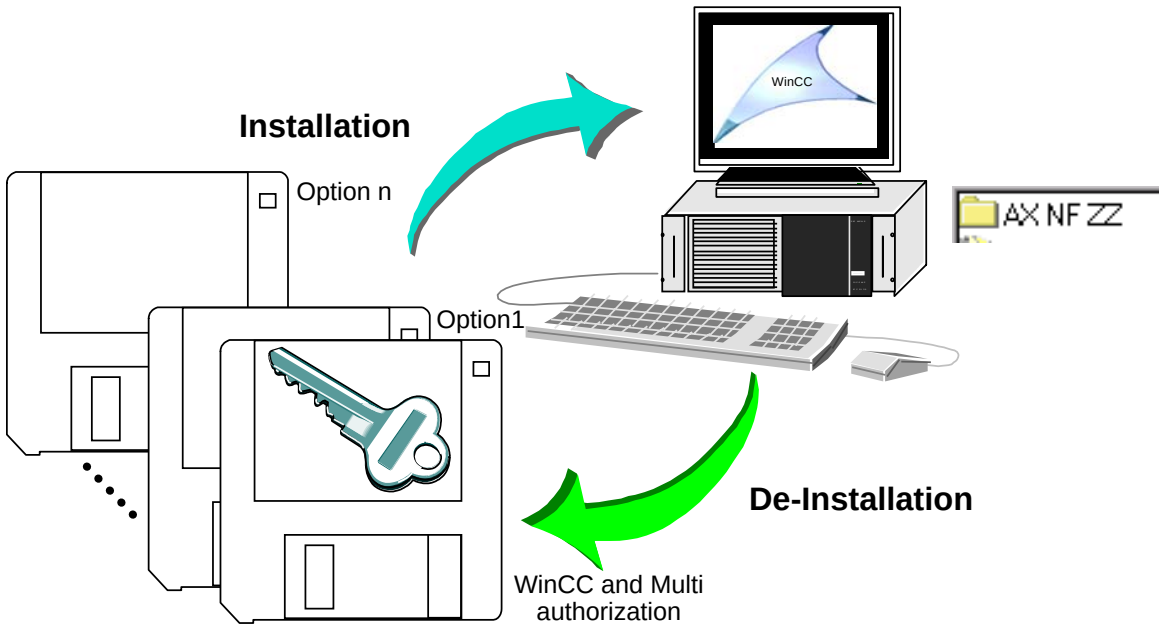
Industrial PC

Industrially hardened PCs are particularly appropriate for use in a harsh industrial environment, and they are resistant to aggressive environmental influences.

Note

You can find the currently offered products and systems on the Internet http://www.siemens.com/wincc.

WinCC Software Protection



No DEFRAG can be performed after the licenses have been installed, since the licenses could thereby be lost. Each WinCC Version has its own authorization.

Key Disk A key disk is provided to enable the WinCC Software.

Emergency License In addition to the normal license, there is also an emergency license on a seperate key diskette. The emergency license makes it possible to operate the software for a time period of 14 days. This allows you sufficient time for obtaining a new normal license.

Installation During the transfer of usage rights from the key disk to the PC, the installation counter on the key disk is decremented.
If all usage rights are assigned, it is no longer possible to enable additional software with this same key disk.

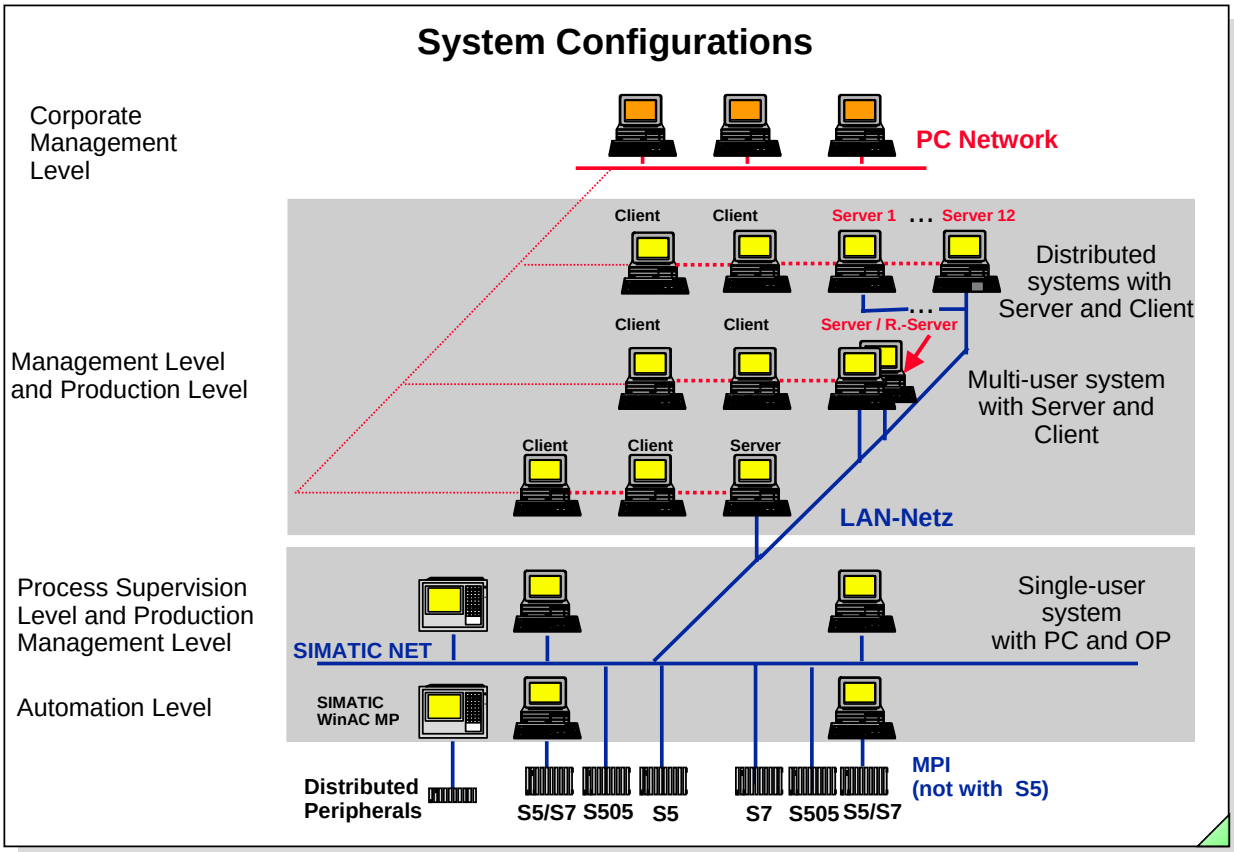
Note No DEFRAG can be performed after the licenses have been installed, since the licenses could thereby be lost.

De-Installation By transferring the usage rights from the PC back to the key disk, the installation counter on the key disk counts up (increments). This makes it possible to enable the WinCC software on another PC during a renewed transfer. The Key Disk of the WinCC software is also used as a Multi Key Disk.

Multi Key Disk The Multi Key Disk (yellow diskette) can store all WinCC licenses.

Virus Scan Do not perform Virus Tests (scans) while in running operation and only activate the process coupling after a hardware reset.

Note Without software protection, you can configure for one hour in the Demo mode. After that, the WinCC Explorer and the Editors are shutdown. The system then prompts you every 10 minutes to acquire a valid license.
In RT, the Demo mode is 30 days and can not be reactivated after that.



SIMATIC HMI

Date: 27.10.2003
File: swincc01e.11

SITRAIN Training for Automation and Drives

Configurations

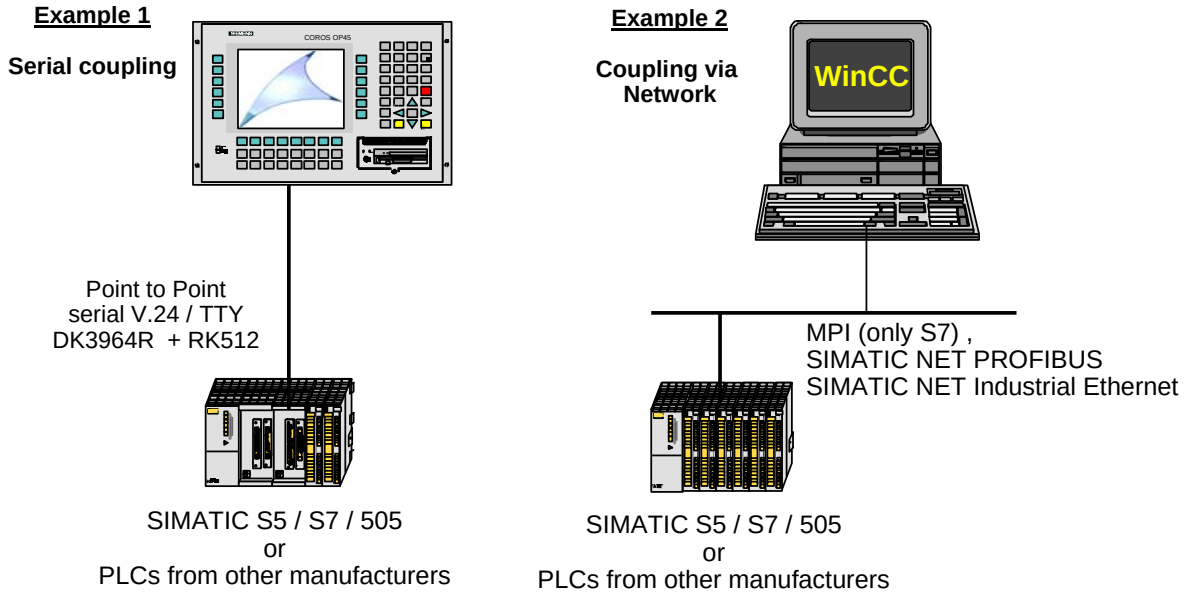
This slide illustrates the flexible configuration possibilities of WinCC**.**

- Extensive visualization tasks can be combined using multi-user systems.
- Distrubuted systems employ the technological distribution of a system or the functional distribution to several WinCC computers.
- A Server takes over process coupling.
- All other clients are connected to a server using a PC network.
- The clients receive all data from the server.
- Even higher-order computers of the management level can access the WinCC database at any time using the terminal bus.
- The SIMATIC Industrial Ethernet process bus can also be used as a terminal bus for industrial applications in a harsh environment.

WinAC MP

- Software PLC on a robust hardware platform (without fan and hard drive) under Windows CE, executable on the multifunctional platform.

Configurations for Single-user Systems



SIMATIC HMI

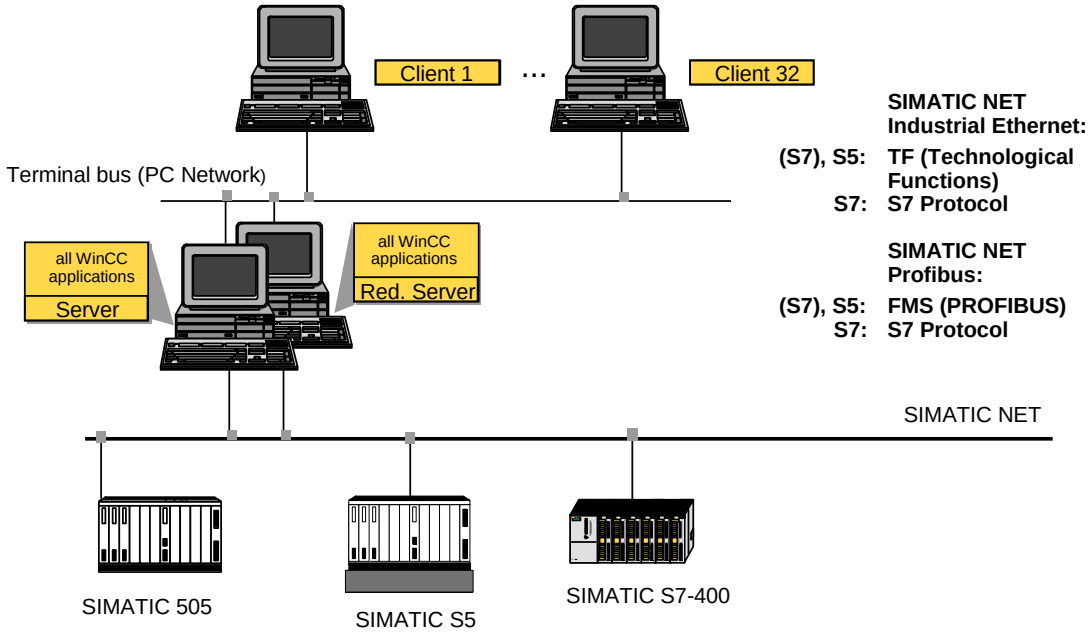
Date: 27.10.2003
File: swincc01e.12

SITRAIN Training for Automation and Drives

Configurations Single-user systems are used in small (number of tags) applications, but they can also be the operator interface for independent system components.

Stand-Alone A single-user system works as a stand-alone when it has everything it needs to operate. These point-to-point couplings and LAN couplings make up the automation level.

Multi-user System



Note: This WinCC configuration is also used for multi-configuration.

Multi-user System

The Server Client configuration makes it possible to have up to 32 operator stations linked to one Server. The configuration of the Server can be made by the Client.

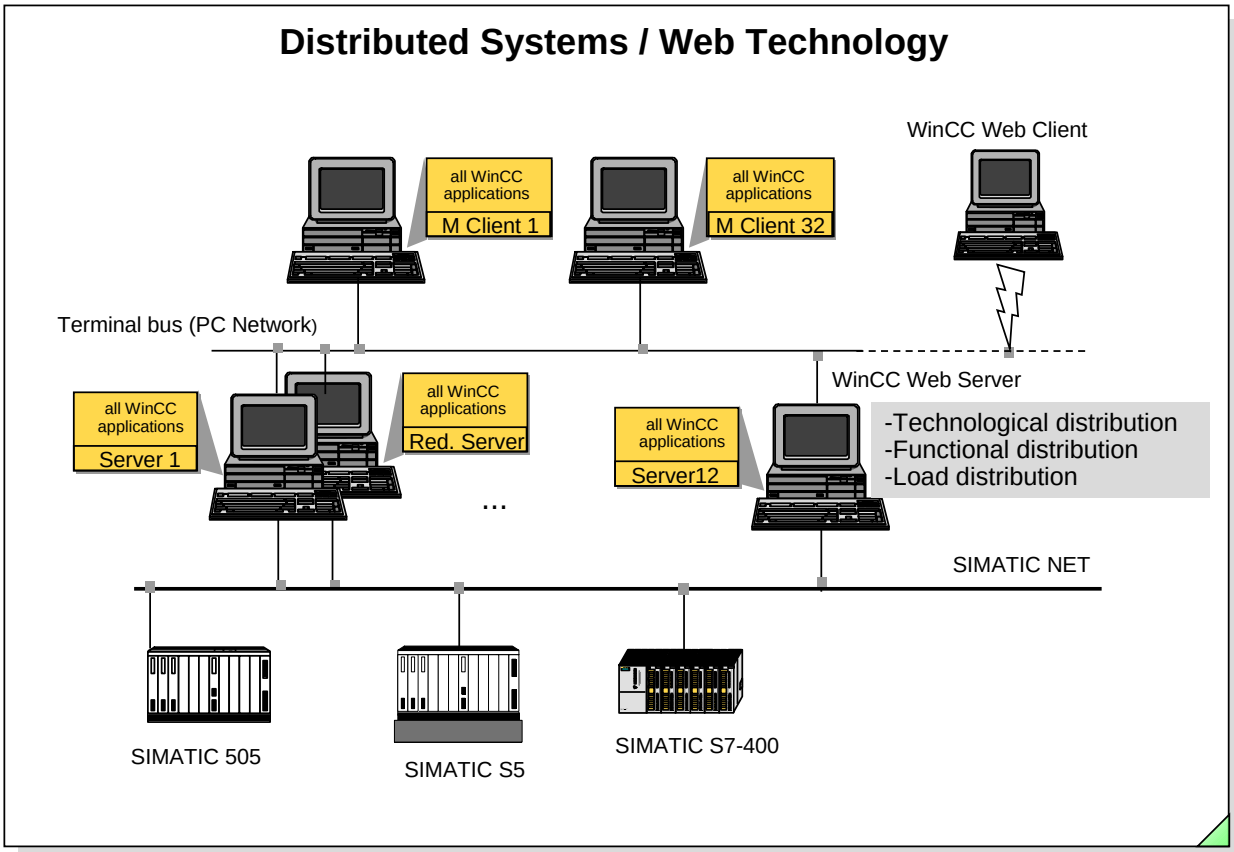
Redundancy Server

When you operate two WinCC stations in parallel, an archive adjustment is automatically made at runtime or after the failure of a Server.
The automatic switching of a Client to the active Server guarantees the continuous process execution and operation.

Note

Number of Clients for Server with operator station 4
Number of Clients for Server without operator station 32

For the Clients, the same operating system must be used. However, different operating systems can be used for Server and Clients.



SIMATIC HMI

Date: 27.10.2003
File: swincc01e.14

SITRAIN Training for Automation and Drives

Distributed Systems The Client can access up to 12 Server projects.

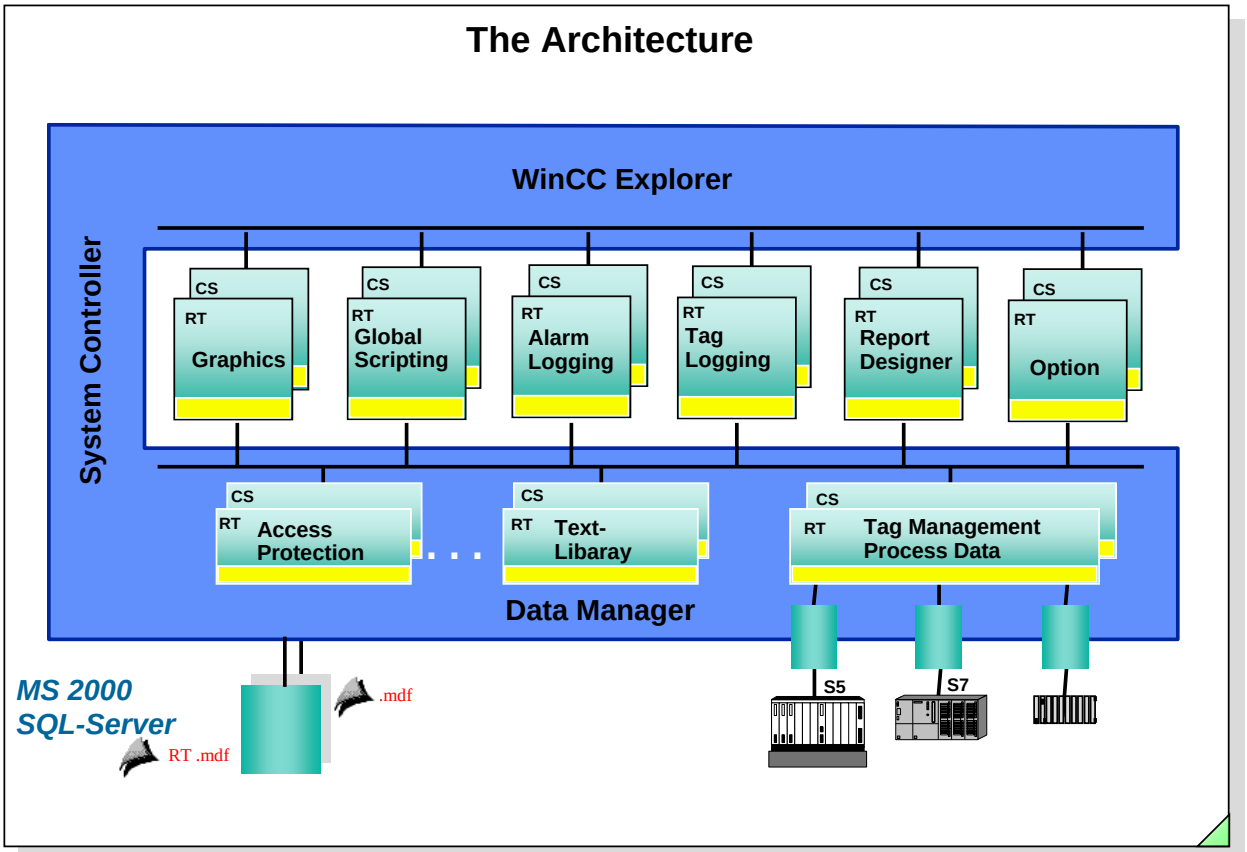
Redundancy Server When you operate two WinCC stations in parallel, an archive adjustment is automatically made at runtime or after the failure of a Server.
The automatic switching of a Client to the active Server guarantees the continuous process execution and operation.

Note
Number of Clients for Server with operator station 4
Number of Clients for Server without operator station 32
For the Clients, the same operating system must be used. However, different operating systems can be used for Server and Clients.

WinCC Web Navigator Process visualization using the Internet.
WinCC/Web Navigator enables you to operate and monitor your system using the Internet or company-internal Intranet and/or LAN.

Option
Web Navigator Server V6.0 (License for 3 / 10 / 25 / 50 Clients)
Web Navigator Client V6.0 (no license)

Thin Client (Web) Based on the terminal service of Windows 2000, Thin Client solutions are possible on different hardware platforms. This is true for clients positioned near the machine that are based on the realtime operating system Windows CE 3.0, as well as Linux and Unix clients.

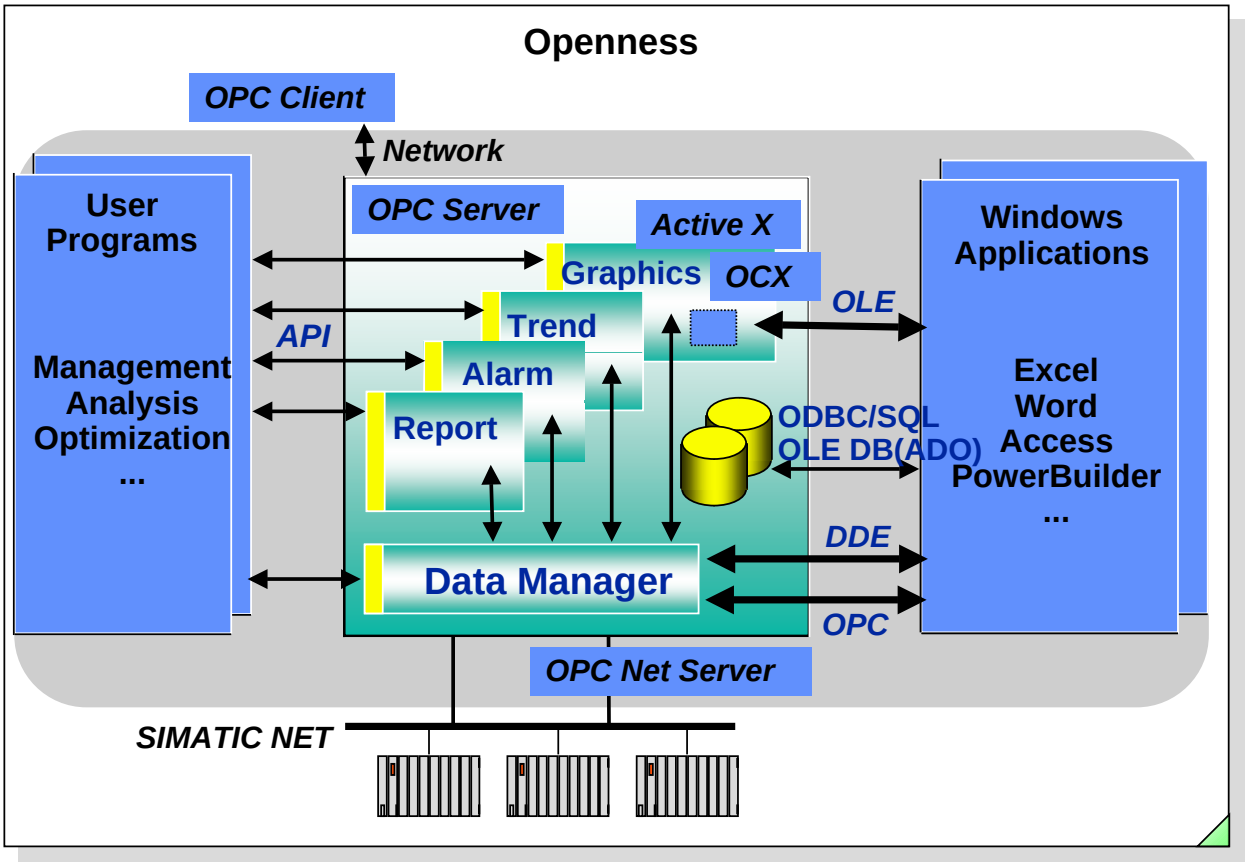


SIMATIC HMI

Date: 27.10.2003
File: swincc01e.15

SITRAIN Training for Automation and Drives

Architecture	This slide displays the relationships of the individual modules within the WinCC Explorer.
WinCC Explorer	The WinCC Explorer combines all the data necessary for operating a single-user or multi-user system and ensures hierarchically-ordered storage.
System Controller	The system controller takes care of system communication between the stations. For multi-user systems, the system controller also provides the network-wide project view, as well as communication between the client and server. The system controller also handles coordination through the terminal bus.
Data Manager	The data manager is the start-up location for processing central tasks in a WinCC project. The data manager exists on every station and works closely together with the system controller. The main task is handling tag management. The communication channels are used to access process data.
Function Modules	-Graphics Designer (graphics system) Graphic display and process connection -Global scripts (script editing) Making the project dynamic to user-specific requirements -Alarm logging (message system) Displaying messages and acknowledgements -Tag logging (measured value archiving, processing and trending) Processing measured values and their long-term storage Processing user-oriented data and their long-term storage -Report designer (report system) Reporting system states
Note	The WinCC Explorer enables you to navigate through the WinCC applications and their data with few operations. From its appearance and operation, the WinCC Explorer is similar to the Explorer in Windows.
MS 2000 SQL Server	Industry-suitable Database Management System from Microsoft



Date: 27.10.2003
File: swincc01e.16

Openness

This slide illustrates all system modules with opened interfaces.
You can use your own programs with Windows applications, as well as data and functions from WinCC here.
User programs can also be started from WinCC.
Applications that run under Windows, such as Excel, Word, or Access can be combined freely with WinCC and individual user programs.
This also makes solutions to unusual management level tasks possible.

User Routines

User routines run interpretively in WinCC. That is, a compiler is not necessary. If an application requires many, extensive user routines with a high processing capacity, standard tools can compile these routines, and they can be attached to the WinCC sequence.

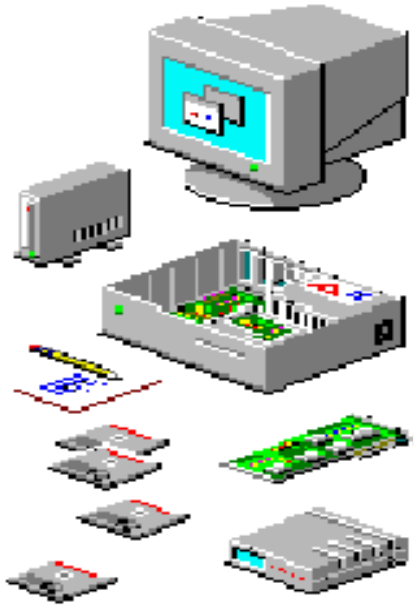
OPC

OPC (OLE for Process Control) integrated Client/Server communication interface.

WinCC supports

-OPC DA Server/Client
-OPC A&E Server (Alarm & Events)
-OPC HDA (Historial Data Access)

Hardware and Software Requirements



SIMATIC HMI

Date: 27.10.2003
File: swincc01e.17

SITRAIN Training for Automation and Drives

WinCC Hardware

-Recommended for Server, Pentium III, 1400 MHz, Main Memory 512 Mbyte.
-Recommended for Client, Pentium III, 800 MHz, Main Memory 256 Mbyte.

You will find further details on hardware in the WinCC Documentation *InstallNotes.pdf*

Software

-Windows 2000
-or Windows XP
-Servicepacks from MS
-Internet Explorer
-WinCC V6.0

You will find further details on hardware in the WinCC documentation *InstallNotes.pdf*

FAQ

You will find a complete list of WinCC and Windows compatibility under FAQ 11 39 080.

Installation

- **Hardware**
  - According to the Installation Guidelines (InstallNotes.pdf)

- **Software**
  - Microsoft operating system Win2000 or WinXP
  - Windows 2000 SP3 or Windows XP SP1 (included in delivery)
  - Microsoft Message Queuing services (from Windows Installation CD)
  - Internet Explorer V6.0 SP 1 (included in delivery)
  - Microsoft SQL Server 2000 SP3 (included in delivery)
  - WinCC V6.0
  - SIMATIC NET, V6.0 SP5 (November 2002) for Windows 2000 and Windows XP
  - Installation of the authorizations

SQL Server 2000

A license is necessary to be able to use the Microsoft SQL Server 2000 database. This exists when you install WinCC with a valid license.
The SQL Server licensed with the installation of WinCC may only be used in connection with WinCC. Use for other applications or your own databases requires an additional license.

Operating system

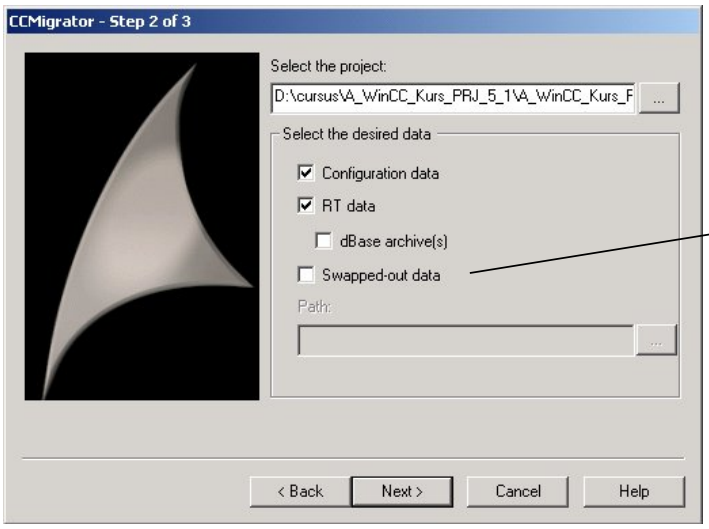
All WinCC Servers must be operated with Windows 2000 Server. All Clients of a project must either be operated only with Windows XP or only with Windows 2000.

Hinweis

You will find detailed information on installation in the WinCC documentation *InstallNotes.pdf .*

Migration of Configuration from V5 to V6 1/2

- **Requirements**
  - WinCC V5.0 SP2 or V5.1 Project
  - Sybase 7 installed
    - from existing WinCC V5 installation
    - from WinCC V5 CD (WinCC\setup\Products\Sybase_7)



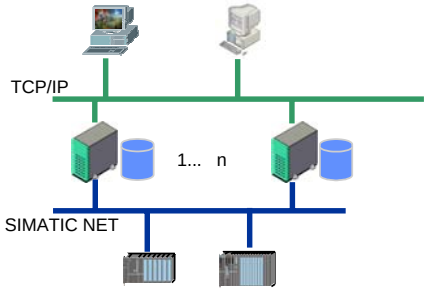
Tools: “Project Migrator“ Wizard

from Storage

- **Steps**
  - Parameterize Migration Tool for the migration of Configuration and Runtime databases (also Storage)
  - Convert Screens, Libraries, etc.

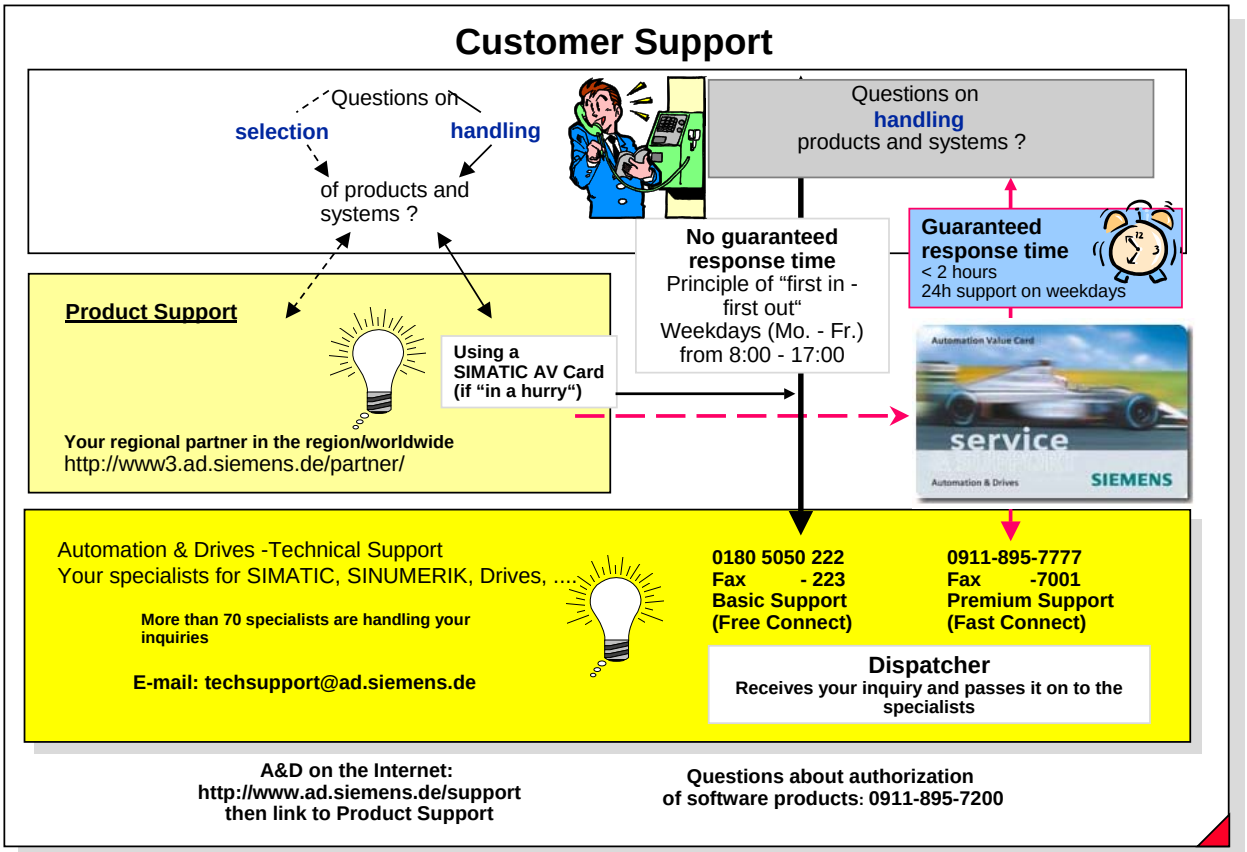
Note	After Migration, the Sybase Database has to be deinstalled.
Converting Sreens and Libraries	Conversion of the projects (screens and libraries) takes place in the WinCC Explorer using the Graphics Designer icon. Take a look at the current documentation from WinCC on “WinCC Information System“. A Migration Tool is only necessary as of >=WinCC V5.0 Servicepack 2.
WinCC Versions < V5.0 Service Pack 2	A project that was created with an earlier WinCC Version than V5.0 SP2 must be migrated step by step: -Upgrade your system to WinCC V5.1 and migrate the project. -Install WinCC V6.0 and migrate the project with the Project Migrator.

Migration of Configuration from V5 to V6 2/2



- **Standard Client/Server**
  - Migration Tool automatically converts the Server project including the computer list
  - Sybase Installation must still exist on the WinCC computer for conversion to MS SQL.

- **Multi-Client/Server**
  - Separate migration of the Server and ALL Client projects
  - Sybase must be present on ALL WinCC computers for the migration
  - Initial Export and Import (create new) of the Server packages is necessary – after that it runs automatically.



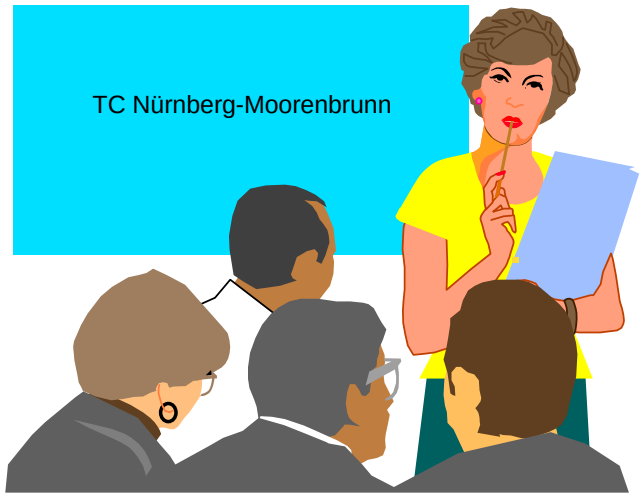
SIMATIC HMI

Date: 27.10.2003
File: swincc01e.21

SIMATIC AV CARD

You use the Automation Value Card to pay for the Technical Support's chargeable services. You can purchase the SIMATIC AV Card immediately online with a credit card (http://www.ad.siemens.de/support). With the SIMATIC AV CARD you can get additional information about the A&D products (such as downloading solutions from FAQs, purchasing the Knowlege Base-CD in the online shop).

Training Center Information



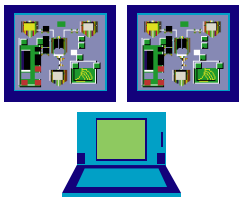
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_02e.1

SITRAIN Training for Automation and Drives

Basic and Advanced WinCC Courses 1/2

Project design/Programming/Commissioning engineers



Experience with graphic interfaces, such as Windows

SIMATIC WinCC
Operating and Monitoring, System Training
ST-BWINCCS 5 Days

SIMATIC WinCC
Advanced Course
ST-BWINCCV 5 Days

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_02e.2

SITRAIN Training for Automation and Drives

ST-BWINCCS

- WinCC system overview
-Create project, PLC coupling, tag simulation
-Using the Windows standard interfaces *
-Graphic screens
-Message display, message archiving
-Trend display, measured value archiving
-User archives
-Report system *
-Background processing (Global Scripts) *
-User openness API (benefits and structure) *
-Practical exercises

Your contact partner: 01805/ 23 56 11
Internet http://www.siemens.com/SITRAIN

Basic and Advanced WinCC Courses 2/2

1)

SIMATIC WinCC
Operating and Monitoring, System Training
ST-BWINCCS 5 Days

1)

SIMATIC WinCC
Advanced Course
ST-BWINCCV 5 Days

1) Programming engineer, Project design engineer, Commissioning engineer possibly Decision makers Maintenance

. . .

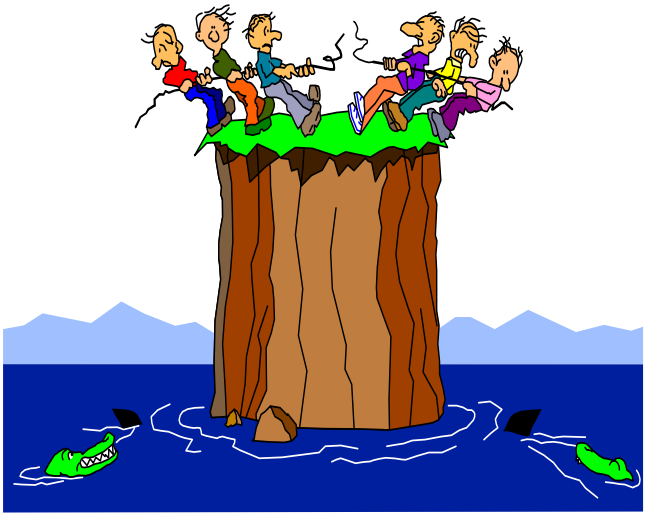
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_02e.3

SITRAIN Training for Automation and Drives

You will find the current contents on the ST-BWINCCV course in the SITRAIN catalog or on the Internet under http://www.Siemens.com/sitrain.

Creating a Project, Coupling a PLC, and Tag Simulation 1



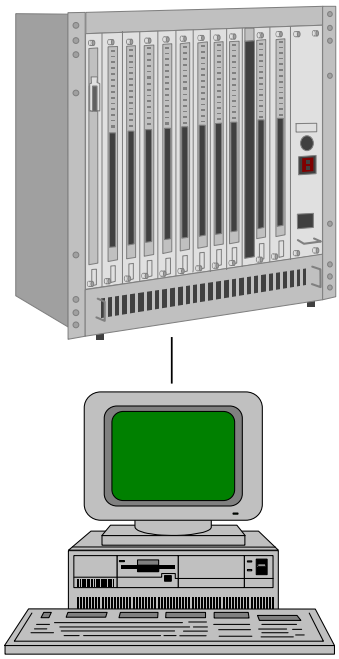
SIMATIC HMI

Date: 27.10.2003
File: a_dSwincc03e.1

SITRAIN Training for Automation and Drives

Contents Page

Creating a Project, Coupling a PLC, Variable Simulation 2



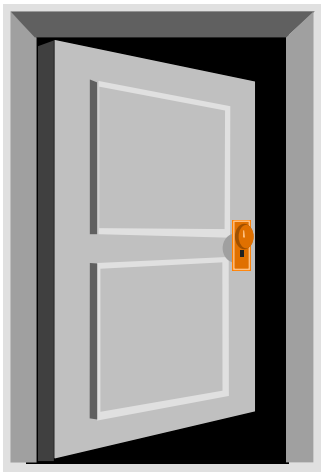
SIMATIC HMI

Date: 27.10.2003
File: a_dSwincc03e.2

SITRAIN Training for Automation and Drives

Contents Page

Objectives

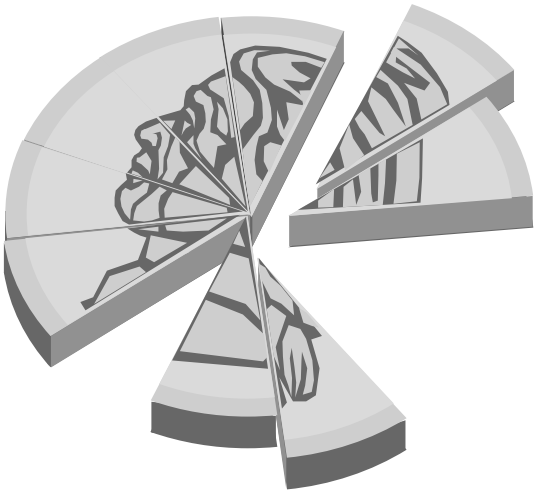


- Creating a Project
- Installing Communication Drivers
- Configuring Tags
- Tag Simulation
- Cross Reference

⇨ **Upon completion of this chapter, the participant should be able to:**

- recognize configuration relationships
- start WinCC and create a project
- describe the start procedure for WinCC
- work with the WinCC Explorer
- use the function modules for WinCC
- describe a data exchange
- use the computer features
- install the communication driver for his connection
- carry out the necessary parameter settings for the communication connection
- configure tags and tag groups
- install the auxiliary (help) program for tag simulation
- operate the auxiliary program for tag simulation
- use the result of the Cross Reference List

Configuration Preparation



- Programmable Logic Controllers
- Type of Coupling
- Process Tags

SIMATIC HMI

Date: 27.10.2003
File: a_dSwincc03e.4

SITRAIN Training for Automation and Drives

⇨ Configuration Preparation

❒ Process coupling

- Specifying the programmable logic controllers (PLCs) for the data exchange
- Specifying the physical level for the data transmission
- Specifying the number of input data and output data for PLC communication
- Specifying data areas for the individual send cycles

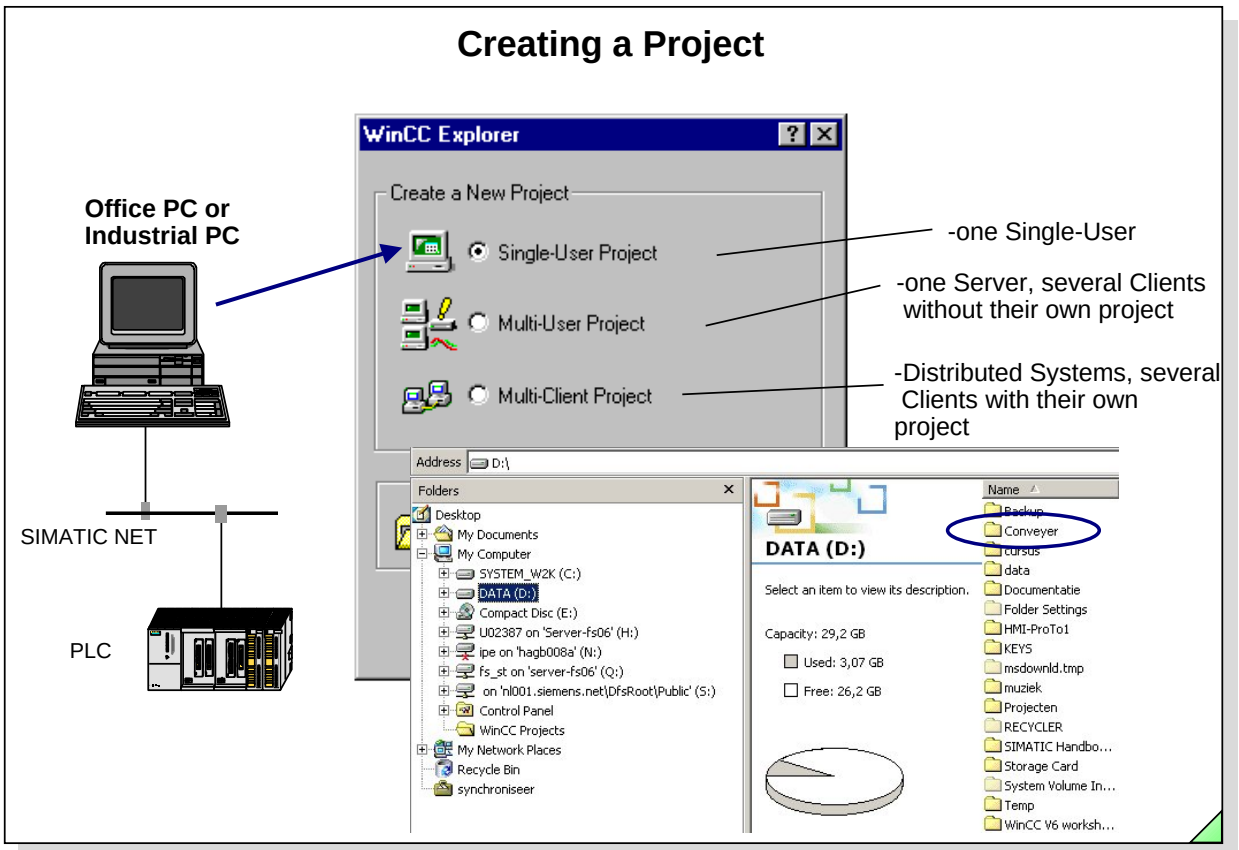
❒ Visualization

see Chapter 4.

❒ Password Protection

see Chapter 4.

Creating a Project



Starting WinCC from Windows

After you start WinCC, you are prompted to create a project.
WinCC independently creates a project folder for each project created (use ue_kurs, for example).
Then the MS SQL Server 2000 database is set up. After configuration is started again, the project last edited is called up.
If a new project is to be created, then select the following in the WinCC Explorer:

1. In the menu bar, click on "File".
2. In the drop-down menu, click on "New".
   The Explorer dialog box appears.
3. Select "Single User"
4. Click on "OK".
5. In the "Create New Project" dialog box, enter the project name.
6. Click on the "Create" button.

Selection Possibilities in the Start Dialog

The Explorer dialog box offers 3 configuration selections for creating a new project.

You can open an existing project in a further dialog box.

Use the Context Help.

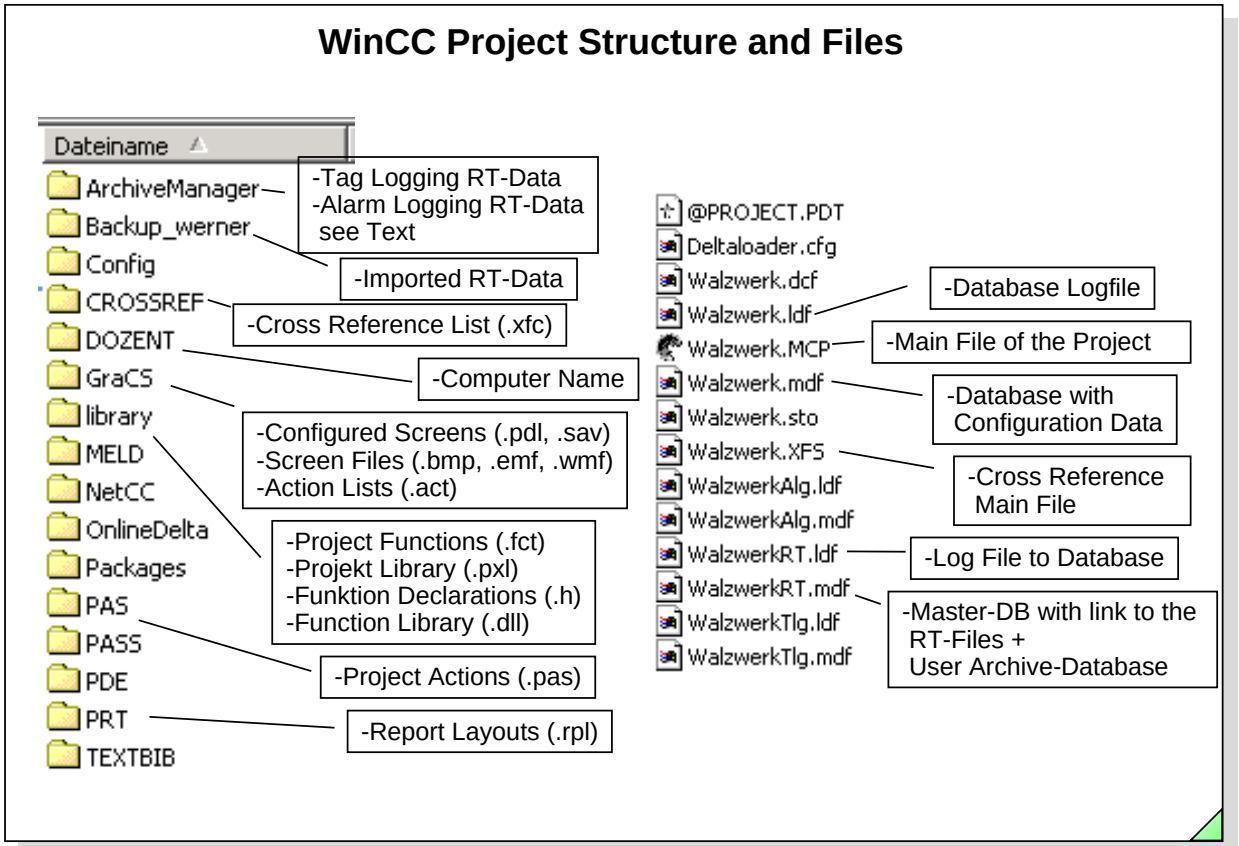
Note

You can use both lower case and upper case characters in a project name.

Characters Not Allowed

. , ; : ! ? " ' + = / \ @ * [ ] { } < > Blank

WinCC Project Structure and Files



SIMATIC HMI

Date: 27.10.2003
File: a_dSwincc03e.6

SITRAIN Training for Automation and Drives

Note The directory structure is dynamic and is dependent on the software tools installed and on the use of WinCC applications (on the WinCC CD).

ldf, mdf Database files always have two files.

ArchiveManager AlarmLogging File name structure:
ComputerName_ProjectName_ALG_FromTime_ToTime.mdf

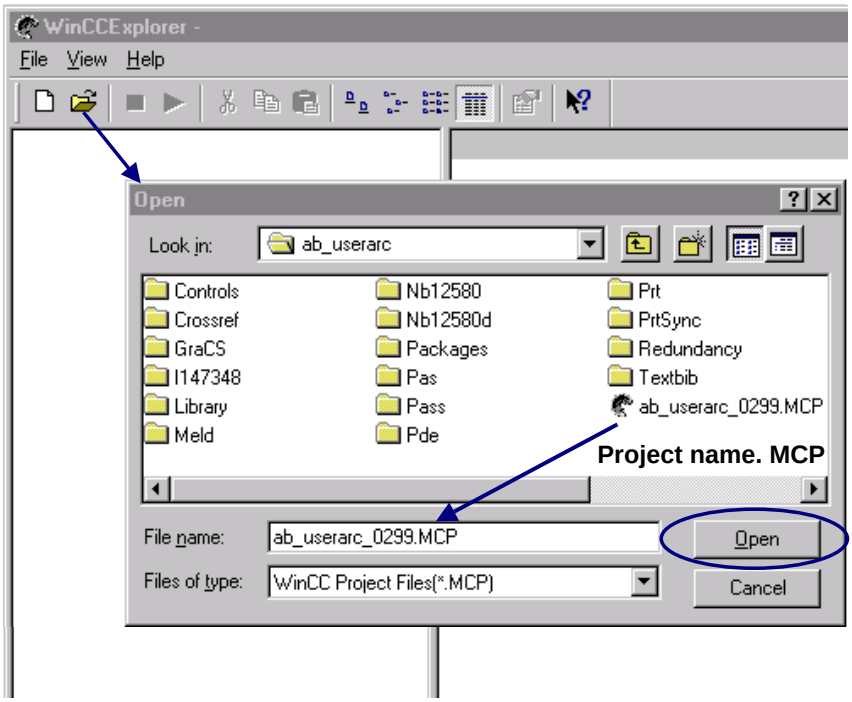
ArchiveManager TagLogging Fast File name structure:
ComputerName_ProjectName_TLG_F_FromTime_ToTime.mdf
Acquisition cycle <1min.

ArchiveManager TagLogging Slow File name structure:
ComputerName _ ProjectName _TLG_S_ FromTime_ToTime.mdf
Acquisition cycle >1min.

Empty RT-DB WinCC_SQL.mdf An empty RT-Database is located in the installation directory: Siemens\\WinCC\\bin\\ WinCC_SQL.mdf, that can be renamed after a copy is made. When an new project is created, the database is automatically generated.

***.Log File** Log Files *.log contain setup data, system status and error messages for the application engineer and service technician.
Installation directory: Siemens\\WinCC\\Diagnose.

Interchanging between Projects



SIMATIC HMI

Date: 27.10.2003
File: a_dSwincc03e.7

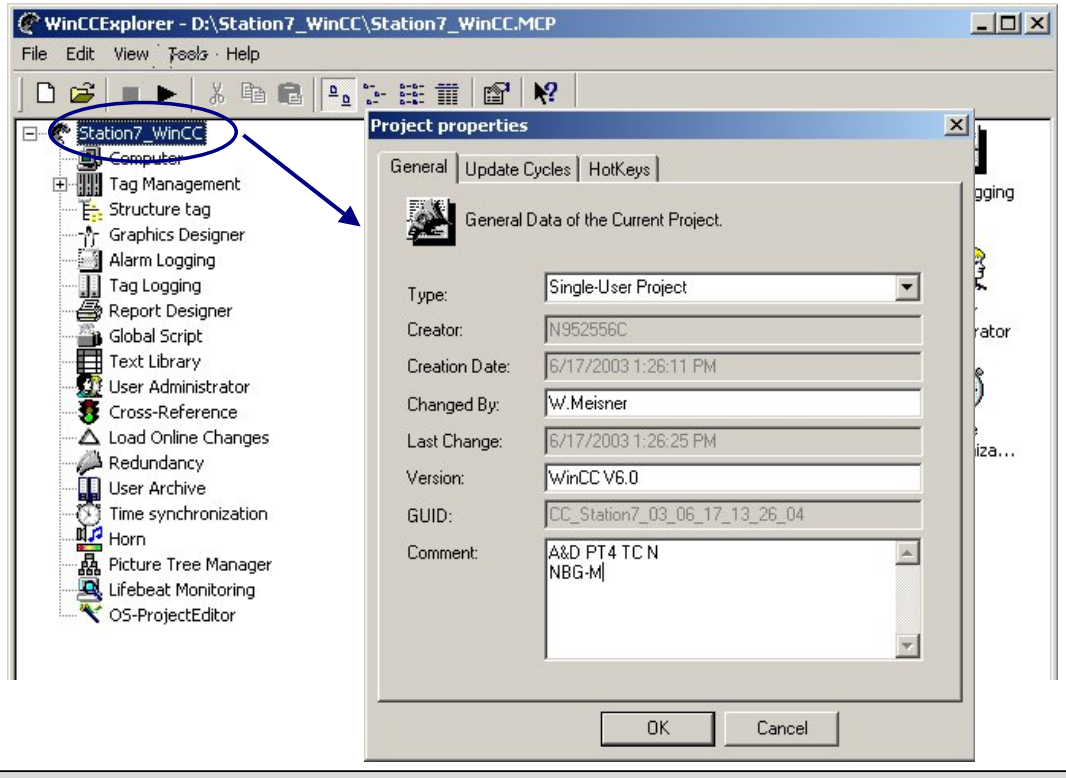
SITRAIN Training for Automation and Drives

Switching Projects

You can switch between existing projects:

1. In the WinCC Explorer menu bar, click on "File" .
2. In the drop-down menu, click on "Open".
3. In the "Open" window, search for your project.
4. Click on your file name, as illustrated in the slide above (File name: Beispiel.MCP).
5. Click on the "Open" button. You should now have switched to another project

Specifying Project Properties



Specifying Project Properties

By means of the shortcut menu for your project name (ue_kurs), you enter the creator, the version, and comments in the "General Information" tab of the Project properties dialog box that appears.

Update Cycle

By means of the "Update Cycles" tab, you can observe the 15 update cycles for the project. Of these update cycles, you can change five settings (update: acyclical, 250 ms to 1 hr).

Hot Keys

Hot keys are key combinations that trigger certain actions within WinCC, such as hardcopy, log on, and log off.

Configuration Structure

The connections and tags to the programmable logic controllers are specified for your own computer.

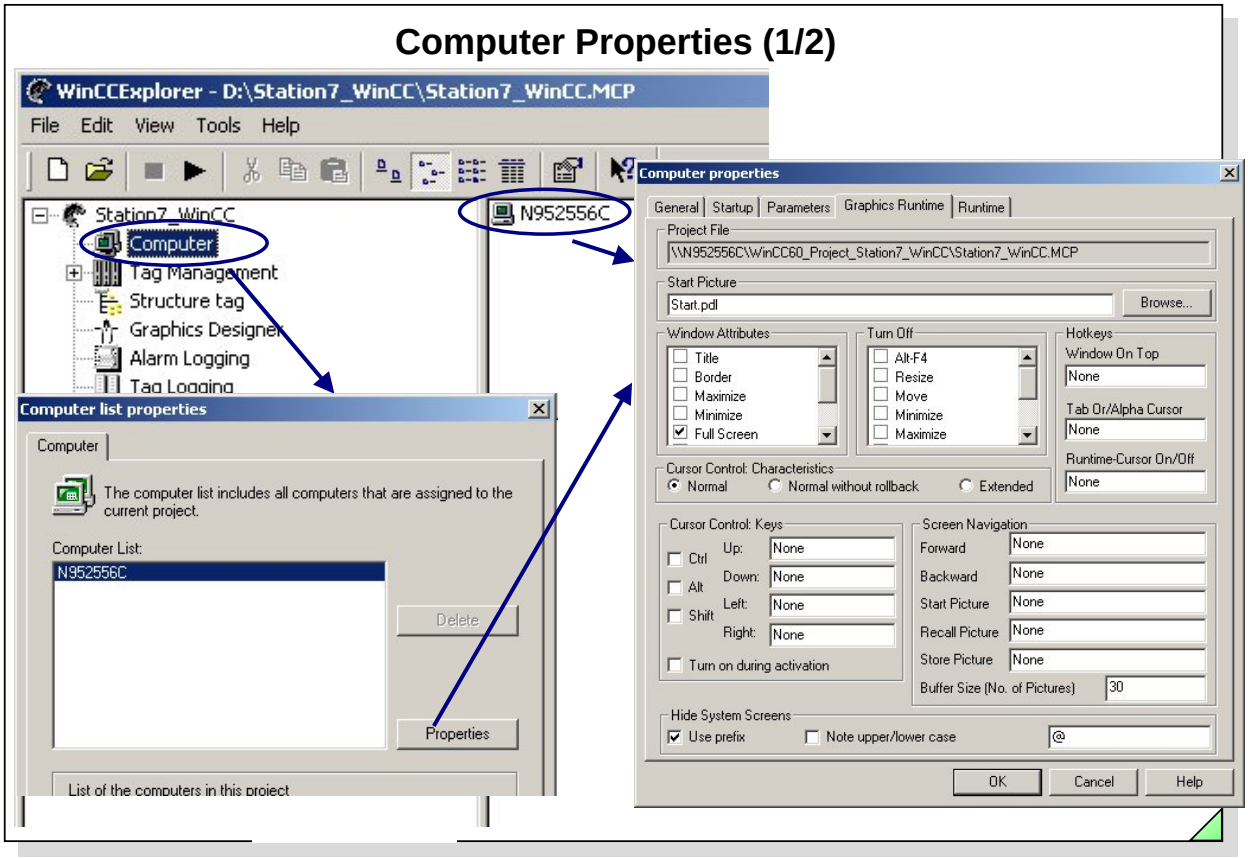
Find Function

The shortcut menu calls up the WinCC Find... function.
Use the right mouse button to click on the Computer icon or the Tag Management icon.

The following Find... functions are available:

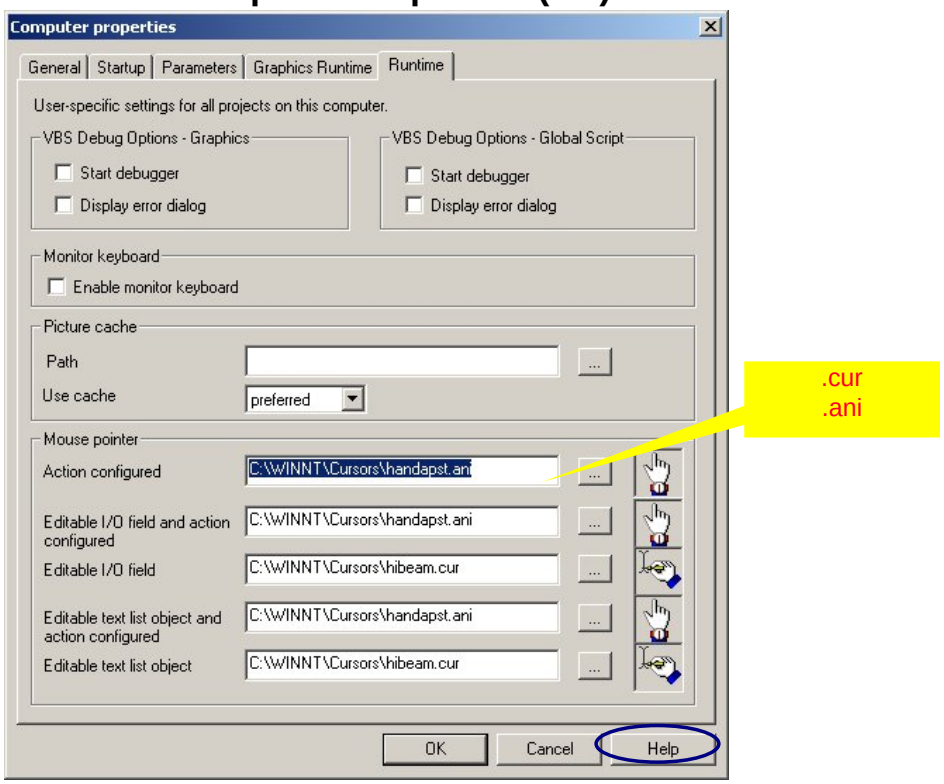
- Name
- Date of change
- Individual object types or all object types

Computer Properties (1/2)



Computer List	The computer list contains all computers assigned to the current project.
Call-up	1. Select the "Computer" icon in the navigation window. 2. Use the right mouse button to call the shortcut menu, and select "Properties". 3. To change the computer name, select "Properties" again. 4. Now enter a new computer name, such as STATIONx (x=lab station number), in the "General" tab.
General	The computer name, such as STATION1, and the computer type are specified in the "General" tab.
Startup	You use the "Startup" tab to specify standard WinCC programs to be loaded when the application is activated (Runtime is started). The selectable WinCC applications are: Graphics Runtime, Report Runtime, Tag Logging Runtime, Alarm Logging Runtime, Global Script Runtime, Text Library Runtime. You can start other (Windows) applications during startup by using the "Add" button.
Parameters	You use the "Parameters" tab to specify the language setting at runtime, and inhibit the Windows key combinations. The PLC Clock Setting selection allows you to display the RT time for WinCC data in the local time zone (PLC is set in the local winter time all year) or in the coordinated time zone -UTC Universal Time Coordinated (The PLC is set to the coordinated world time).
Graphics Runtime	You can make individual project settings (see slide above) in the Graphics Runtime tab. Select the Online Help.
Note	The computer name may not contain a blank, a hyphen or an underscore. The first character must be a letter. All characters must be capitalized. It is recommended that you only use the characters a to z, A to Z, 0 to 9.
Characters Not Allowed	. , ; : ! ? " ' ^ ´ ` ~ - + = / \ ¦ @ * # $ % & § ° ( ) [ ] { } < >

Computer Properties (2/2)



Runtime

The settings in this tab are valid (as of WinCC V6.0) for all WinCC projects on the computer used. The following settings are possible möglich:

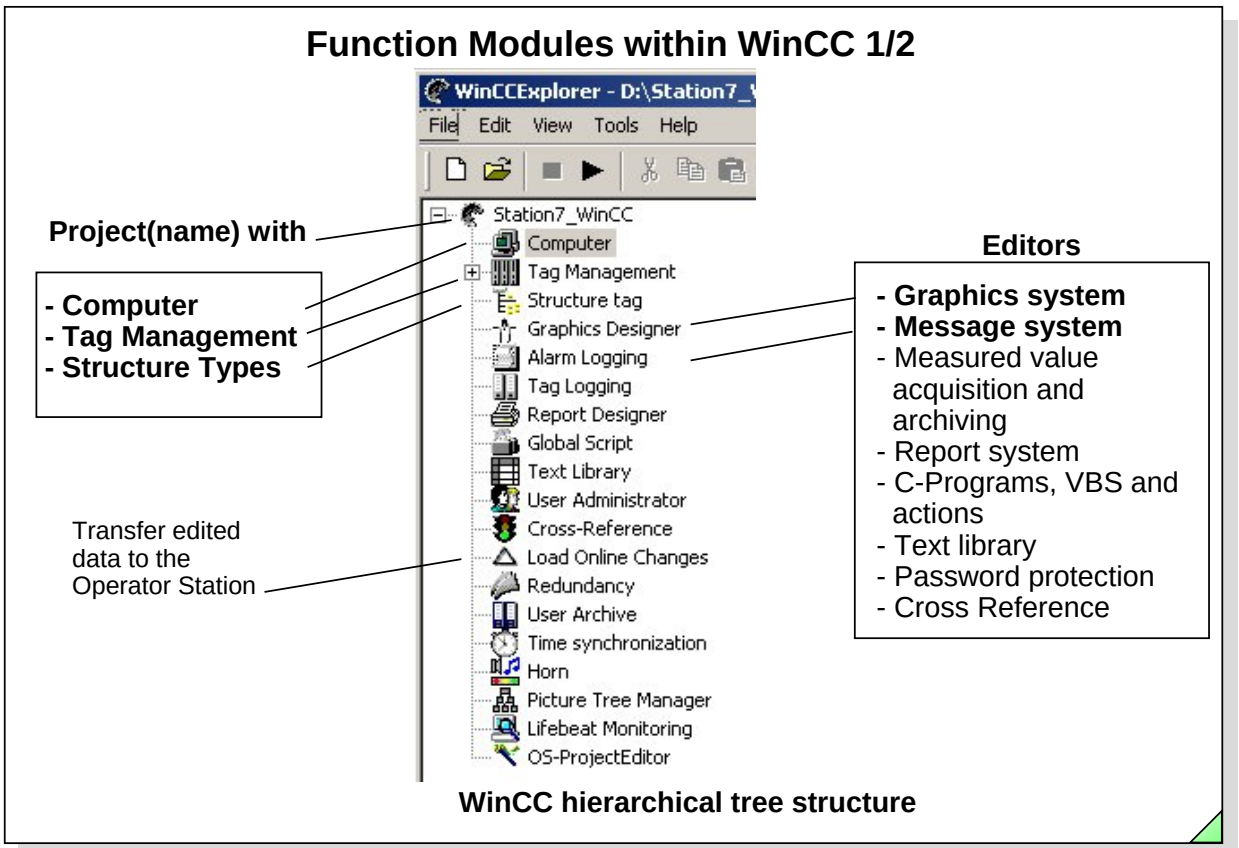
-Set Debug Options for Visual Basic Scripts in the Graphics Designer

-Set Debug Options for Visual Basic Scripts in Global Script

-Enable monitor keyboard

-Use picture cache

-change mouse pointer



WinCC Explorer After you have created a project, you see the project navigation window with four nodes as shown in the slide.

Computer You use the computer icon to define the computer properties of a workstation, such as the station name, start-up characteristics (program list) and parameters (language setting and key combinations).

Tag Management Tag management manages the drivers for coupling, logical connections, process and internal tags, as well as tag groups.

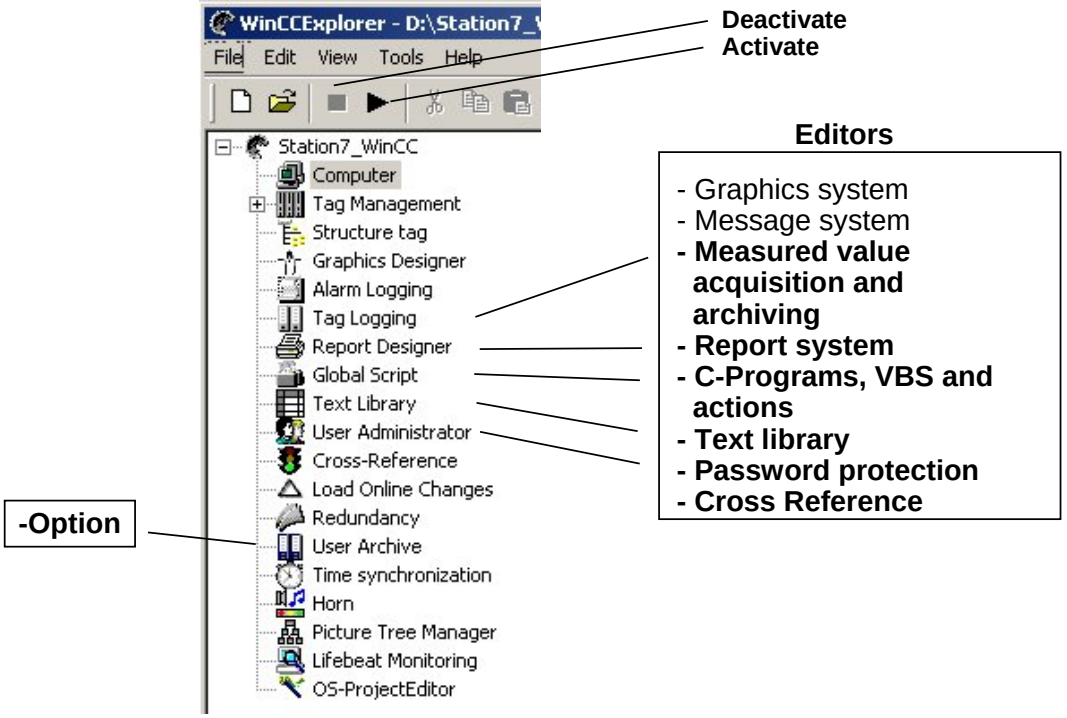
Structure Types You are able to generate a data structure with the help of this data type.

Editors

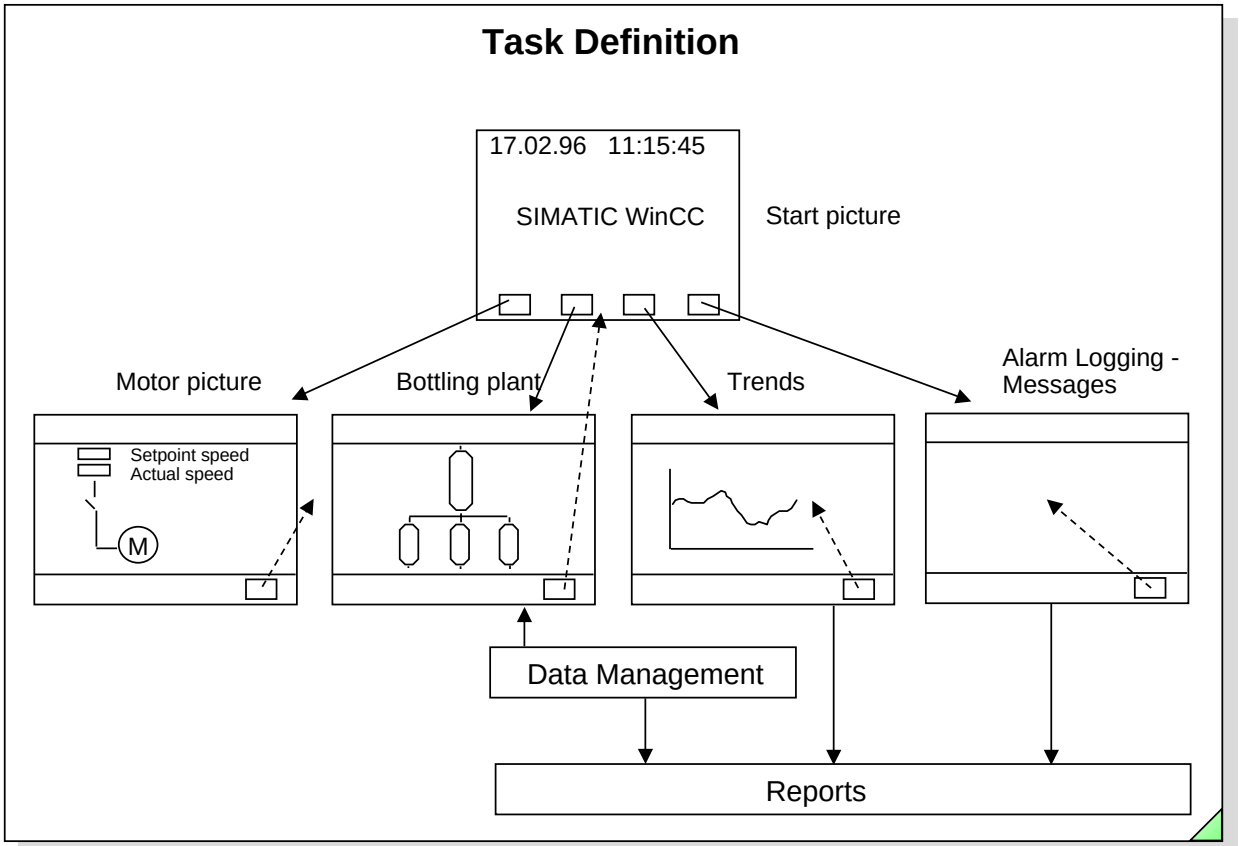
Graphics Designer Graphic Display and Process Connection:
User-friendly and simple user interface with tool palettes and graphic palettes.
Efficient configuration with an integrated object and symbol library.
Open interfaces for graphics import and support of the OLE 2.0 interface.
Configurable, dynamic performance characteristics of the screen objects with ANSI C, VBS and the support of an assistant (Dynamic Wizard).
Integration of your own functions and graphics objects that you develop yourself.
With Visual Basic for Application, you can carry out expansions for the Graphics Designer and automate procedures according to the COM Object Model. VBA represents an open solution for accessing every application according to the COM Object Model.

Alarm Logging Message system: Transfer of messages from processes for their preparation, display, acknowledgement, and archiving.

Function Modules within WinCC 2/2



Tag Logging	Archiving and measured value acquisition: processing of measured values and their long-term storage (can be used for trending).
Report Designer	Reporting: An integrated report system with which you can log user data, current and archived process values, current and archived messages, and your own system documentation, among other things.
Global Scripts	General term for C-functions, VBS functions, and actions: Making the project dynamic (animating the project) to user-specific requirements. This Editor makes it possible to create C-functions and VBS functions and actions that can be used for an entire project or even for more than one project, depending on the type.
Text Library	Text library: User texts, among other things, for the message system, multi-lingual
User Administrator	Password protection: Management of access rights for groups and users.
CrossReference	Cross reference list for indicating the application locations of various objecttypes, such as tags, screens and functions. It also contains the jump to Place of Use and Linking functions.
Option	Licensed software with which the WinCC basis system can be expanded.
Activate	When you activate a WinCC project, you switch to the Runtime operating mode. Runtime and Configuration mode can now be operated in parallel. Alternative: **In the menu bar click on File, in the drop-down menu, click on Activate.**
Deactivate	When you deactivate a WinCC project, you end the Runtime operation. Alternative: **In the menu bar click on File, in the drop-down menu, click on Activate.**



SIMATIC HMI

Date: 27.10.2003
File: a_dSwincc03e.13

SITRAIN Training for Automation and Drives

Task Definition

You are to configure the plant simulation diagrams depicting the start picture, motor picture, bottling plant picture, trends and message (alarm logging) picture. The configuration must then undergo a function test.

Start Picture

When you switch to runtime, the start picture (company logo) is to appear. In the start picture you can call the motor picture, the bottling plant picture, the trends picture and the message picture.

Motor Picture

Input and output of digital values, display change of the switch setting.

Bottling Plant

Display of digital values in graphic form (bar graph).

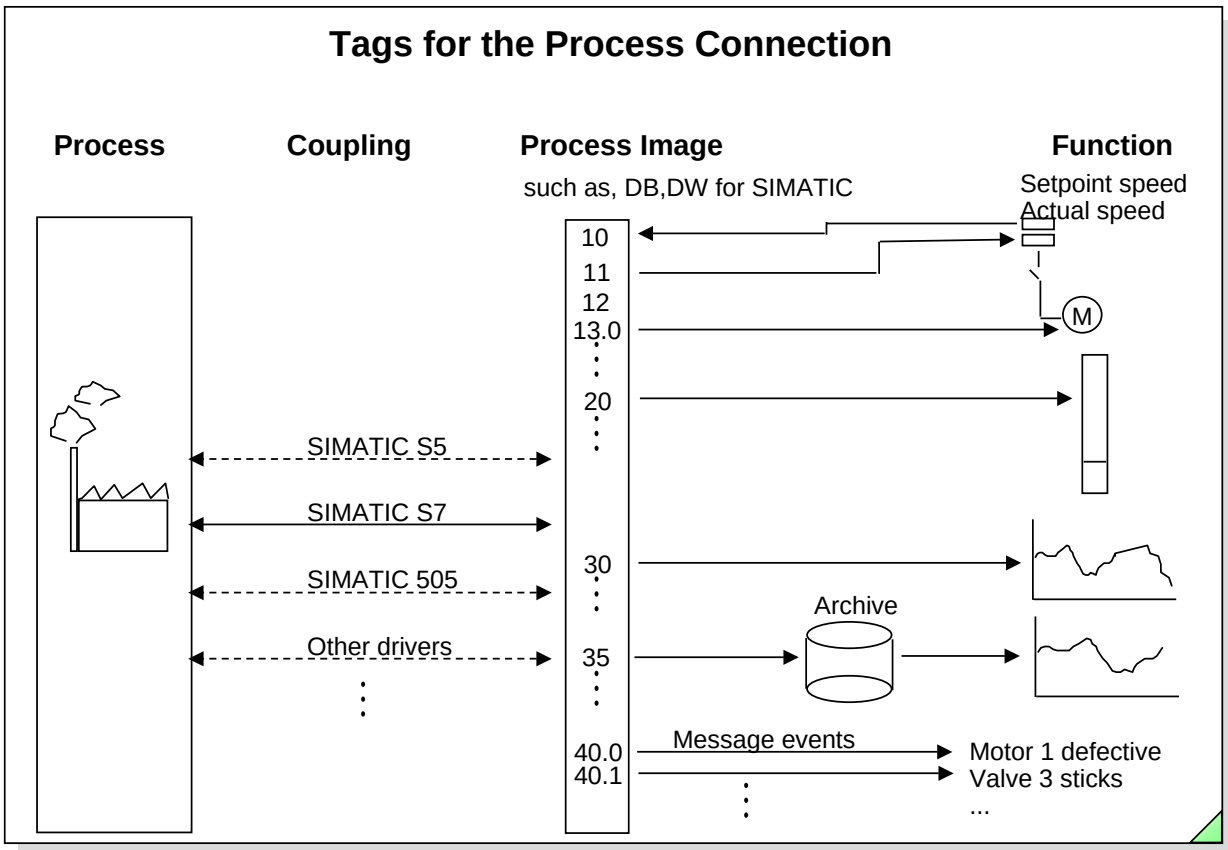
Trends

Display of digital values in graphic form (trends).

Messages

Display of error states and operating states.

The "Return" key must be used to select (return to) the start picture in all pictures that were selected from the start picture.



SIMATIC HMI

Date: 27.10.2003
File: a_dSwincc03e.14

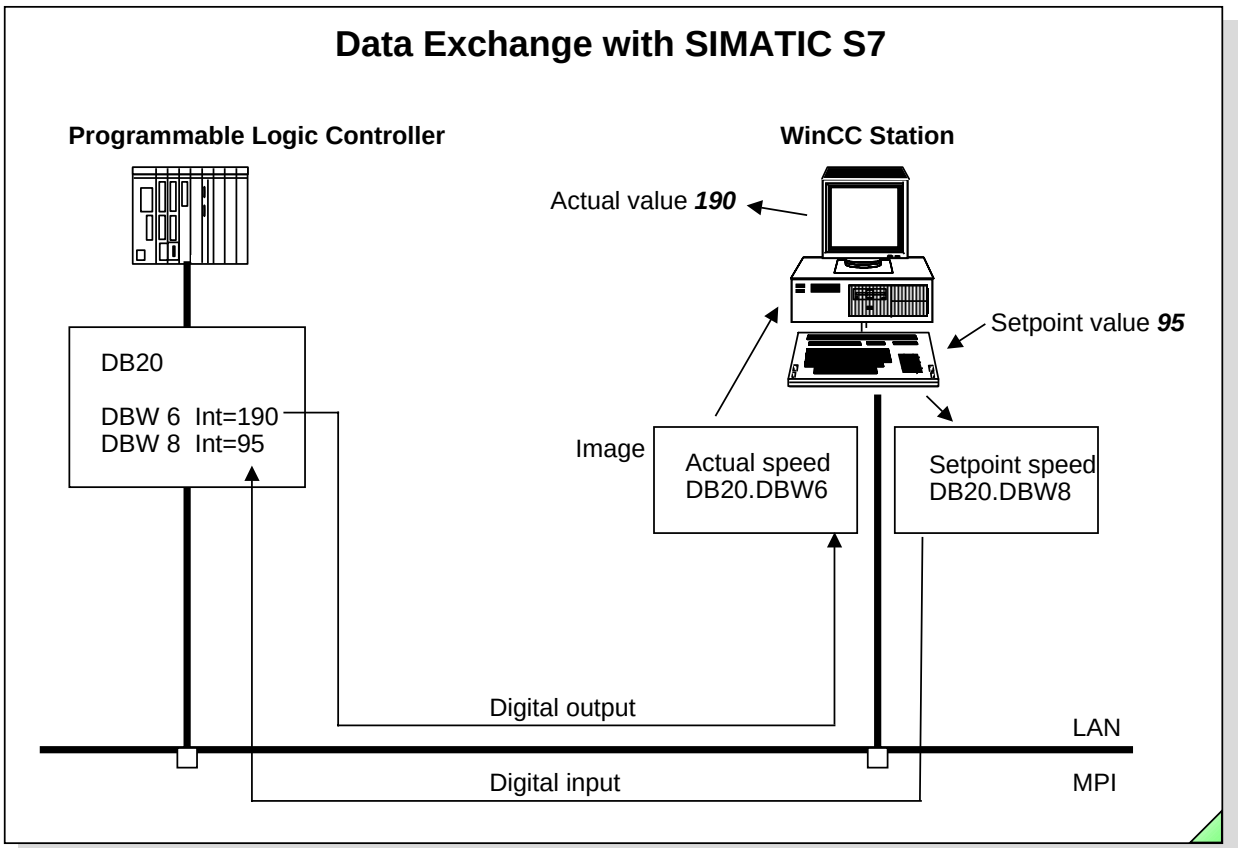
SITRAIN Training for Automation and Drives

Process For example, bottling plant, automatic assembly equipment etc.

Coupling The WinCC coupling can take place with SIMATIC drivers or other drivers.

Process Image The data manager manages the process image from the tag data from DBs, DXs, inputs, outputs, and bit memories (flags).

Function You use the editors to configure the view and control functions made available in WinCC and to connect the functions with the process by means of the tags.



Communication

SIMATIC S5
- serial 3964R
- serial AS511
- Ethernet Layer 4
- Ethernet TF
- etc.

SIMATIC S7
- S7 Functions with MPI
- S7 Functions with PROFIBUS
- S7 Functions with Ethernet
- etc.

SIMATIC 505
- serial interface

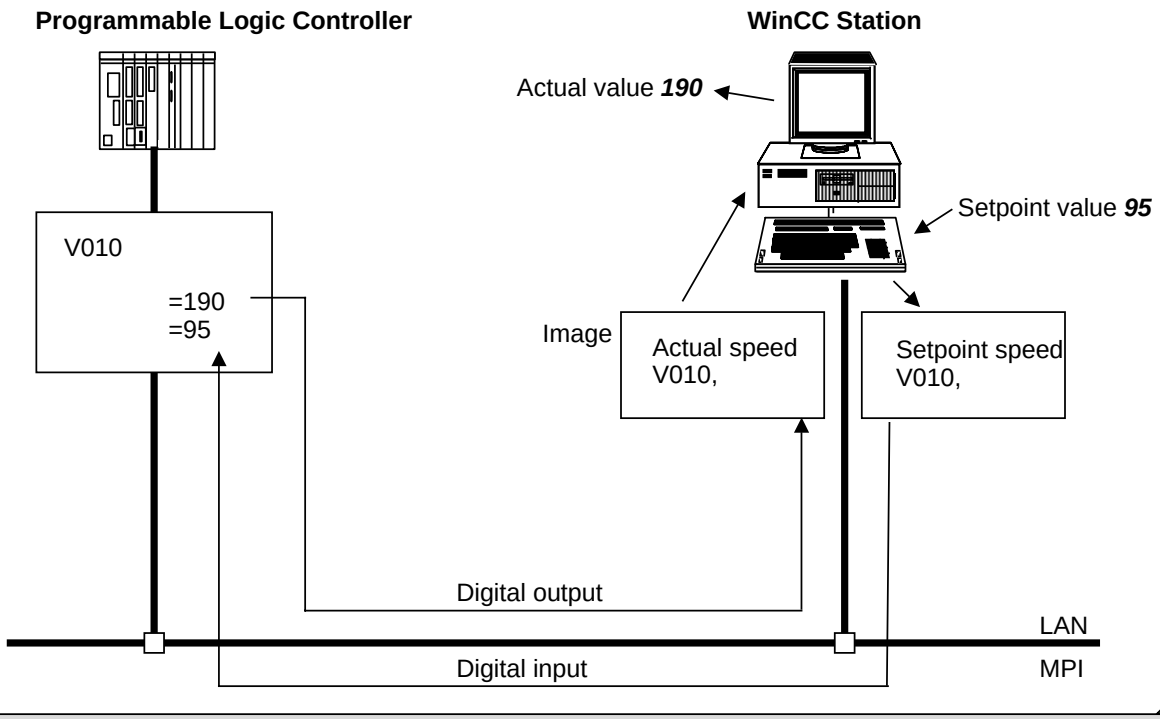
Other Couplings
- Allen Bradley Serial DF1
- Modbus Serial (Modicon)
- SIPART
- etc.

Supplier Independent
- OPC
- Windows DDE
- PROFIBUS FMS
- PROFIBUS DB

Note

You will find the currently available communication drivers on the Internet **http://www.siemens.com/wincc** -> WinCC Options -> WinCC Channels.

Data Exchange with SIMATIC 505

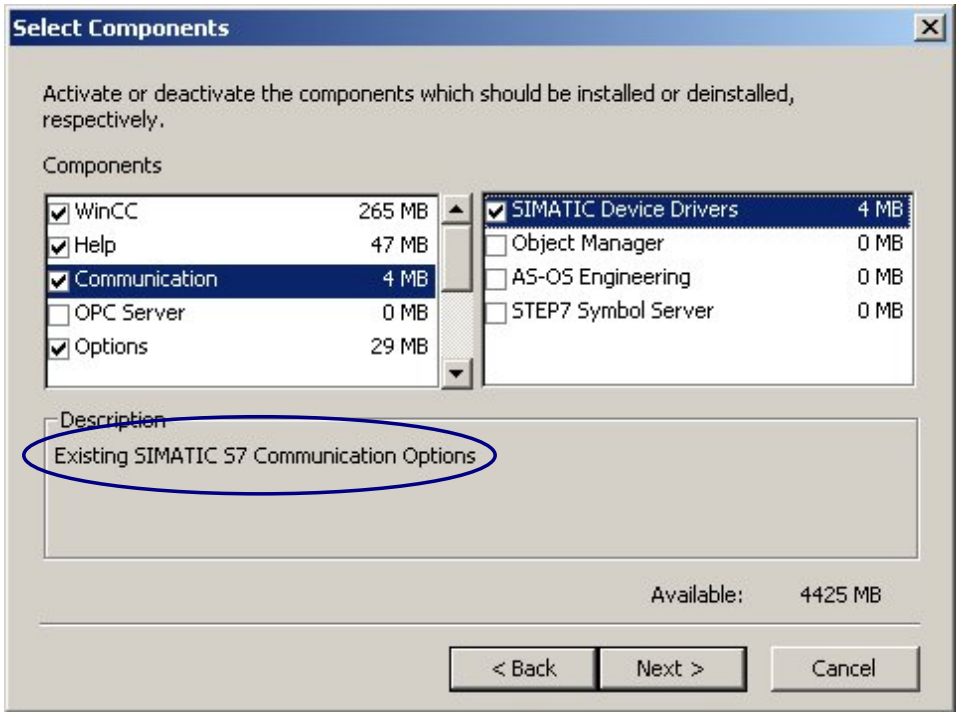


SIMATIC HMI

Date: 27.10.2003
File: a_dSwincc03e.16

SITRAIN Training for Automation and Drives

Installation of the SIMATIC Device Drivers (S7-DOS)



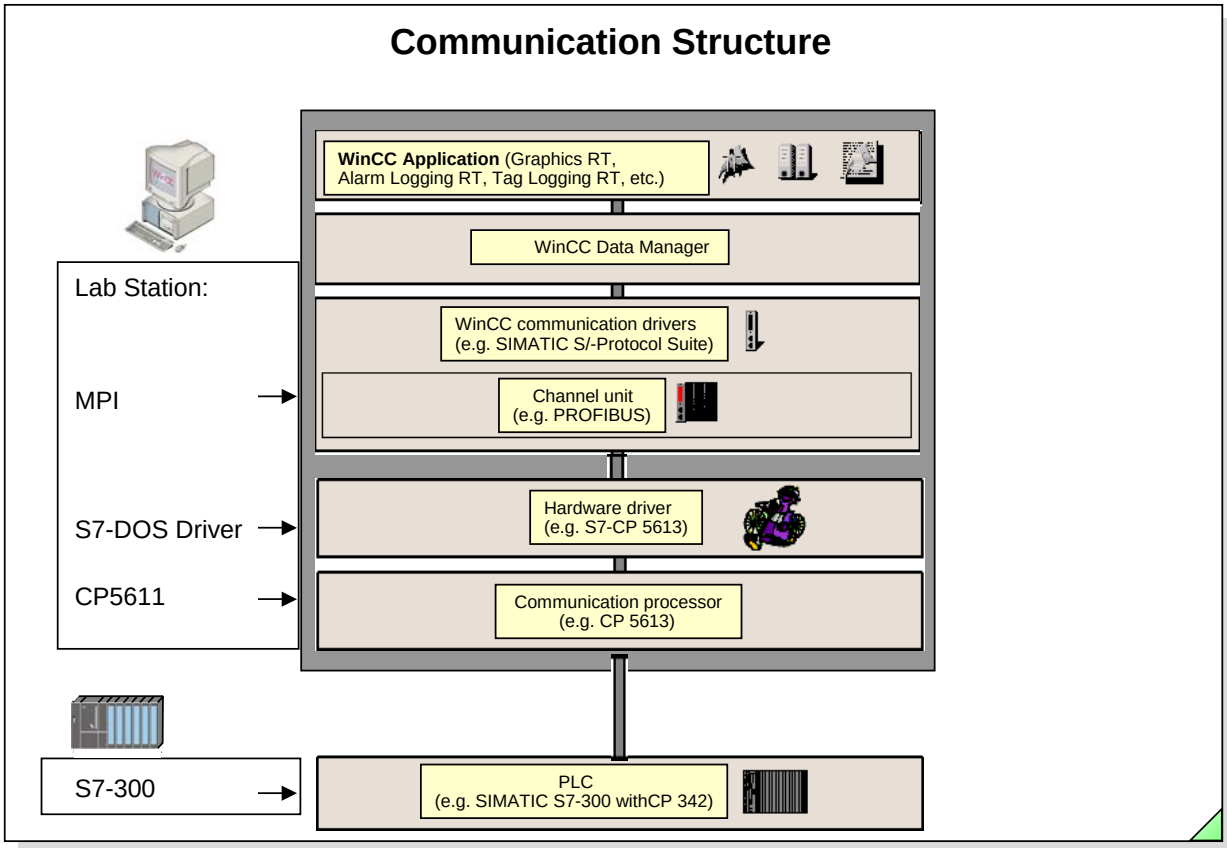
Configuration Example

The following example shows all the steps of a configuration for communication between WinCC on a Windows computer and a SIMATIC S7 300/400. The settings on the S7 side, the communications cards (CPs) and WinCC are shown by means of examples, thus producing a guide for the configuration. Depending on the system configuration, it may be necessary to use different settings, such as for the slot number of the CPU.

The exchange of data to the data manager in WinCC takes place using a data block in the S7 (DB 85). In this example one variable each of the bit, byte, word and double-word type are transmitted.

Note

The SIMATIC Device Drivers is to be selected under Communication for the WinCC installation. If this has not taken place, it can be made later on with a Delta Installation.



SIMATIC HMI

Date: 27.10.2003
File: a_dSwincc03e.18

SITRAIN Training for Automation and Drives

The Way it Works

The WinCC data manager manages the WinCC tags during Runtime execution. The various WinCC applications address their tag requirements to the data manager.
The data manager then has the task of fetching the demanded tag values from the process by using the communication driver integrated into each WinCC project. The communication driver makes the interface between WinCC and the process using one of its channel units. The hardware connection to the process is usually made using a communications processor. The WinCC communication driver then sends request telegrams to the programmable logic controller using this communications processor. The programmable logic controller sends the requested process values back to the WinCC in corresponding reply telegrams.

Data Manager

The data manager takes over the administration of the data in WinCC. This is not visible to the user. The data manager works with the data created in the WinCC project and stored in the project database. It takes over the complete management of the WinCC tags while WinCC is running in Runtime. All WinCC applications have to request the data from the data manager in the form of WinCC tags. These applications are, among others, Graphics Runtime, Alarm Logging Runtime and Tag Logging Runtime.

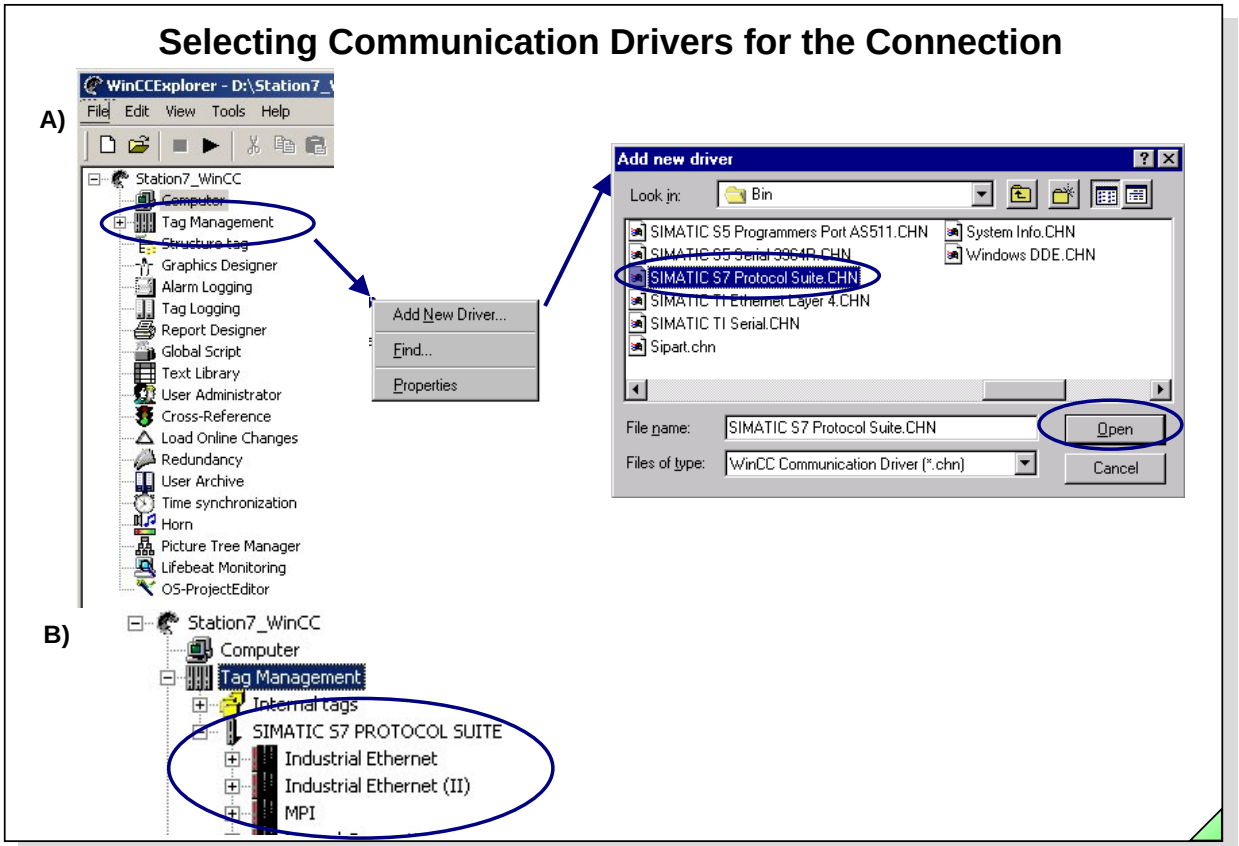
Communication Driver

Various communication drivers are used so that WinCC can communicate with the different types of available programmable logic controllers. The WinCC communication drivers link the data manager to the programmable logic controller.
The communication driver is a C++ -DLL which communicates with the data manager through the interface specified by the data manager. This interface is a so-called Channel API. The WinCC tags are supplied with process values through this communication driver.

Hardware Driver

The hardware connection to the process is usually made using a communications processor. Every channel unit forms the interface to exactly one subordinate hardware driver and therefore to exactly one communication module in the computer.

Selecting Communication Drivers for the Connection



A) Selecting Communication Drivers

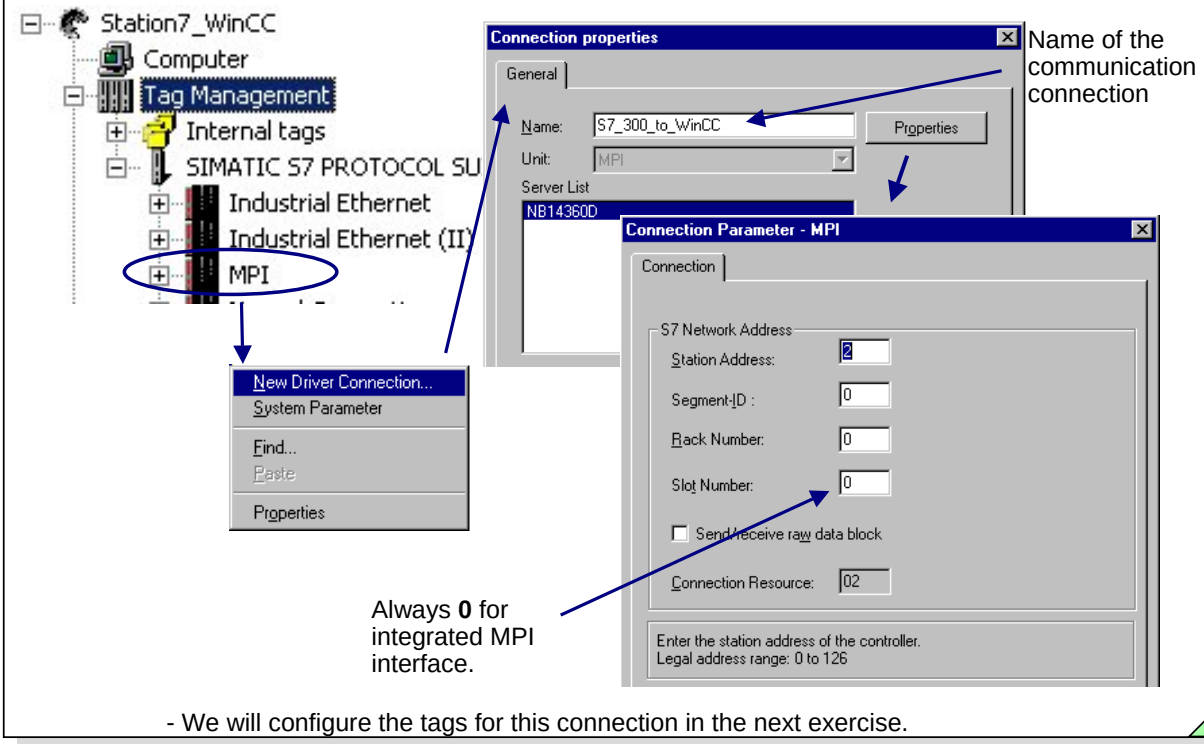
1. Use the "Tag Management" icon in picture A) to call the shortcut menu.
2. Select "Add New Driver..." in the shortcut menu.
3. In the window, search for the communication driver and then choose the driver for your procedure (for example, for SIMATIC S7Protocol Suite.CHN).

B) Configuring the Connection

In picture B) you see the updated WinCC Explorer.

1. Use the "MPI" icon to call the shortcut menu.
2. Choose "New Driver Connection..." from the shortcut menu.
3. In the "Connection Properties" window, enter a name for the connection.
4. In the "Connection" tab, enter the connection parameters for the S7 network address:
   Station address: 2 (MPI address)
   Segment-ID: 0 (not currently in use)
   Rack Number: 0 (Rack number of the CPU)
   Slot No.: 0 (CPU Slot)

Exercise: Creating a Project, Configuring the PLC Connection 1



- We will configure the tags for this connection in the next exercise.

SIMATIC HMI

Date: 27.10.2003
File: a_dSwincc03e.20

SITRAIN Training for Automation and Drives

Objective

You are to define the connections and tags necessary for data exchange between the PLC and the HMI computer (WinCC).

Exercise

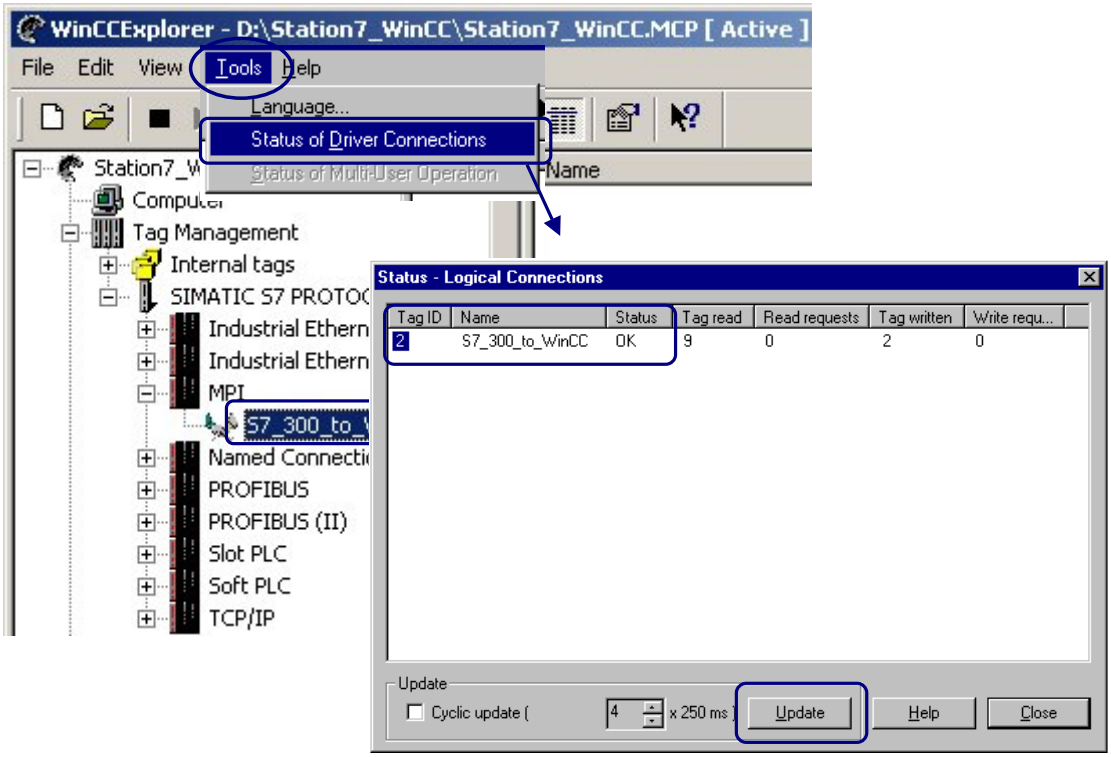
- Setup the project "ue_kurs" in drive D:

- Specify the project creator under Project Properties.

- Change the "User cycle 1" to 15 sec (setting in ms).

- Configure the properties of a connection (see slide above).

Status of the Connection



Diagnosing

You can diagnose the coupling from the WinCC side. It is done in the WinCC Explorer. If a communication link is created using the S7 Protocol Suite, you can access the diagnostic capabilities of the S7.

1. You reach the Standard Diagnosis by clicking on **Tools in the menu bar and then clicking on Status of Driver Connections in the drop-down menu**

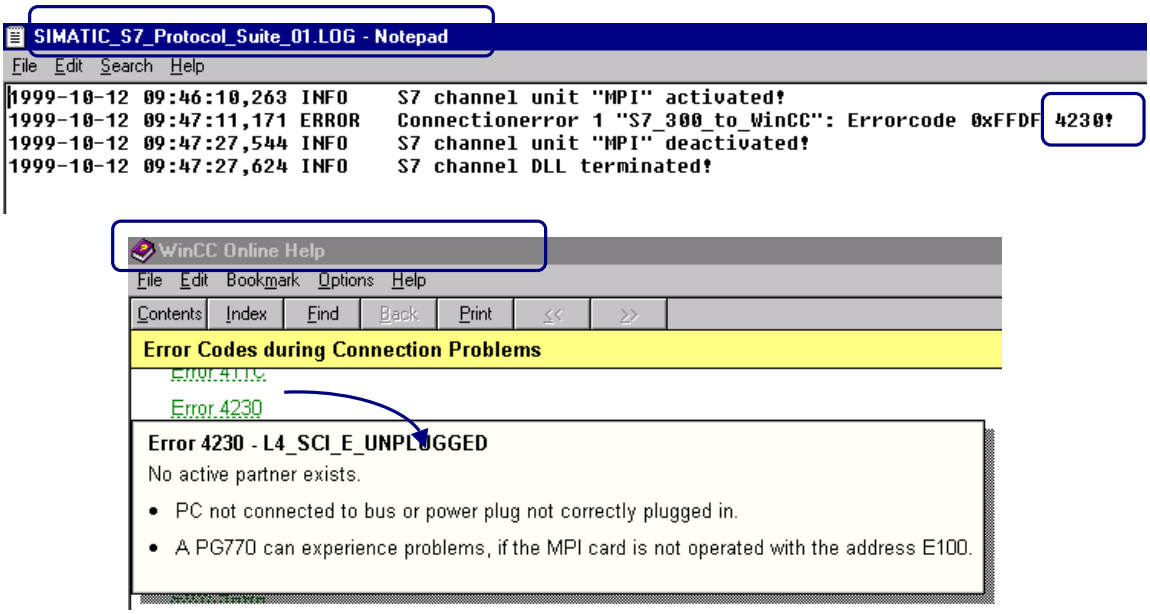
   If the number of read-only and/or write-only tasks that are displayed increase

   continually and no reduction takes place, it can lead to an overload of the data link.

2. Expanded Diagnosis, for example, C:\Siemens\WinCC\Diagnose**SIMATIC_S7_PROTOCOL_SUITE_01.LOG** whereby an exact error diagnosis is created.

   You will find an explanation for the Error No. that is displayed in the Online Help. You will find an example on the next page.

Connection Diagnostic with SIMATIC S7 (*.LOG)

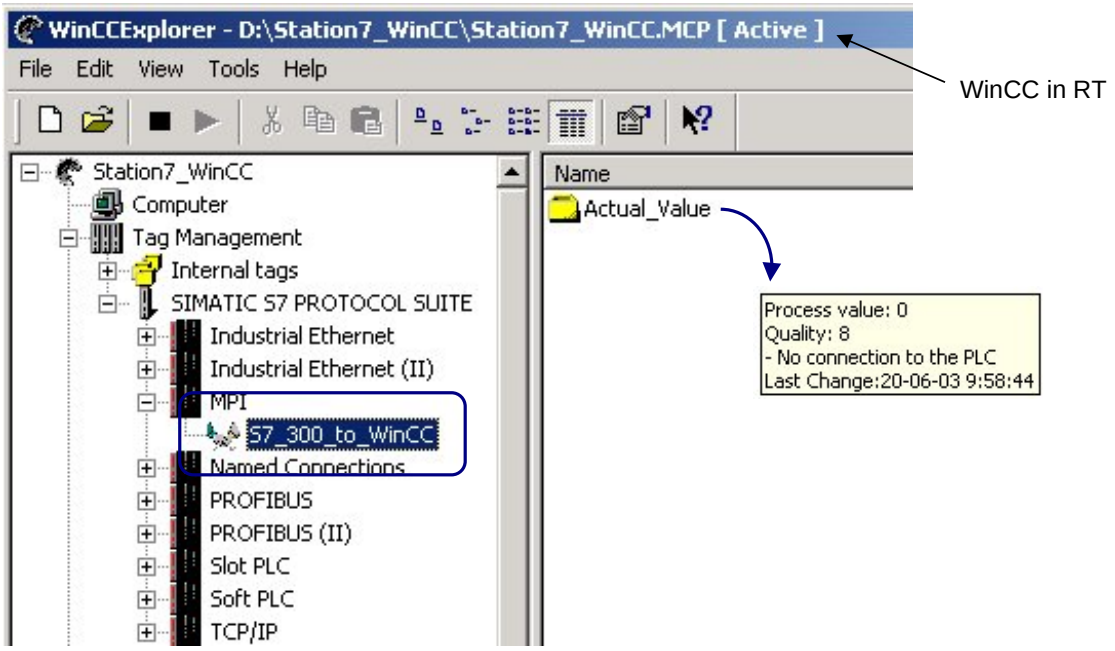


C:\Siemens\WinCC\Diagnose\SIMATIC_S7_PROTOCOL_SUITE_01.LOG

You will find important *.LOG files in theWinCC diagnostic directory.

For example,
C:\Siemens\WinCC\Diagnose\SIMATIC_S7_PROTOCOL_SUITE_01.LOG

Quality Code

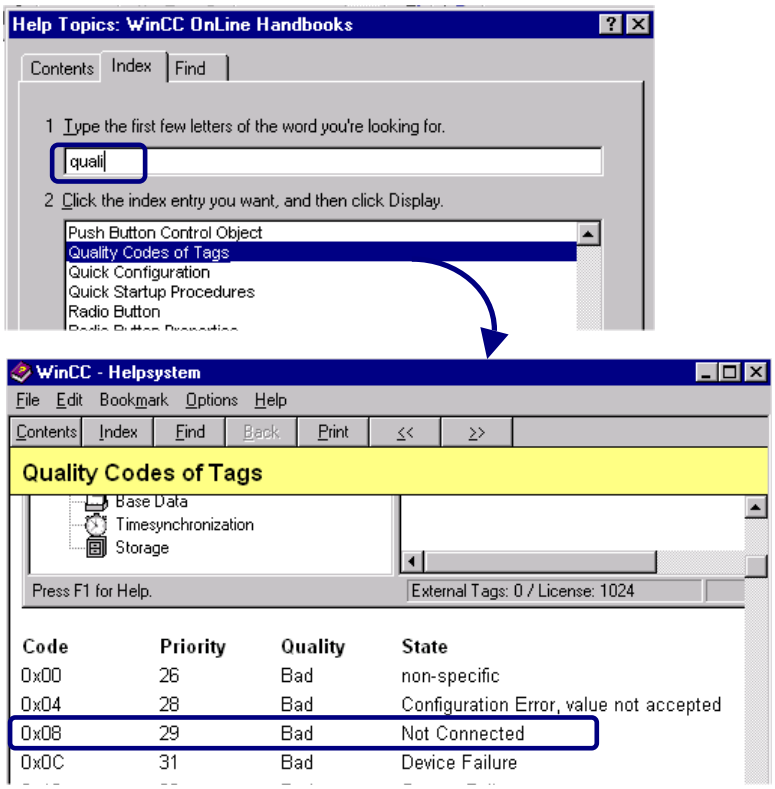


WinCC has to be in Runtime in order to display the Quality Code.

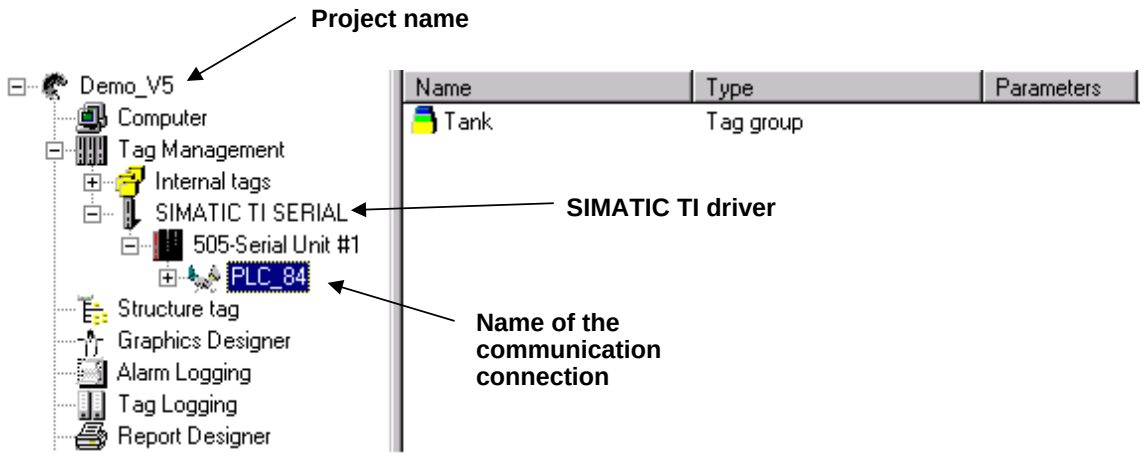
Handling

If you point to a tag with the mouse while in Tag Management during WinCC Runtime, a Quality Code appears. In addition to the current process value and the time of the last change, an error code is displayed. This error code can be used to check the status and the quality of the tag.

Online Help



Exercise: Creating a Project, Configuring the PLC Connection 2



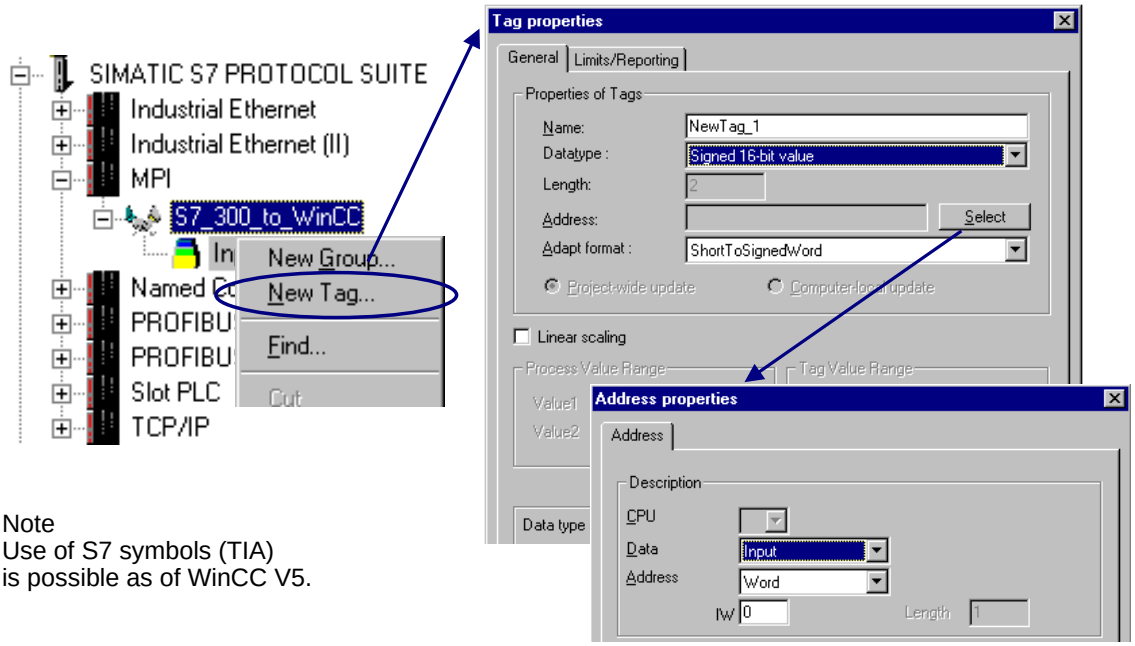
Objective

You are to define the connections and tags necessary for data exchange between the PLC and the HMI computer (WinCC).

Exercise

- Setup the project "ue_kurs" in drive C: (or D - instructors preference).
- Specify the project creator under Project Properties.
- Change the "User cycle 1" to 15 sec (setting in ms).
- Configure the properties of a connection (see slide above).

Configuring Process Tags for SIMATIC S5/S7



Note
Use of S7 symbols (TIA)
is possible as of WinCC V5.

Process Tags

Process tags establish the connection between the process and the HMI station. They are configured after a PLC connection is setup.

New Group

WinCC allows you to combine tags in technological units. This results in a tag structure that is easy to read. Before you begin with tag configuration (New Tag) create a "New Group...".

New (Process) Tag

The slide above illustrates all the configuration steps.

Linear Scaling

The process values are scaled before they are displayed.

Internal Tags

WinCC makes it possible to define tags, which do not affect process coupling, for an entire project. Tags can also be configured here either individually, or combined in groups.

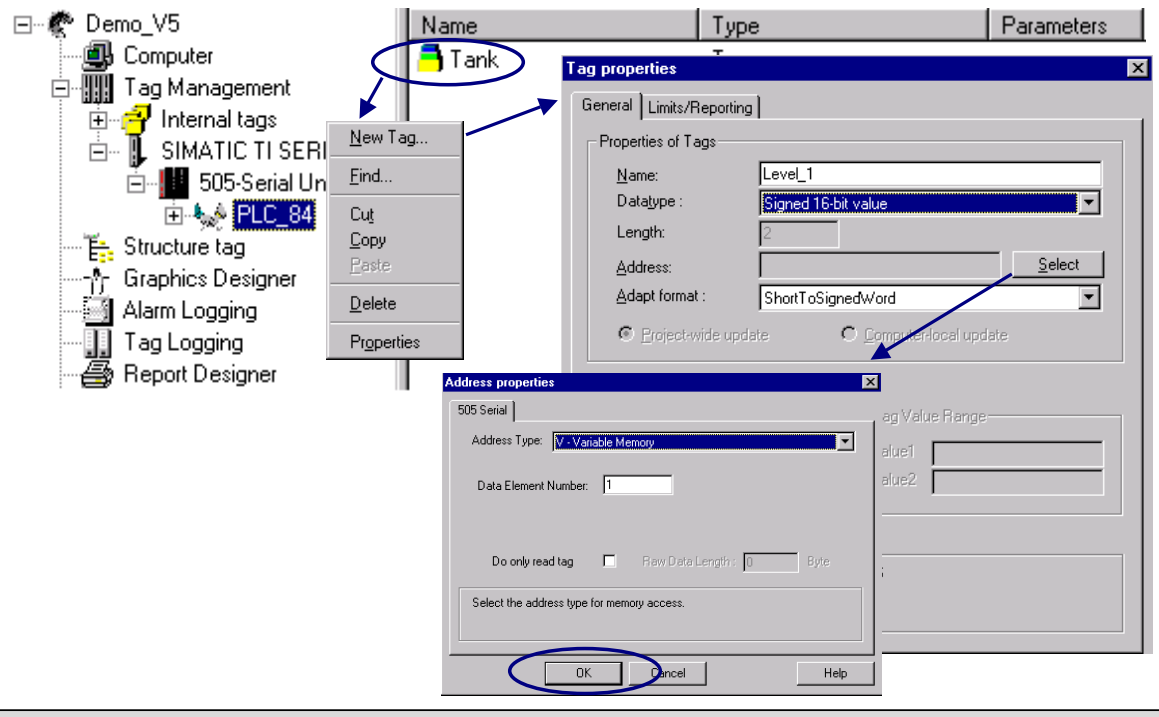
Note

The use of S7 symbols (TIA) is possible as of WinCC V5 and can be seen in an example as of Page 49.

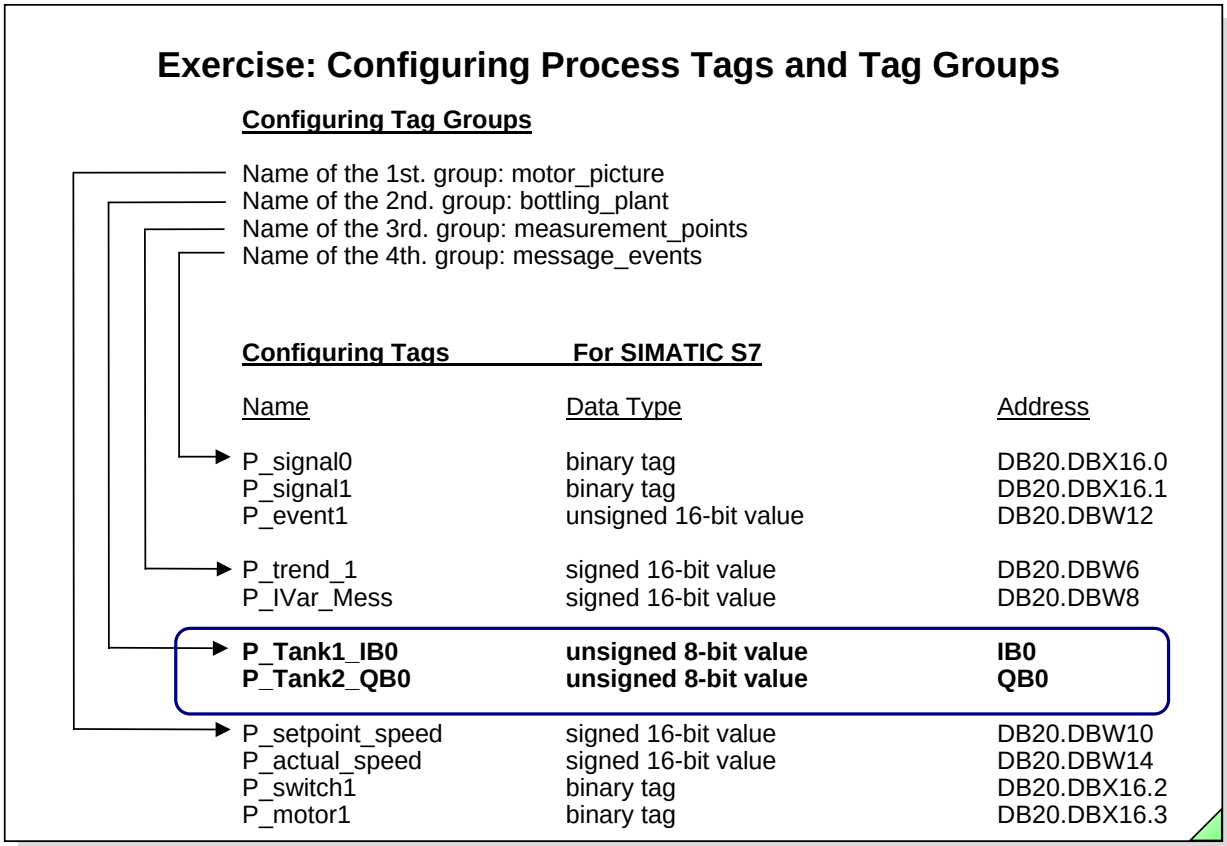
Characters Not Allowed

When you give a tag a name, you have the following restrictions:
The @ character should be reserved for WinCC system tags; it is, however, possible to use this character.
. : ? „ ' \ * $ % Blank
Capital / small letters is irrelevant.
The period is used as a separator in structure tags.
Names that begin with "$" are not visible in the Tag Management.

Configuring Tags for SIMATIC 505



Example The slide above shows a segment “New Tag" with SIMATIC 505.



Exercise

Configuration for SIMATIC S7 tag coupling.

Edit the tags P_Tank1_IB0 and P_Tank2_QB0.

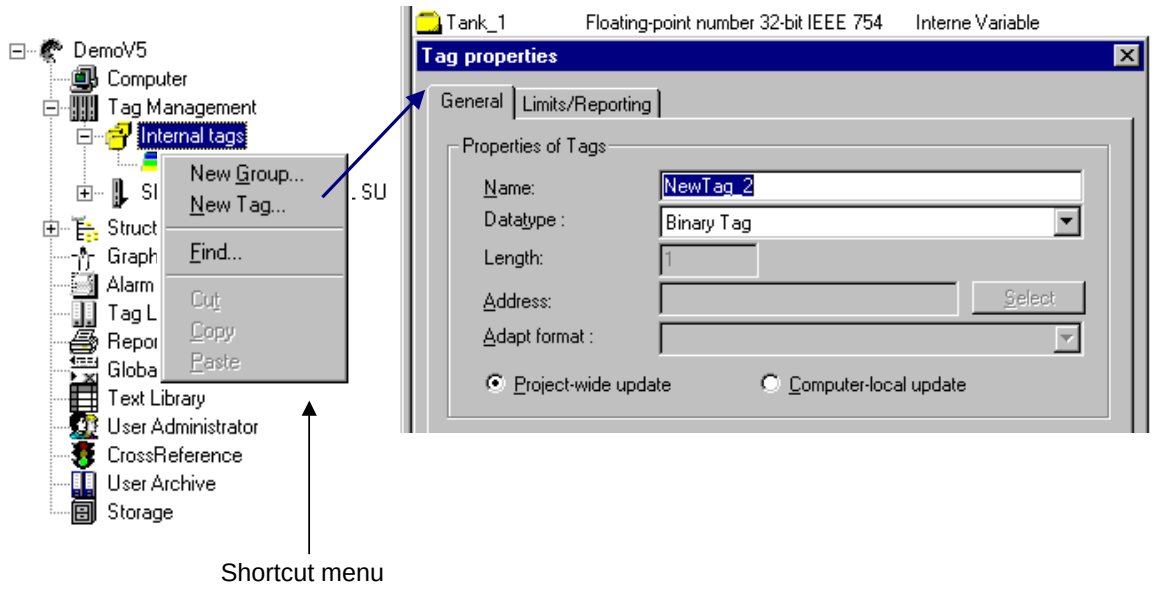
Excel

You can create and maintain large amounts of tags very effectively and easily with Excel, for example. If you use a fixed structure when you design the name of the tag, it makes it easier to create the tag lists in Excel. The tag lists that you create in Excel can then be imported into the current WinCC project with the program which can be found on your WinCC CD (install).
Call: WinCC -> Tools -> Tag Export Import.

Excel 2000, Excel XP

As of WinCC V5.1, you can install a Configuration Tool (Add-In for Excel) from the WinCC CD. With this tool you can configure mass data in Excel for connections, tags, structures, messages, archive tags etc.
The Excel spreadsheet can also be executed on **non** WinCC computers.

Configuring Internal Tags



Internal Tags

WinCC makes it possible to define tags, which do not affect process coupling, for an entire project. Tags can also be configured here either individually, or combined in groups.

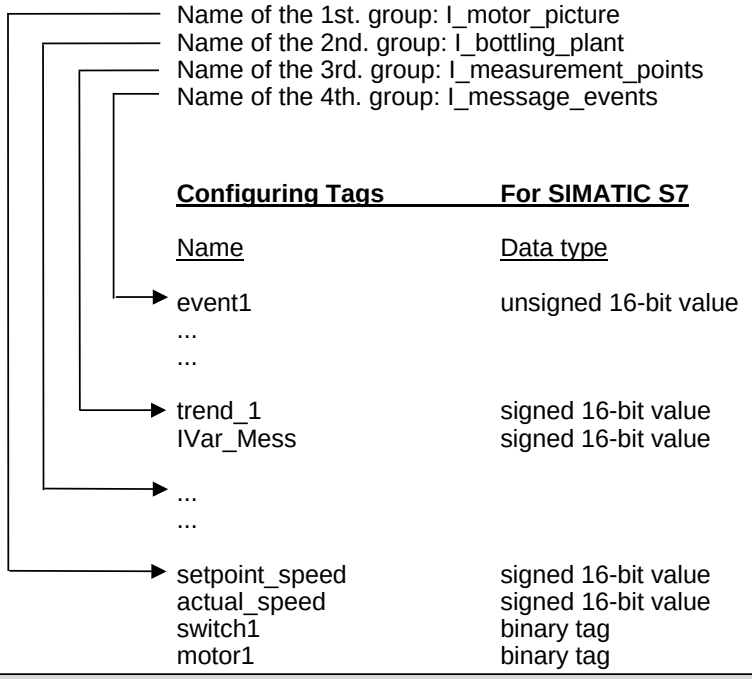
New Group

WinCC allows you to combine tags in technological units. This results in a tag structure that is easy to read. Before you begin with tag configuration (New Tag) create a "New Group...".

New (Internal) Tag

The slide above illustrates all the configuration steps.

Exercise: Configuring Internal Tags and Tag Groups 1



SIMATIC HMI

Date: 27.10.2003
File: a_dSwincc03e.29

SITRAIN Training for Automation and Drives

Exercise

You are to configure the three tag groups displayed in the slide above with the respective internal tags within the group.

Result

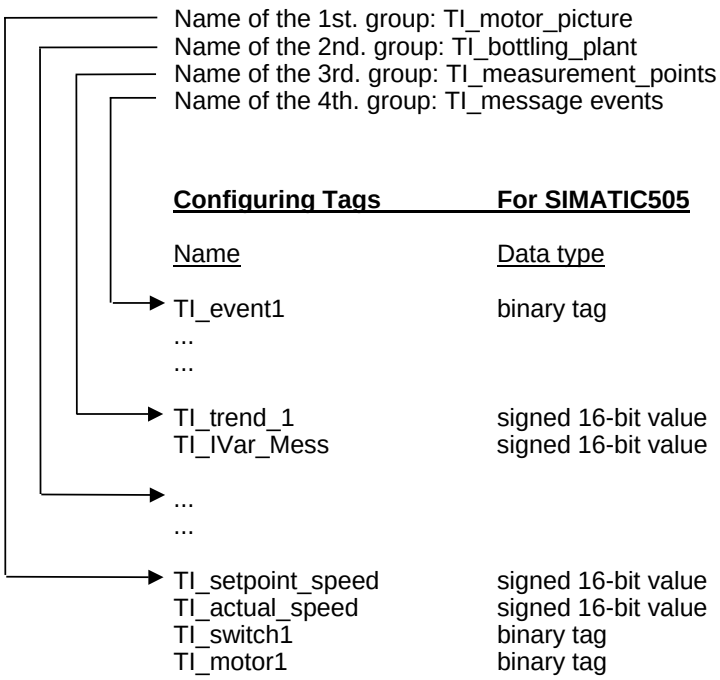
The internal tags you configured can now be linked with objects to make them dynamic (action configuration).

Note

The exercises are designed in such a way that the viewing and controlling functions can also be tested without a controller.
The tag simulation can now only affect internal tags.

Continuation on Page 37 with the installation of the hardware components and the necessary settings.

Exercise: Configuring Internal Tags and Tag Groups 2



SIMATIC HMI

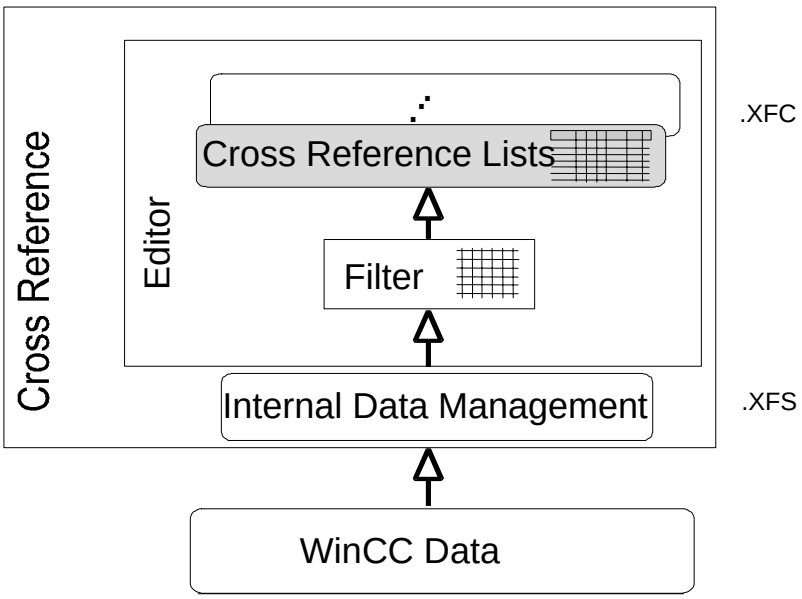
Date: 27.10.2003
File: a_dSwincc03e.30

SITRAIN Training for Automation and Drives

Exercise You are to configure the three tag groups displayed in the slide above with the respective internal tags within the group.

Result The internal tags you configured can now be linked with objects to make them dynamic (action configuration).

"Cross Reference" (Cross Reference List 1/3)



Cross Reference

With the WinCC component "Cross Reference" (**Cross Reference List**) it is possible:

-to find all the application locations of specific objects, such as tags, screens, etc.

-to directly display the application location of an object in order to change it or delete it (Function: "**Jump to Place of Use**").

-for tags, to change the tag name using the function "**Linking**" without causing inconsistencies in the configuration.

Filter

You can use an adjustable filter to search for the following object types and create a cross reference list:

-process tags and internal tags

-screens and screen windows

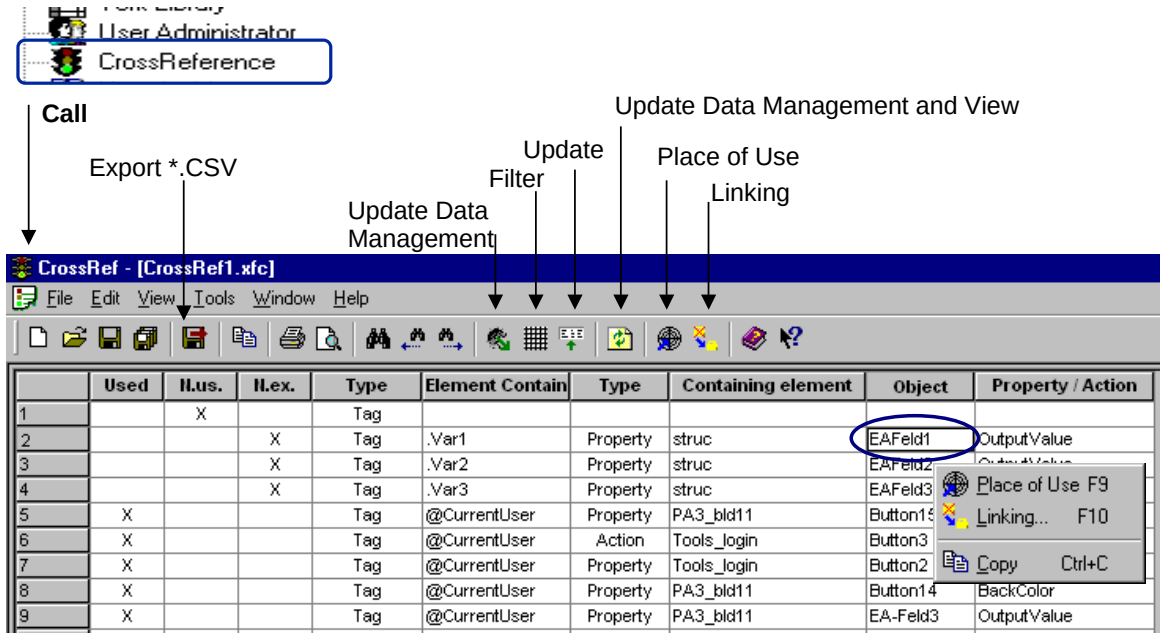
-archives

-project functions and standard functions

-OLE Objects

-ActiveX Controls

"Cross Reference" (Cross Reference List 2/3)



Tags in C-Actions

Tags in actions can only then be found and replaced by the Cross Reference if the **configuration rules for tags and screen names** predefined in the C-Scripts are adhered to.

Configuration Rules for C-Scripts

// WINCC:TAGNAME_SECTION_START
// syntax: *#define TagNameInAction "DMTagName"*
// next TagID : 1
#define TAG_ SoLLM1 "SoLLM1"
// WINCC:TAGNAME_SECTION_END

// WINCC:PICNAME_SECTION_START
// syntax: #define PicNameInAction "PictureName"
// next PicID : 1
// WINCC:PICNAME_SECTION_END

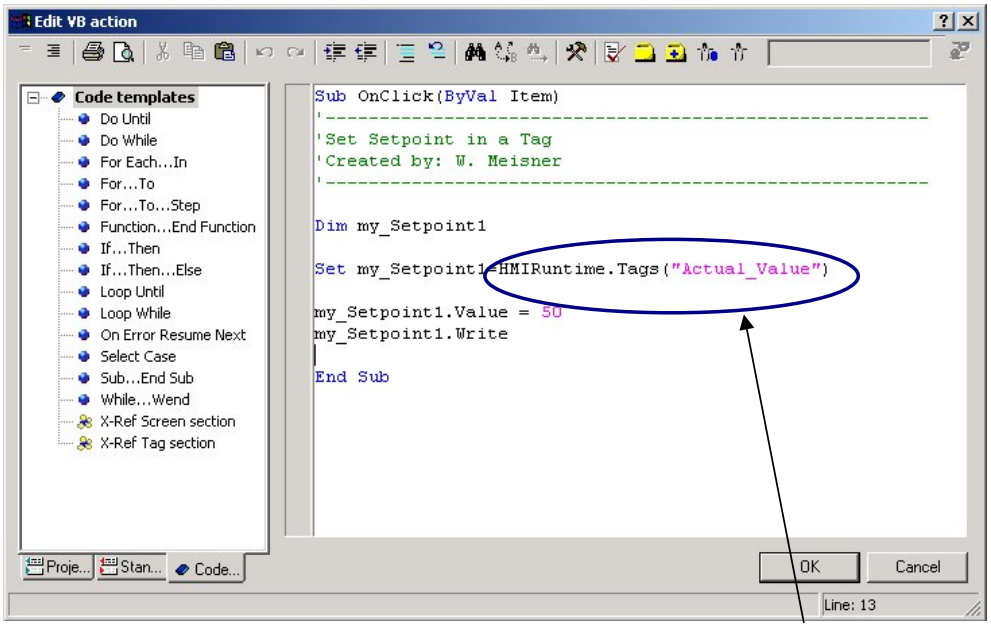
SetTagDWord(TAG_ SoLLM1,100);

Note

The configuration rules are already taken into consideration by the Dynamic Wizards during generation of the scripts.
When you set up C-Scripts, a comment recognizes the configuration rules.
Screen names are to be written without the file extension ".PDL"

You can find further information in the WinCC **Online Help** under Contents, Editors, Cross Reference.

"Cross Reference" (Cross Reference List 3/3)



Tag Actual_Value is automatically listed in der Cross Reference.

Tags in VBS Actions

With the WinCC CrossReference, you can also quickly find all Places of Use of tags and screen names in VBS Actions. You can "rewire" tag triggers from actions in the Graphics Designer with CrossReference; in other words, replace them with other tags in all or in selected positions.

Configuration Rules for VBS Scripts

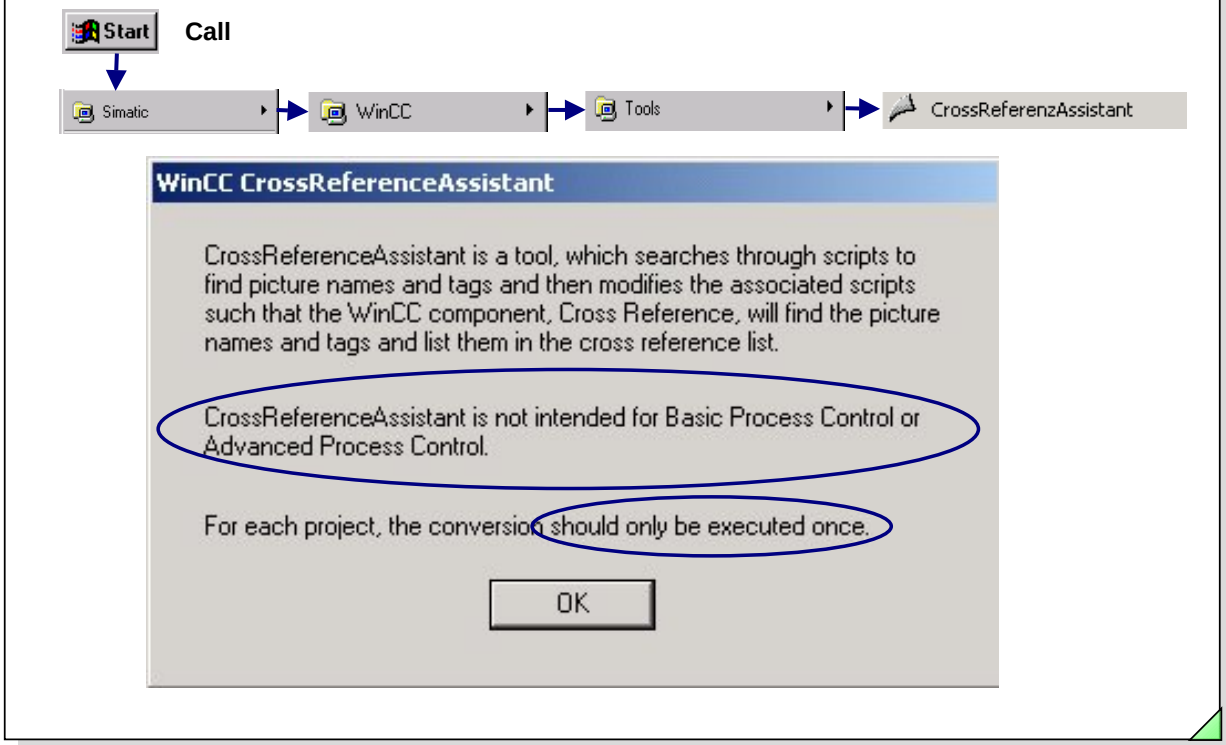
All tags that you address with the standard format
HMIRuntime.Tags("Tagname")
are automatically recorded by the WinCC CrossReference and are listed in the screen properties.
Should you address tags with other formats in your code, you can access them through the following section of the CrossReference:
'WINCC:TAGNAME_SECTION_START
'Const TagNameInAction = "TagName"
'WINCC:TAGNAME_SECTION_END

All screens that you address with the standard format
HMIRuntime.BaseScreenName("Screenname")
are automatically recorded by the WinCC CrossReference and are listed in the screen properties.
Should you address screens with other formats in your code, you can access them through the following section of the CrossReference :
'WINCC:SCREENNAME_SECTION_START
'Const ScreenNameInAction = "ScreenName"
'WINCC:SCREENNAME_SECTION_END

Note

Screen names are to be written without the file extension ".PDL".

Tools: CrossReferenceAssistant



Objective

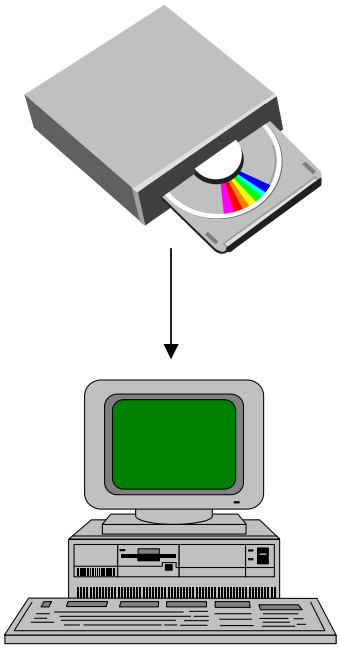
A WinCC CrossReferenceAssistant, also included in the package as of Version 5.0 SP1, generates comment and define lines in C-Action lists.

WinCC CrossReferenceAssistant is a tool which searches through scripts to find screen (picture) names and tags and then modifies the associated scripts so that the WinCC component, CrossReference, can find the screen (picture) names and tags and list them in the Cross Reference List.

Note

The Tool is installed from the WinCC CD.

Tag Simulation 1/2



Simulation software on WinCC CD

- Sine
- Oscillation
- Random numbers
- Increment
- Decrement
- Slider

Testing functions with internal and process tags.
Process tags can only be simulated if the communication connection to the PLC has been made. Since the tag value in the PLC can be changed by the simulation, you have to take the necessary precautions (such as separating the load current circuits).

SIMATIC HMI

Date: 27.10.2003
File: a_dSwincc03e.35

SITRAIN Training for Automation and Drives

Simulation

The simulation software provides the person configuring the system or the installer with six different functions.
You can assign one of these six functions to each individual tag.
You can use these functions to supply the configured objects with realistic values.

Practical Functions

The simulation software provides the following practical functions:

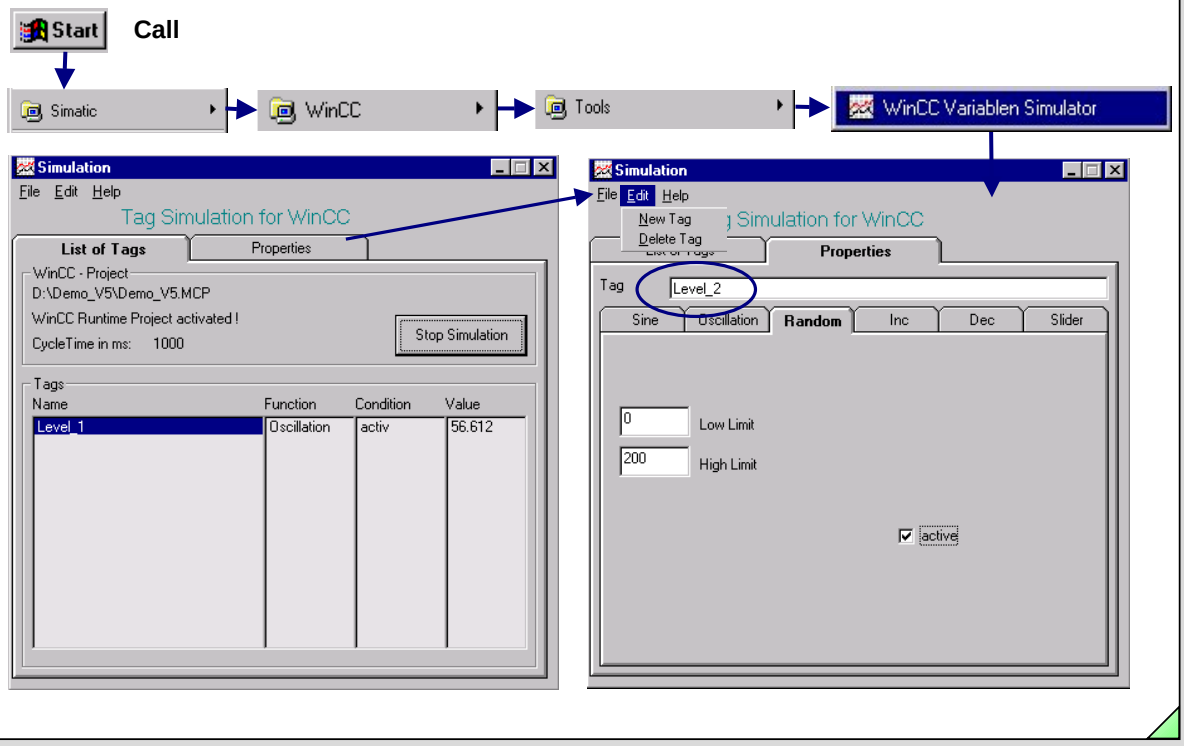
- Sine (as periodic, non-linear function).
- Oscillation (for simulating jumps of a reference input variable).
- Random numbers (provides you with randomly generated values).
- Increment (an up counter that begins again at the minimum value when it reaches the maximum value).
- Decrement (a down counter that begins again at the maximum value when it reaches the minimum value).
- Slider (a linear regulator that you can use to set a fixed value).

Note

For installation, you can find this simulation program on your WinCC-CD.

To simulate process tags, you can also use the SIMATIC tool "PLCSIM“ (see FAQ 6859162). You do not require SIMATIC hardware for PLCSIM.

Tag Simulation 2/2



Simulation

All tags can be simulated during Runtime-mode with the simulation program provided. The application is used for presentations and for testing functions.

Note

Process tags can only be simulated if the communication connection to the PLC is made. Since the tag value in the PLC can be changed by the simulation, you have to take the necessary precautions (such as separating the load current circuits).

Simulating Tags

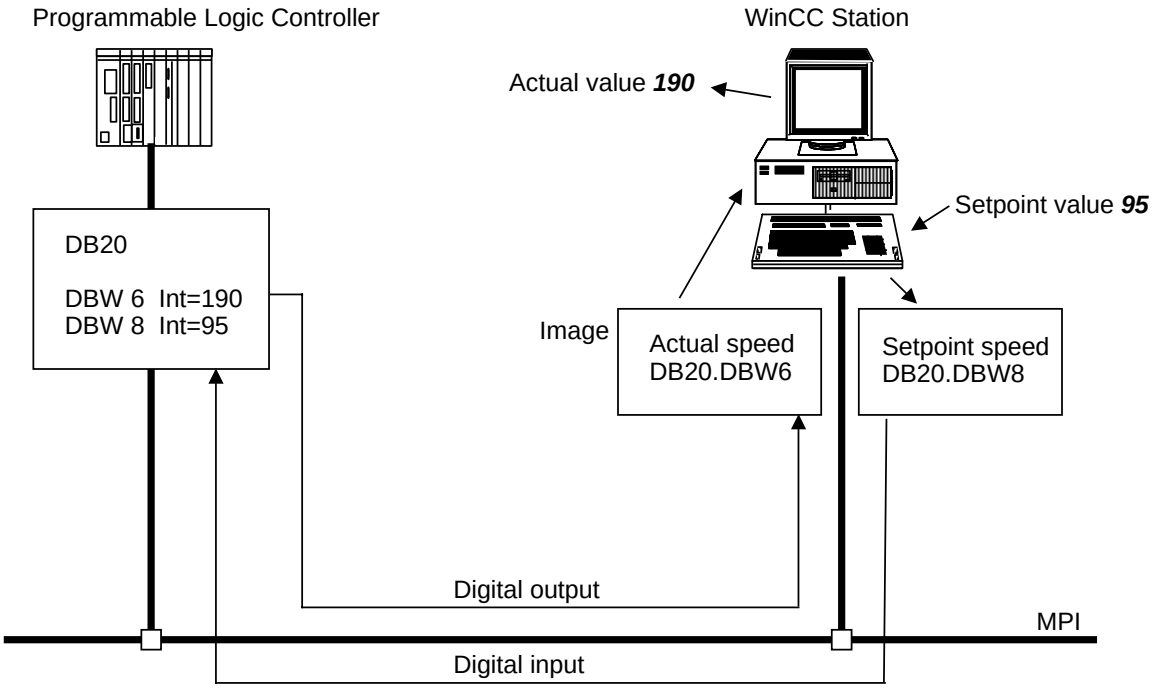
Creating tags for the tag table

1. In the menu bar, click on “Edit” .
2. In the drop-down menu, click on “New Tag”.
3. Enter the following settings in the properties window that is displayed:
   for example, Level_2,
   the function, for example, Random,
   Low Limit **0**
   High Limit **200**
   and then click (check mark) the ‘active’ check box.

Note

The update time for tag values is 1 second. A change does not go into effect until the function is activated.
A **maximum of 300 tags** can be configured.

WinCC Communication with S7 Protocol using MPI to S7- 300/400



SIMATIC HMI

Date: 27.10.2003
File: a_dSwincc03e.37

SITRAIN Training for Automation and Drives

Requirements
Hardware

- A SIMATIC S7
- An MPI bus for at least two nodes
- A WinCC station with CP5611 card

Software

- SIMATIC STEP 7 programming software
- WinCC Software or WinCC station

Note

The "SIMATIC Device Drivers" driver software is necessary for the softnet modules CP5511and CP5611 and must be installed from the WinCC CD. The WinCC communication drivers are found in the SIMATIC S7 Protocol Suite.CHN.

Communication Components

On the PLC side

CP	CP Software
MPI interface of the CPU	-

On the PG/PC side

CP	CP Software
CP 5611	(S7 DOS from WinCC) SIMATIC Device Drivers

On the WinCC side

WinCC Channel DLL	Core / Option
S7 Protocol Suite - MPI	Core

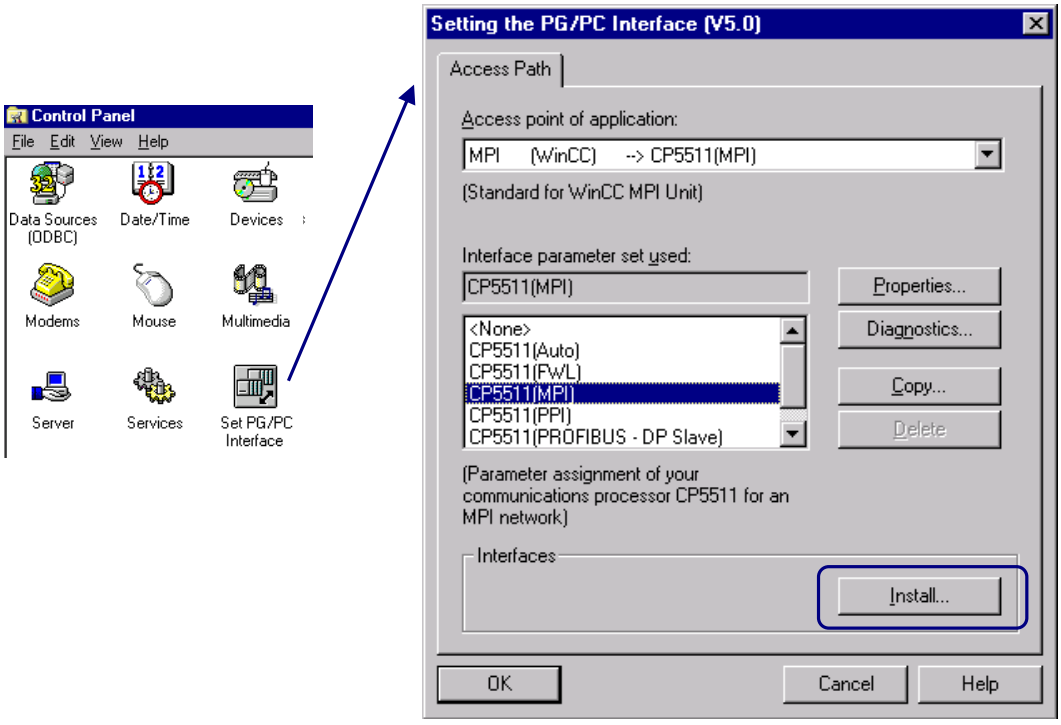
Procedure

1) **Installation of the CP5511 card according to the product information**

2) **Configuration on the WinCC side,** see Page 19.

3) **Configuration in the SIMATIC CPU.**
   No communication programs are necessary here.

Installation of the CP5511 Card in the PC

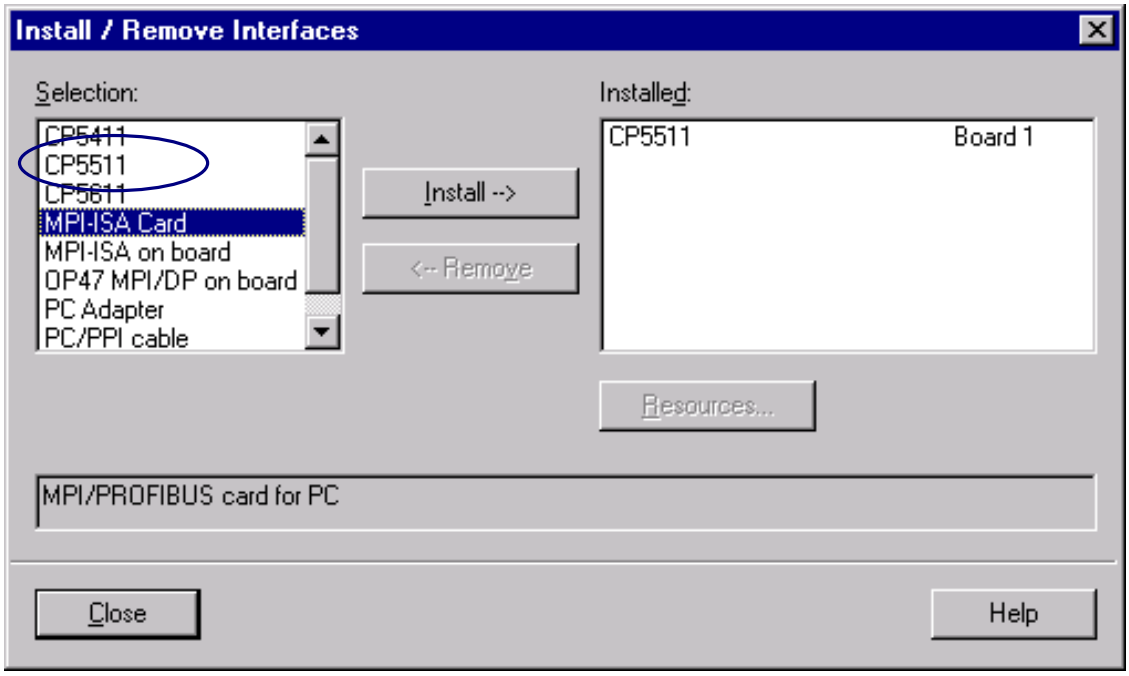


Date: 27.10.2003
File: a_dSwincc03e.40

The first step is to install the CP5511 card in the PG/PC. To do so, select Control Panel --> Set PG/PC Interface.

With Interfaces --> Install, you select the next window.

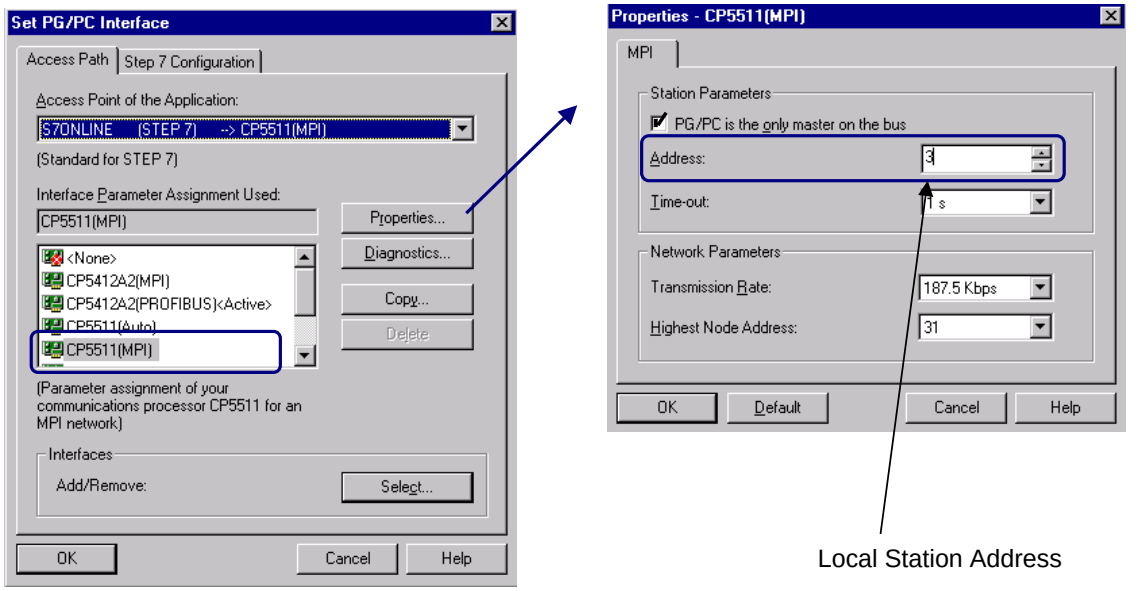
Installing Modules



The CP5511 is selected from the ‘Selection‘ browser and the module is installed. The Resources setting is no longer required since the CP is configured independently.

Note The entries in the Selection window are dependent on the module driver already installed.

Specifying Properties of the CP5611 Card

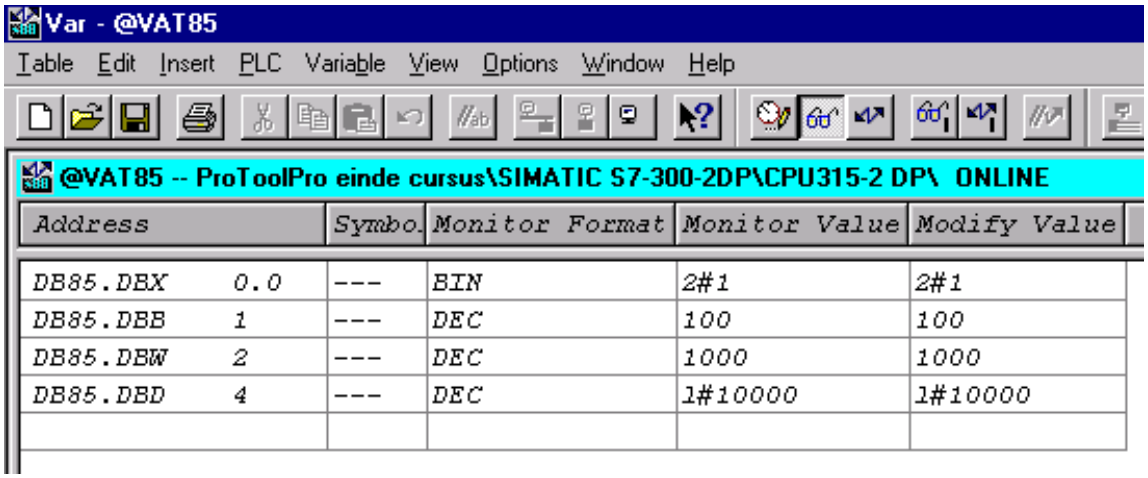


Date: 27.10.2003
File: a_dSwincc03e.42

The last step in the installation of a CP5511 card in the PC is to specify the properties of the card.

In our case, enter ***0*** as the Station address for the card. The Transmission Rate of 187.5 Kbps is fixed for the S7 300. For the S7 400, 1.5Mbps is also possible.

S7 Variable Table for Monitoring and Modifying



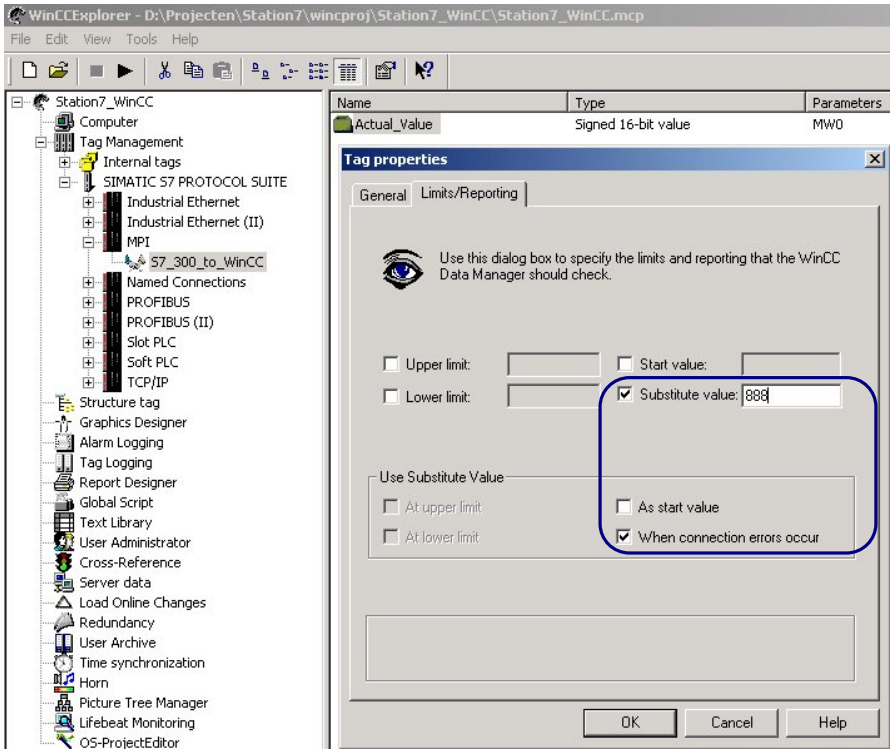
Address		Symbo.	Monitor Format	Monitor Value	Modify Value
DB85.DBX	0.0	---	BIN	2#1	2#1
DB85.DBB	1	---	DEC	100	100
DB85.DBW	2	---	DEC	1000	1000
DB85.DBD	4	---	DEC	L#10000	L#10000

For an MPI coupling on the side of the S7, no additional card is required if the existing default PG port is used. Only the MPI node (network) has to be created and the MPI address of the PLC has to be defined. In our example it is Address **2**, which we also have to specify in WinCC.

It is assumed that a project has been created and that the hardware components have been defined and a data block has been programmed.

In our example, we have also created a variable table (VAT85) for the Monitoring and Modifying Variables and have predefined the variables for purposes of control.

Connection Status



Date: 27.10.2003
File: a_dSwincc03e.44

General Information

If process tags or even internal tags are to be used in actions, you must read the values via GetTag actions.
Permanent monitoring of the process connection from the PLC to WinCC can be implemented with the GetTagState action (for example, a warning box pops up in case of a fault).
Set Tag actions are made available for assigning the results of calculations.
The SetTagState action is available if the result of the SetTag action is to be checked.
The success of a value change can be queried with the SetTagState action (writing from WinCC to the PLC). This is especially important for process tags because the most varied coupling faults can occur there.

The status is transferred as a bit register and can be either evaluated by the Dynamic Dialog or by the status' own actions.

Evaluation

-available as a system function by displaying a shading. No configuration required.

-definition of a substitute value for a process tag that pops up when there is a connection error. The setting is made in the *Limits/Reporting* tab (see slide above).

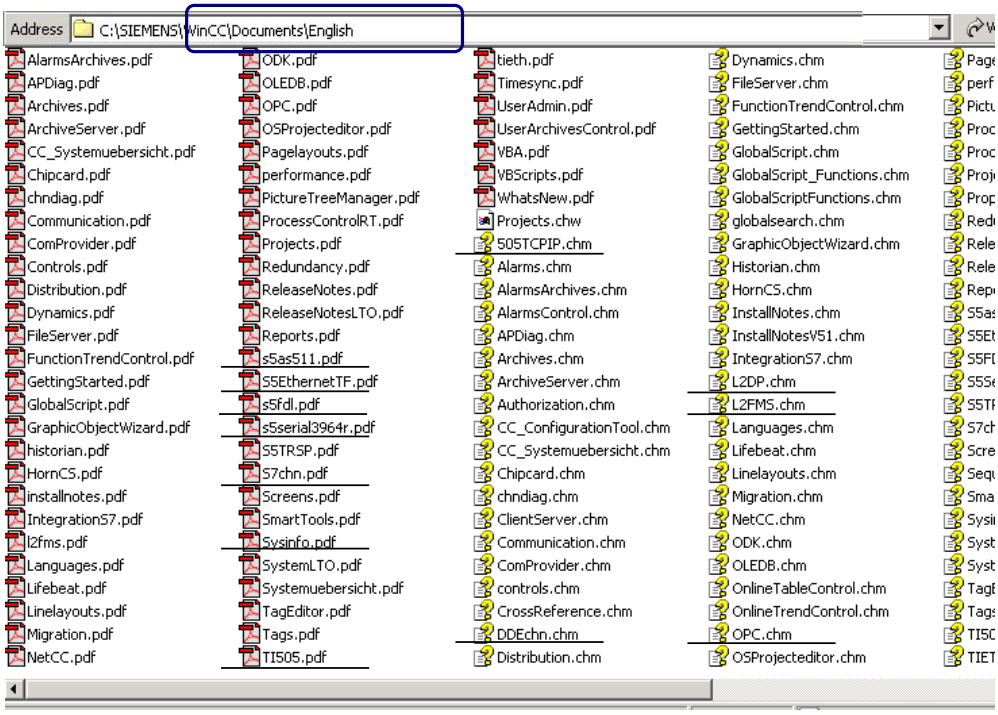
-the status of a process tag can also be evaluated with the Dynamic Dialog (see the Chapter on Graphics Screens). Configuration with many possibilities for making dynamic.

-available as a standard function in the option package Basic Process Control Package.

-configuration with Toggle in the Simatic program, for example.

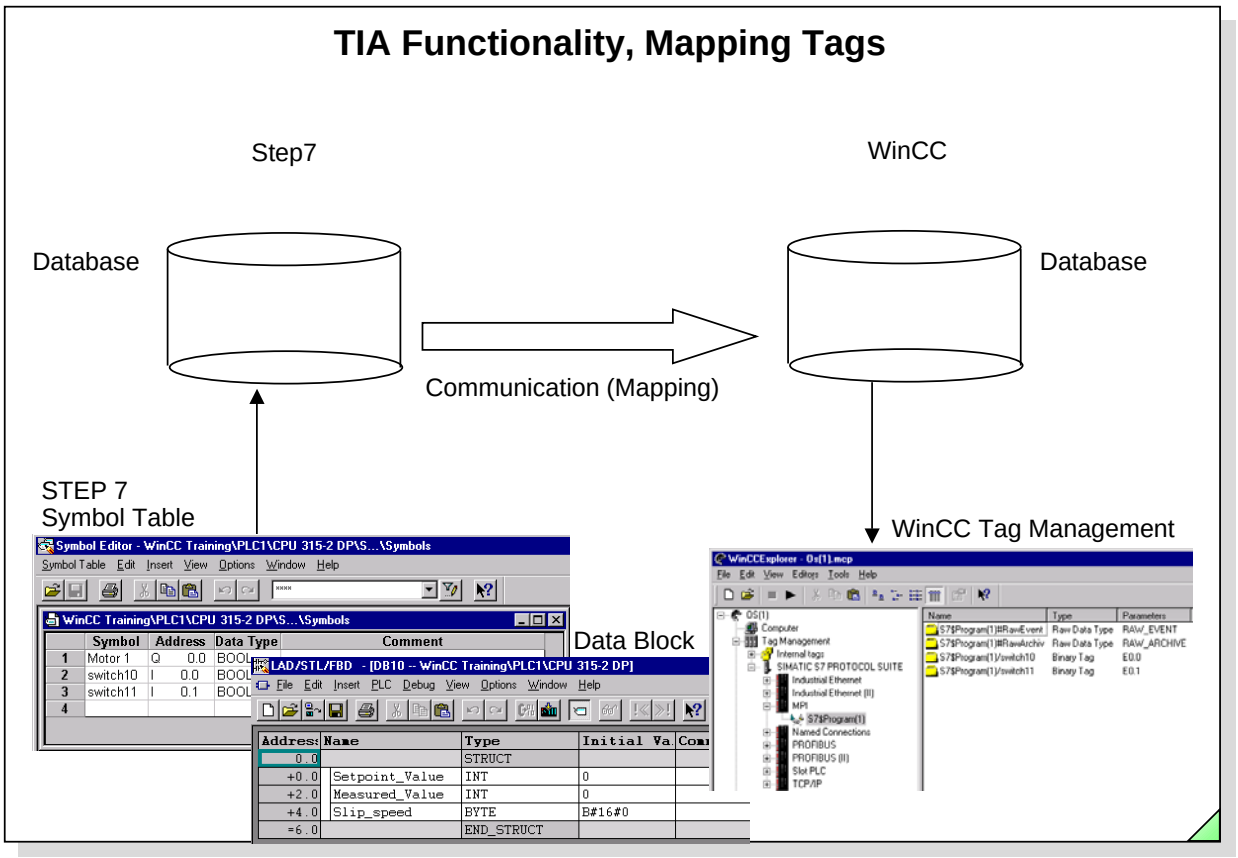
-configuration using C-Scripts, see Chapter 15.

Documentation for Communication



Date: 27.10.2003
File: a_dSwincc03e.45

You will find additional examples in the Communication Manual, Order No.: 6AV6 392-1CA05-0AA0 and on the CD or on the hard disk after installation of the software (see slide above).



SIMATIC HMI

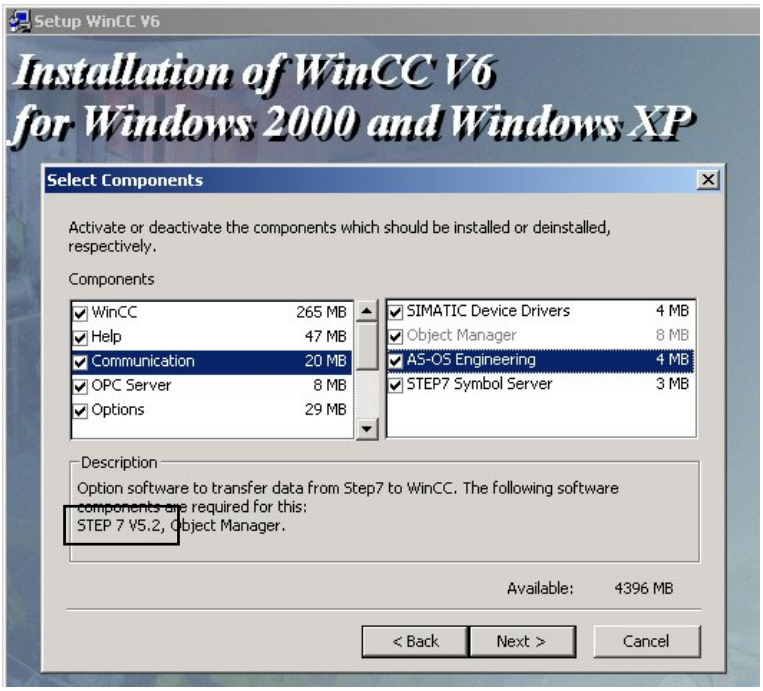
Date: 27.10.2003
File: a_dSwincc03e.46

SITRAIN Training for Automation and Drives

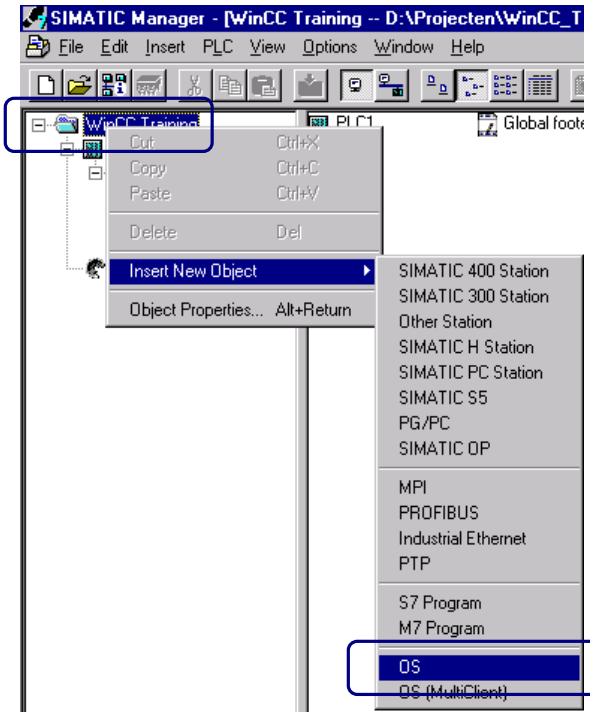
Requirements

-versions must fit together (**Step7 V5.2 and WinCC V6.0**).
-in the User Defined Installation of WinCC, under Communication, you have to enable the selection "Object Manager" and "PLC-OS Engineering".

Inputs, outputs, bit memories and elements of data blocks are mapped.

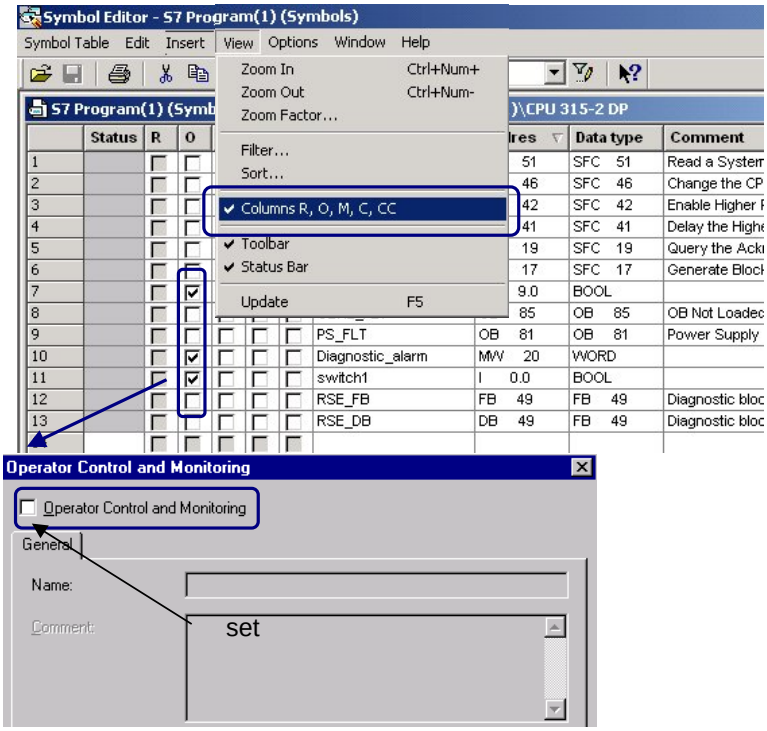


Creating an OS (Operator Station)



In order to generate connection data, it is necessary to create an OS station (operator station) in the SIMATIC Manager.

Creating a Symbol Table, Generating a Data Block 1/2



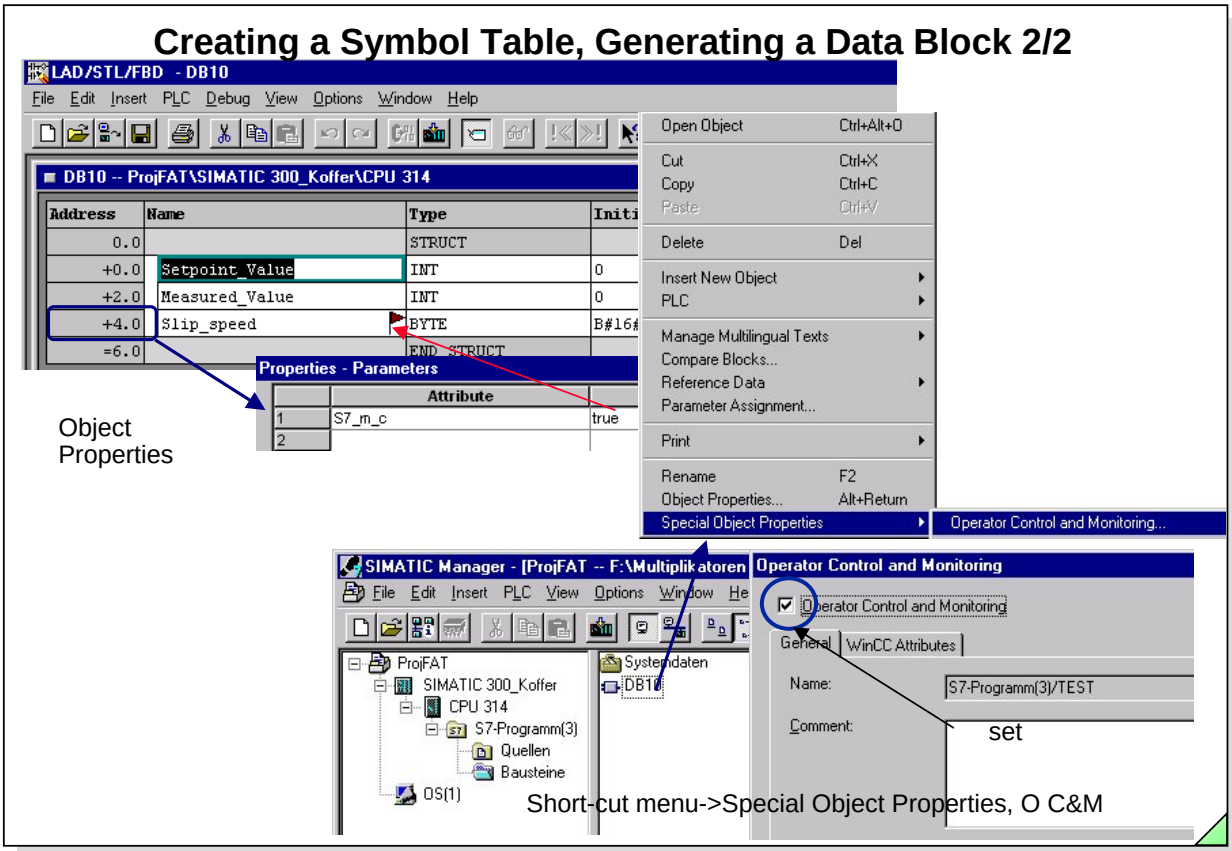
Column B in the Symbol Table tells the program designer whether or not the attribute has been set for operating and monitoring.
Through the shortcut menu and the call of special object properties for operating and monitoring, the attribute can be set for every individual address.

This is necessary so that during transferring (mapping) from the S7, the tags in the Tag Management of WinCC are created.

The various possibilities of mapping are explained on the following pages.

Note The flags are also set when mapping takes place from the WinCC side.

Creating a Symbol Table, Generating a Data Block 2/2



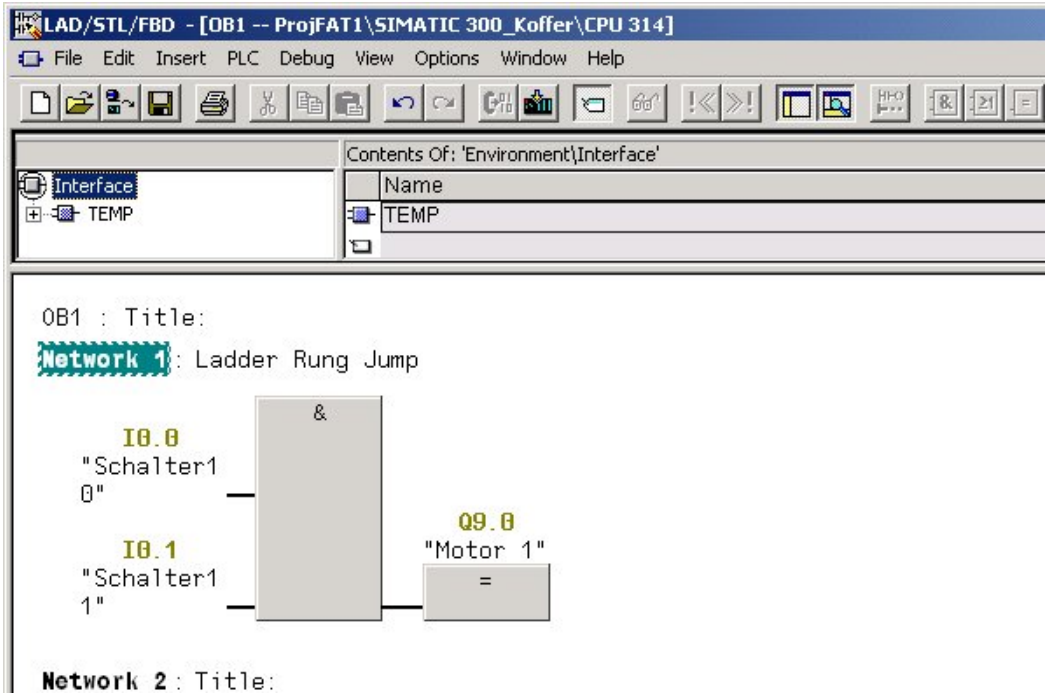
The flag (only for DBs) tells the program designer whether or not the attribute has been set for operating and monitoring.
This is necessary so that during transferring (mapping) from the S7, the tags in the Tag Management of WinCC are created.

S7_m_c This attribute is used when the parameter is to be operated or monitored from an HMI device.

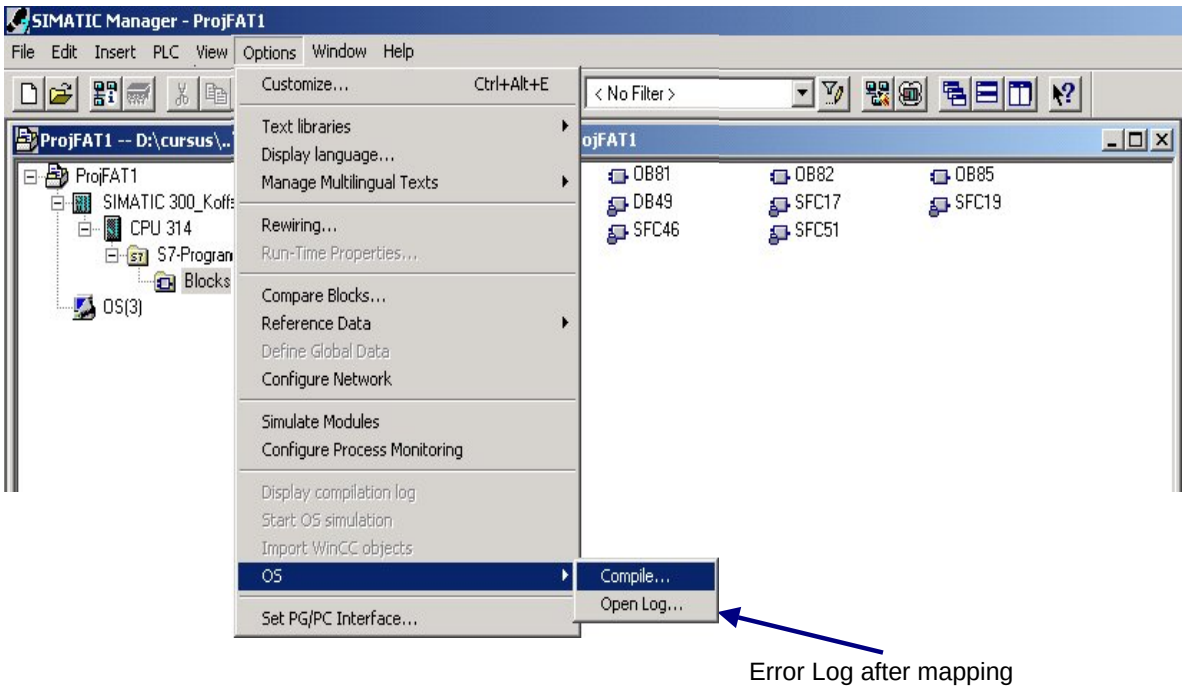
The various possibilities of mapping are explained on the following pages.

Note The flags are also set when mapping takes place from the WinCC side.

Creating an S7 Program



Generating PLC/OS Connection Data (Mapping)



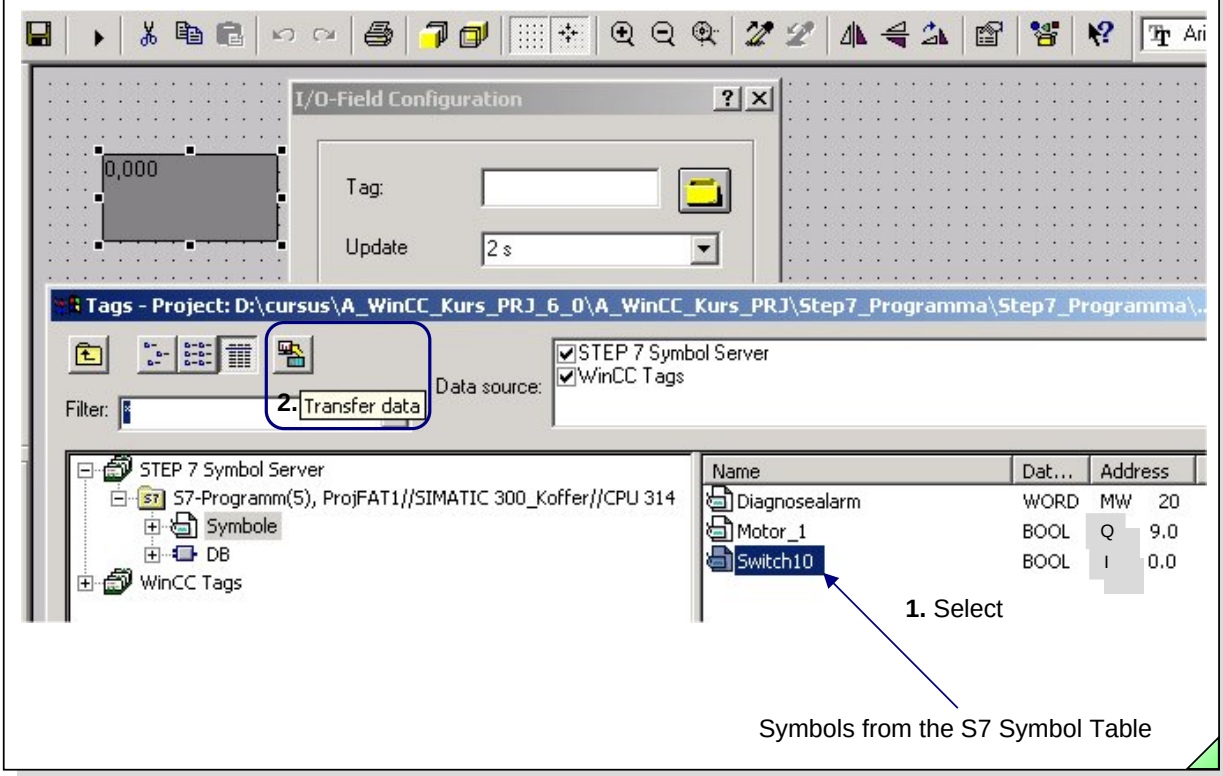
There are three ways of generating the connection data.

-from STEP 7 PLC/OS Connection Data, Transfer (Only if operating and monitoring attributes were set).

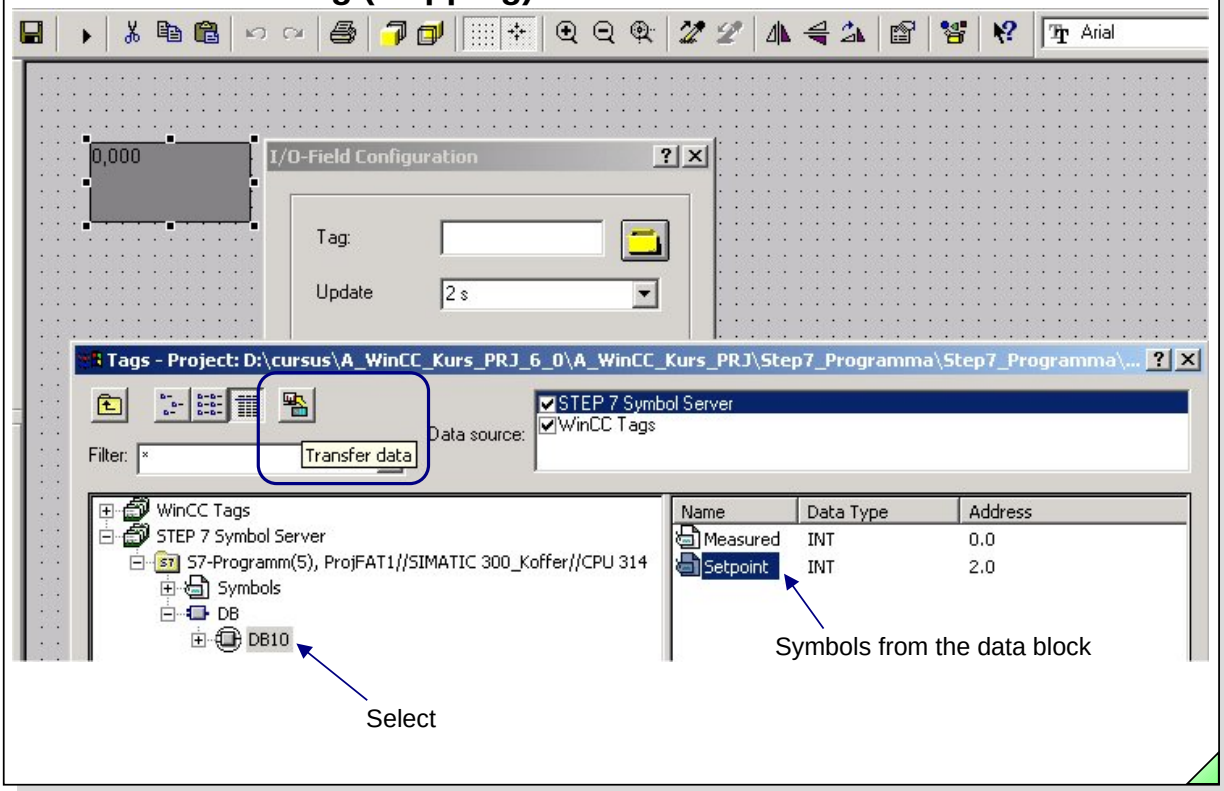
-from WinCC by triggering Transfer Data (Possible without setting attributes).

-from WinCC automatically in the background (Possible without setting attributes).

Transferring (Mapping) Data from Step7 to WinCC 1/2

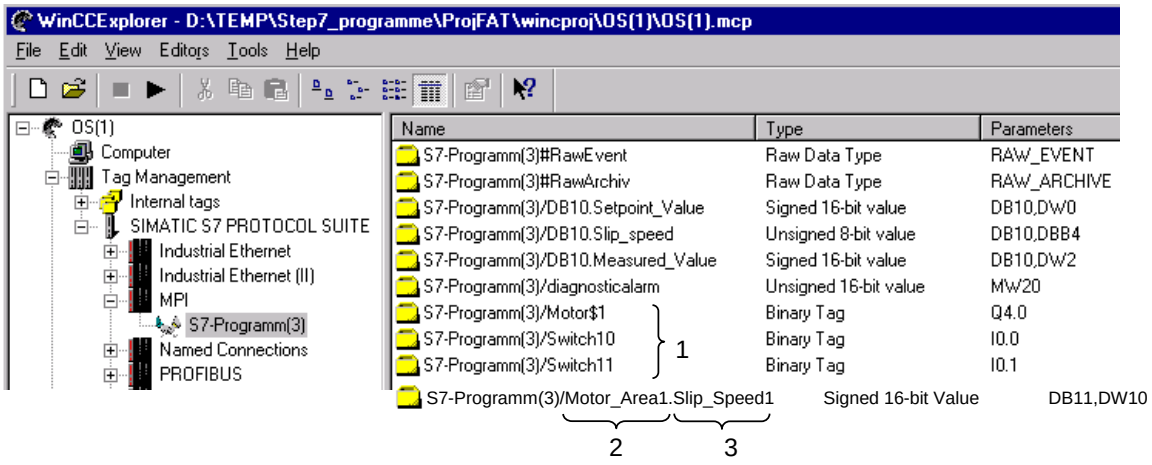


Transferring (Mapping) Data from STEP 7 to WinCC 2/2



Symbol names from DBs with the definition of Arrays are not displayed.
The DB is always displayed as absolute.

Mapped Symbols in the WinCC Explorer

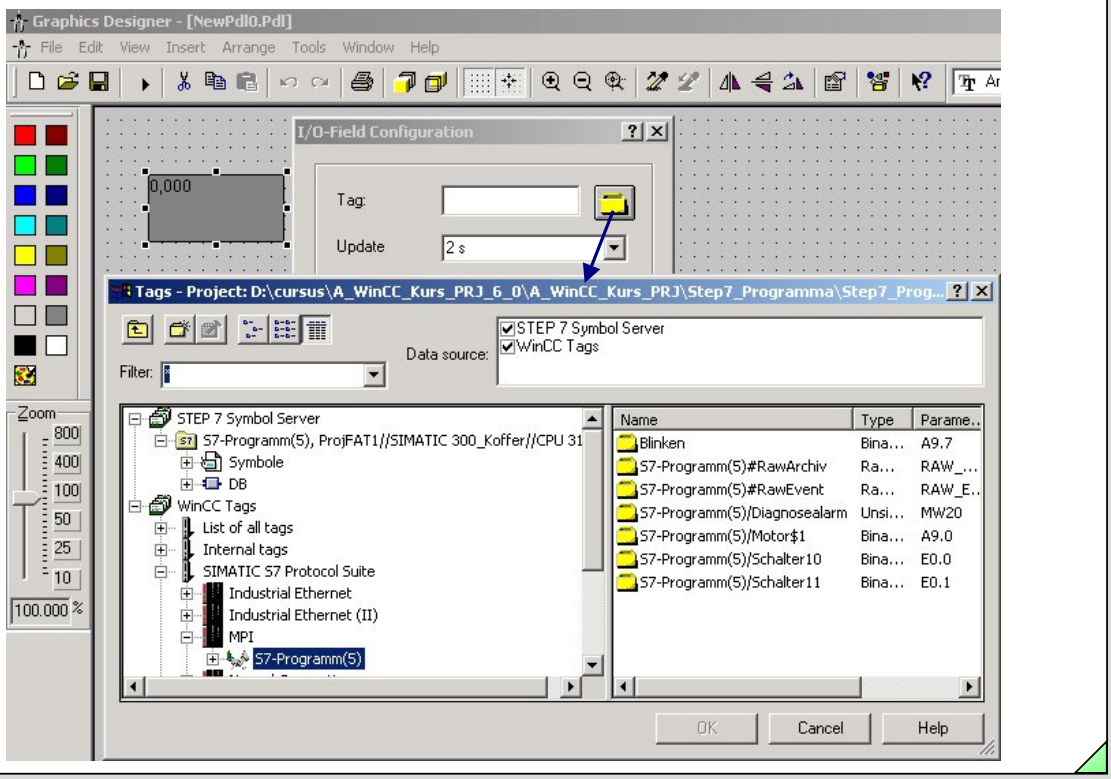


Date: 27.10.2003
File: a_dSwincc03e.54

A connection is required for mapping. This is automatically created.

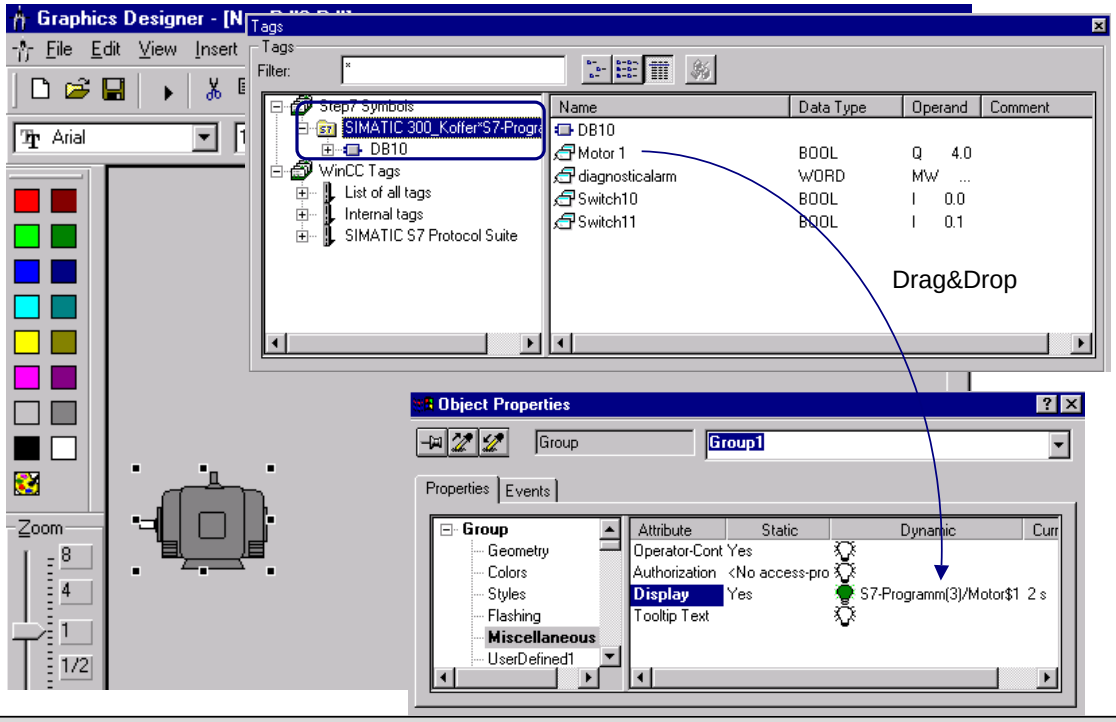
to 1 Symbol names from the symbol table for I, Q, M
to 2 Symbol names from the symbol table for data blocks
to 3 Tag names from the data block

Mapped Symbols for Tag Connection



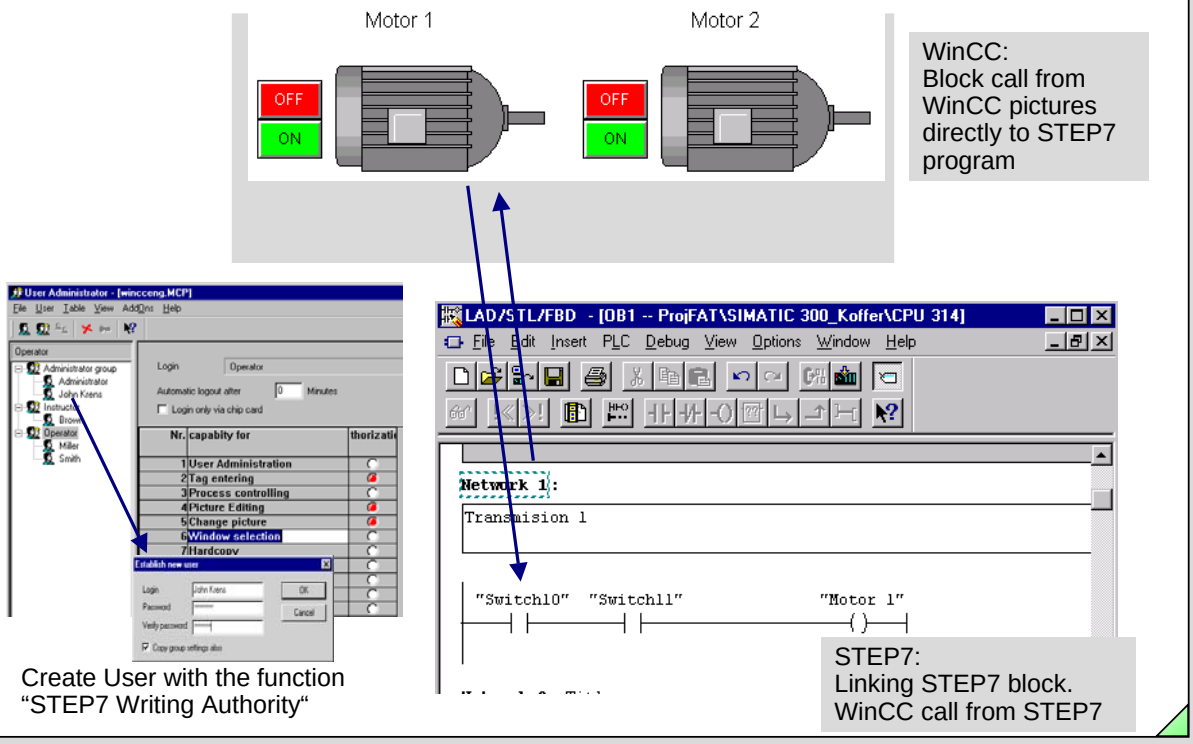
Date: 27.10.2003
File: a_dSwincc03e.55

Automatic Mapping in the Background



Date: 27.10.2003
File: a_dSwincc03e.56

TIA: Ladder Rung Jump with Writing Authority in STEP7



SIMATIC HMI

Date: 27.10.2003
File: a_dSwincc03e.57

SITRAIN Training for Automation and Drives

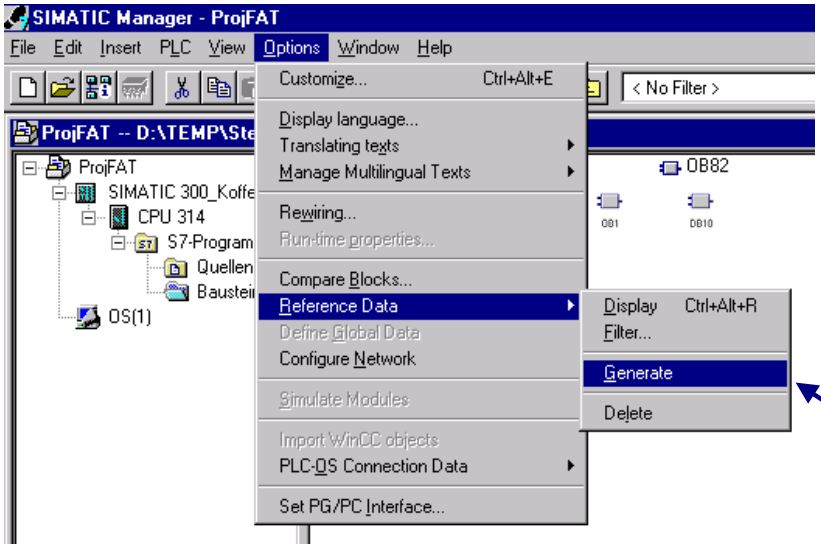
Objective

Define a ladder rung jump with writing authority in STEP7.
In the WinCC User Administrator, a separate level for the STEP7 writing authority is generated. The system then checks this level that is entered by the active user before the ladder rung jump is made and passes it as a flag to STEP7 (STEP7 V5.1). Depending on the status of this flag, the user has a read-only right or the authority to change STEP7 data.

Configuration

-In the User Administrator, create a function “STEP7 Writing Authority“.
-Call the Dynamic Wizard at a graphic object and assign parameters for the function “Ladder Rung Jump“.

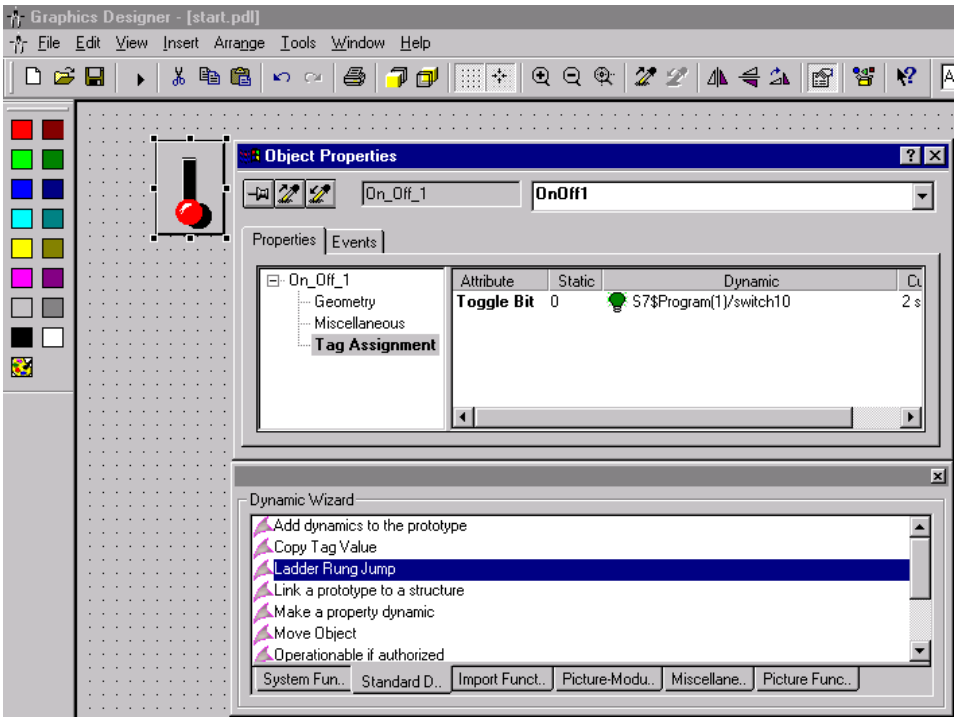
Generating Reference Data for Ladder Rung Jump



Necessary for ladder rung jump from WinCC

A SIMATIC (CPU) must be inserted in the Step7 program.

Ladder Rung Jump to STEP 7



For the ladder rung jump with the Dynamic Wizard, assign parameters for the right mouse button and accept the displayed properties.

The function can now be tested in Runtime.

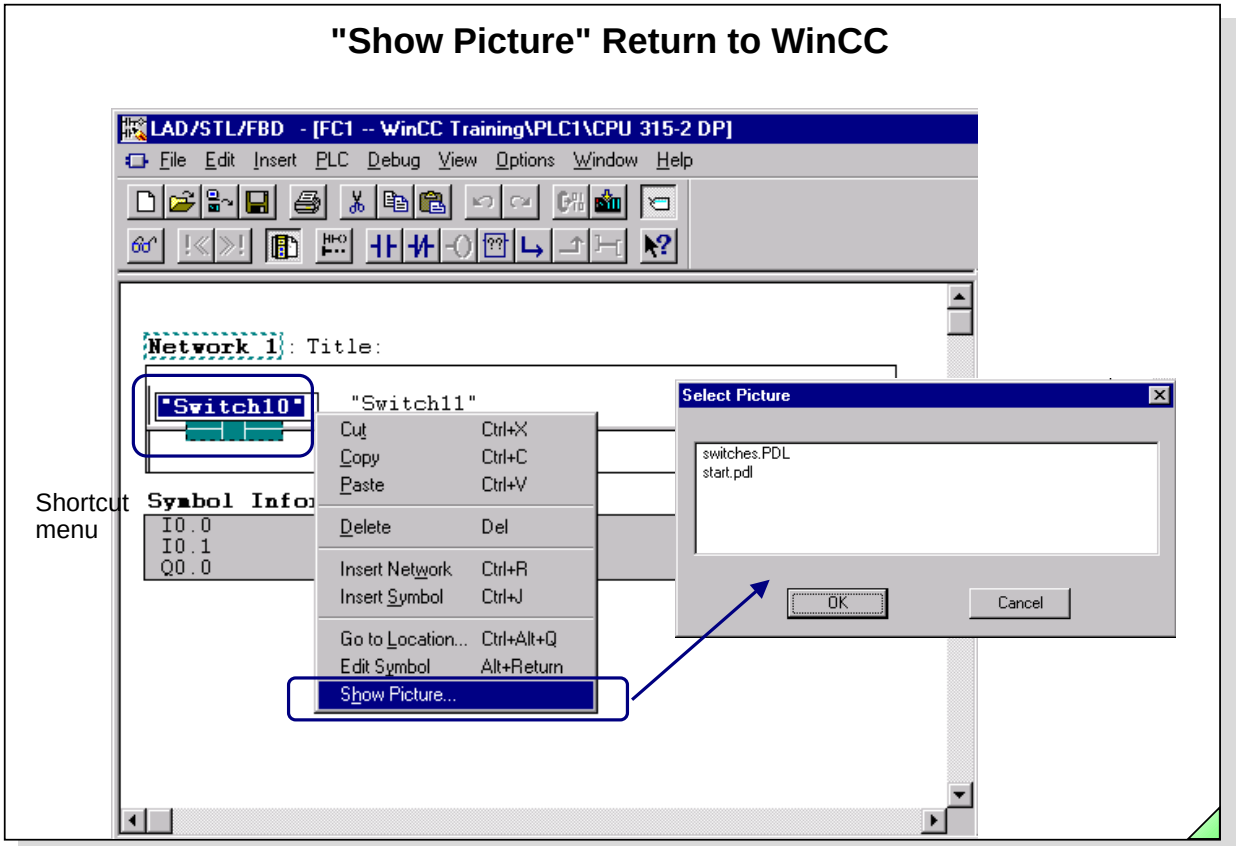
Note

If the call “Show Picture“ is made from S7 and you switch back to STEP 7 with <ALT><TAB>, you should not carry out the call “Show Picture“ again, since a correct screen design cannot otherwise be guaranteed.

If, after a ladder rung jump, you switch back to WinCC with <ALT><TAB> and then you navigate therein, then the new picture to be displayed after a return jump is the picture that was stored in the picture hierarchy.

The correct functioning of the ladder rung jump is only guaranteed if the data of the "Cross Reference" Editor correspond to the current project data.

"Show Picture" Return to WinCC

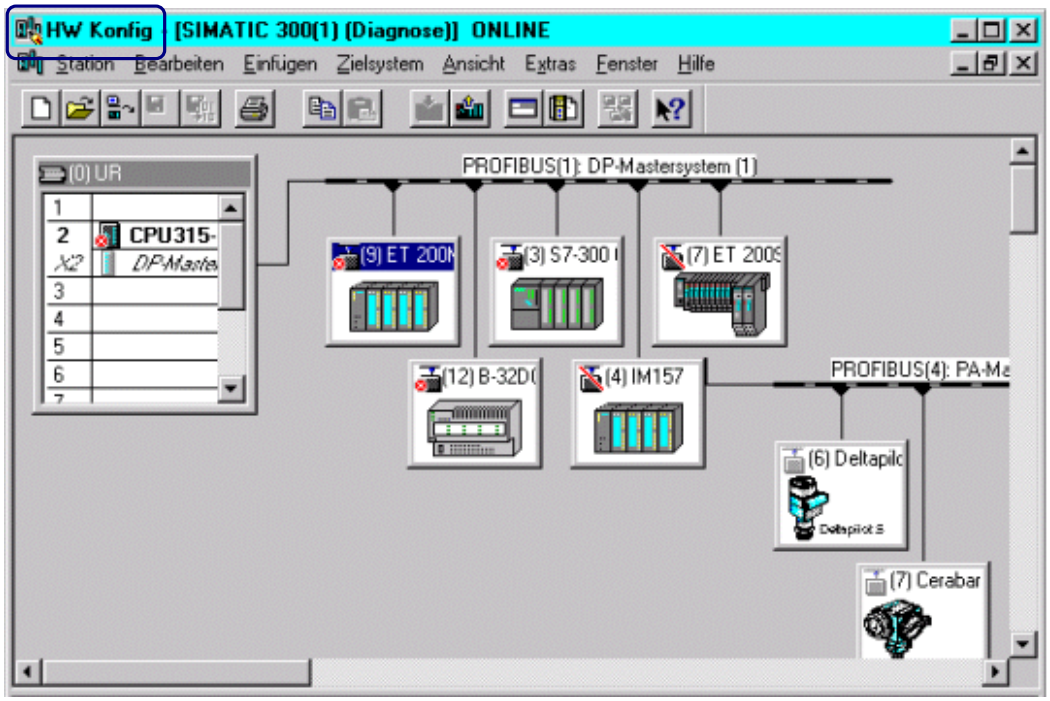


Shortcut menu

The function “Show Picture“ is possible in all three programming languages (as of STEP7 V5.1).

Call the Cross Reference in the WinCC Explorer and update it. After that, all pictures in which the tag “Switch 10“ was used are displayed in the STEP 7 program.

TIA Functionality, Hardware Diagnosis from WinCC Pictures

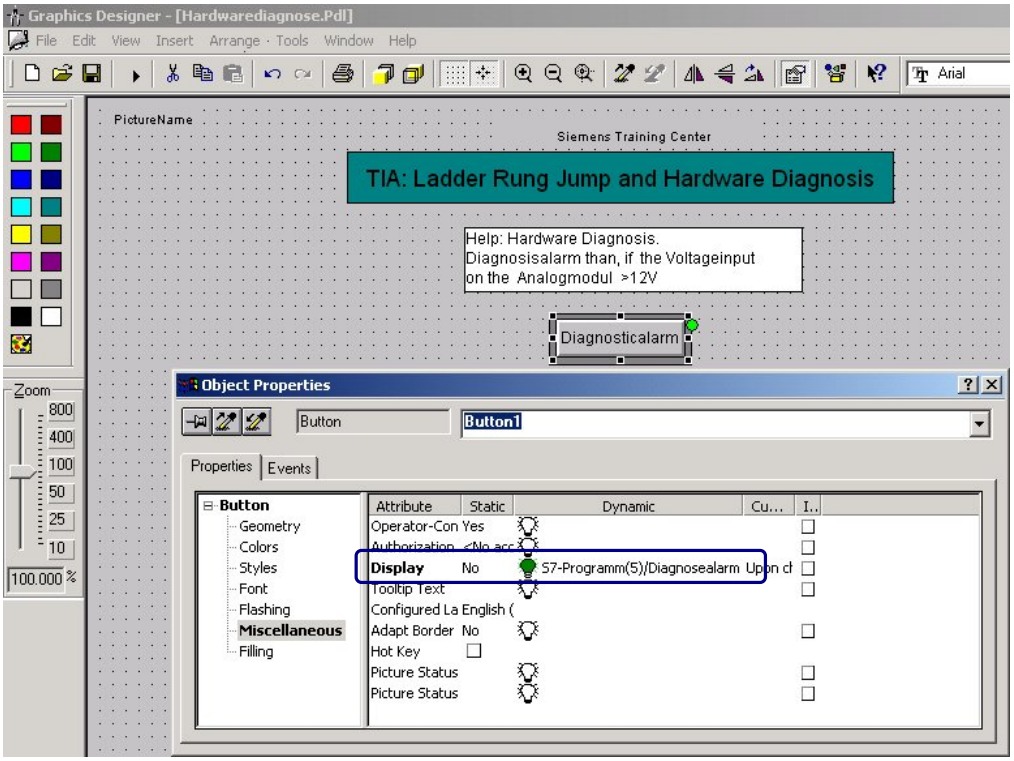


Date: 27.10.2003
File: a_dSwincc03e.61

The Hardware Diagnosis function enables you to start the STEP 7 Diagnosis directly from WinCC pictures.

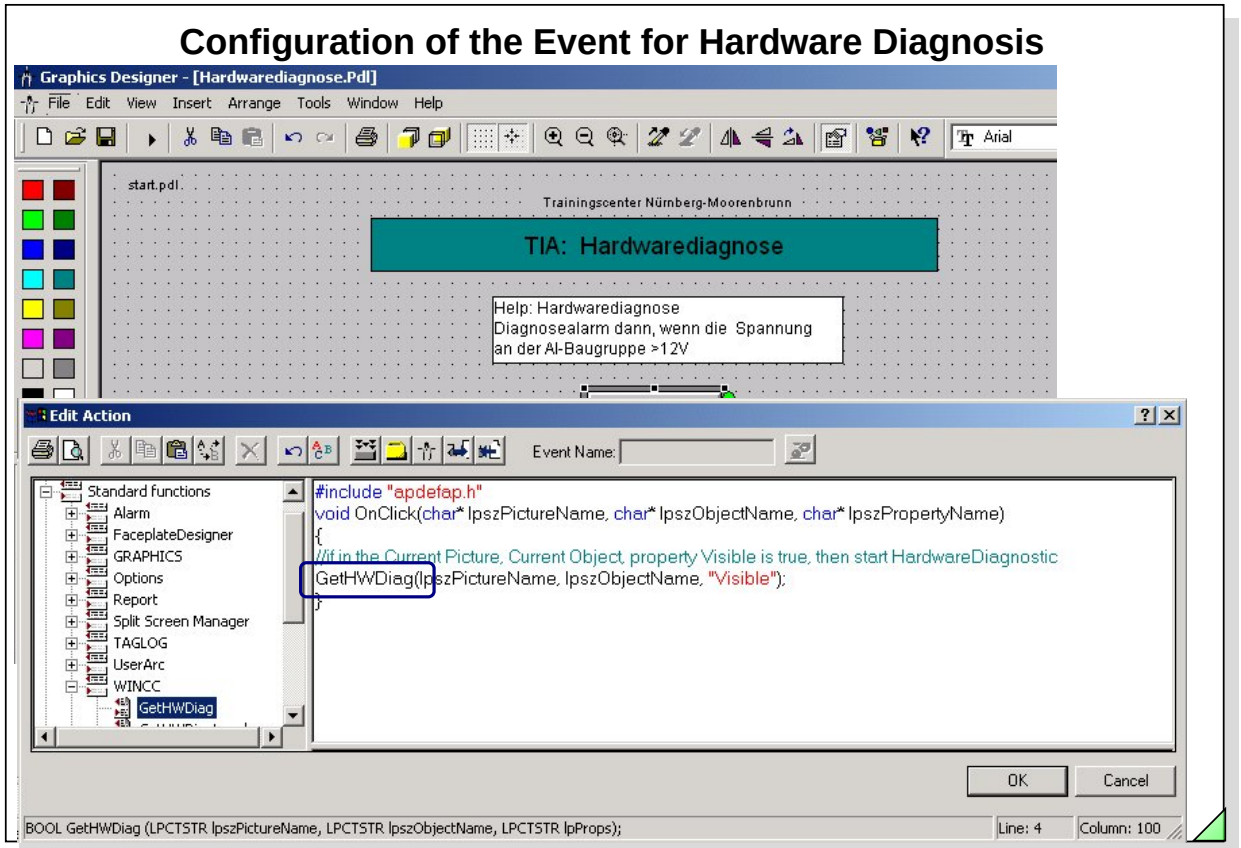
Depending on a configurable event, the STEP 7 “Diagnose Hardware” function is started for the associated controller.

Configuration of an Attribute for Hardware Diagnosis



A process tag has to be specified for the attribute.

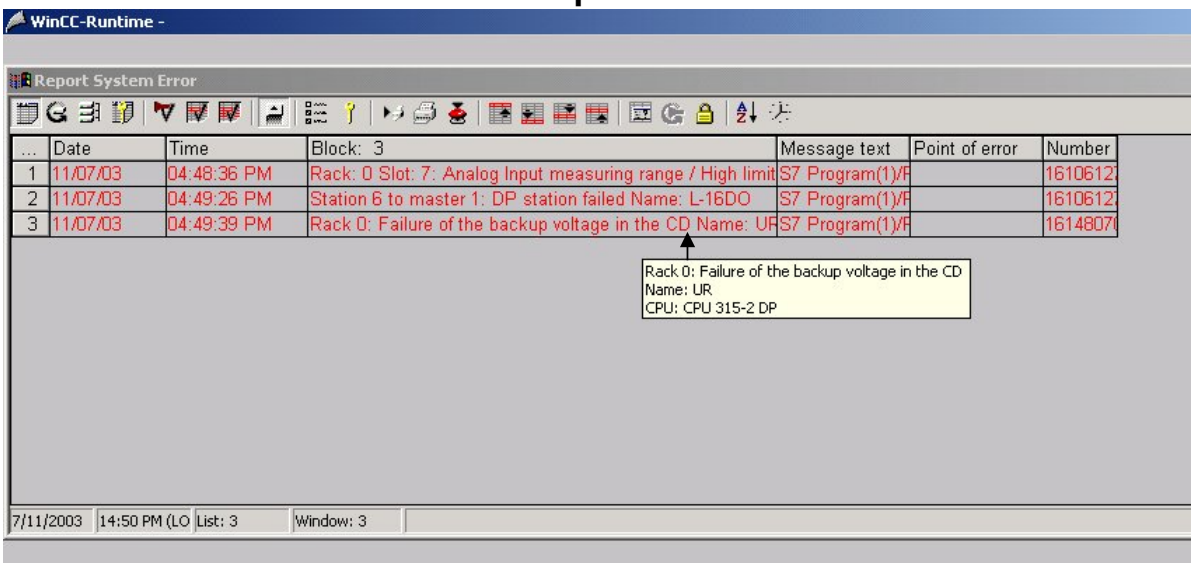
Configuration of the Event for Hardware Diagnosis



Configuration

No separate wizard currently exists for the jump to the Hardware Diagnosis. You can however, use the wizard for the ladder rung jump to configure the jump to the hardware diagnosis. The jump to the hardware diagnosis can then be configured in WinCC in an object generated in the Graphics Designer using the "Ladder Rung Jump" Dynamic Wizard. In a picture, insert the Windows object "Button", for example. Start the Dynamic Wizard and select the "Ladder Rung Jump" from the "Standard Dynamics".
The function call created by the Dynamic Wizard has to be adjusted so that it matches the one in the slide.

TIA Functionality, SIMATIC “Report System Error“ in WinCC Graphic Pictures



"Report System Error" supports the components of S7-300 stations, S7-400 stations, DP Slaves and WinAC, provided that they support functions such as diagnostic interrupt, insert/remove interrupt and channel-specific diagnosis.

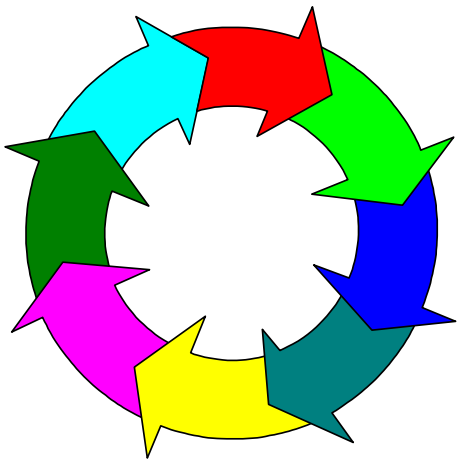
With the function "Report System Error" (>= Step7 V5.1 and >= WinCC V5.1), STEP 7 provides a user-friendly opportunity to display the diagnostic information provided by the components in the form of messages.

The blocks and message texts needed for this are automatically generated by STEP 7. All the user has to do is download the generated blocks into the CPU and transfer the texts to connected HMI devices.

Configuration

In HW Config, select the CPU for which you want to configure the system error reporting. Then choose the menu commands **Options > Report System Error**.

Graphic Screens (1)



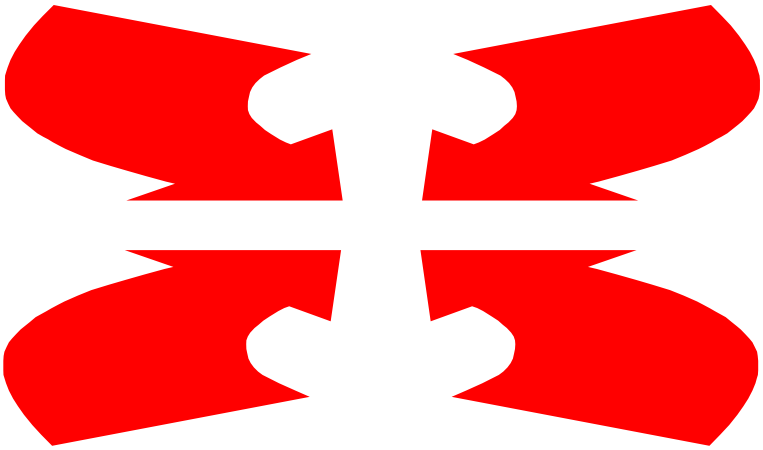
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.1

SITRAIN Training for Automation and Drives

Contents Page

Graphic Screens (2)

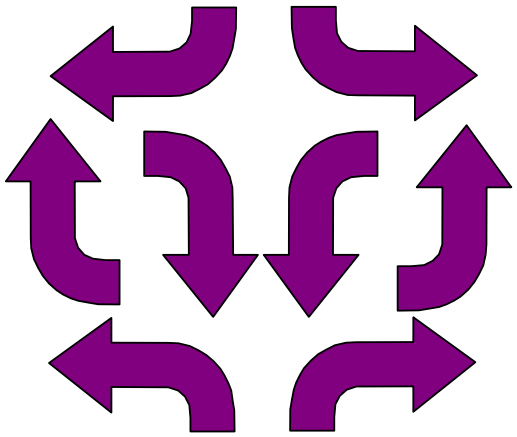


SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.2

SITRAIN Training for Automation and Drives

Graphic Screens (3)

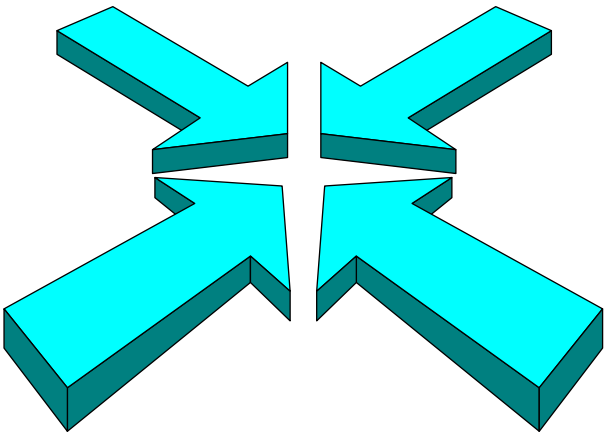


SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.3

SITRAIN Training for Automation and Drives

Graphic Screens (4)

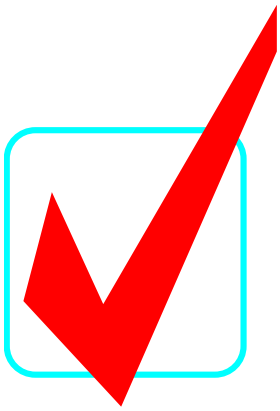


SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.4

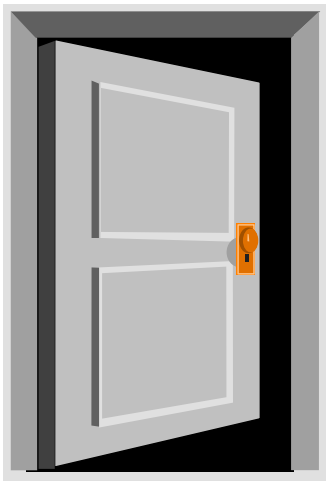
SITRAIN Training for Automation and Drives

Graphic Screens (5)



Date: 27.10.2003
File: SWINCC_04e.5

Objectives



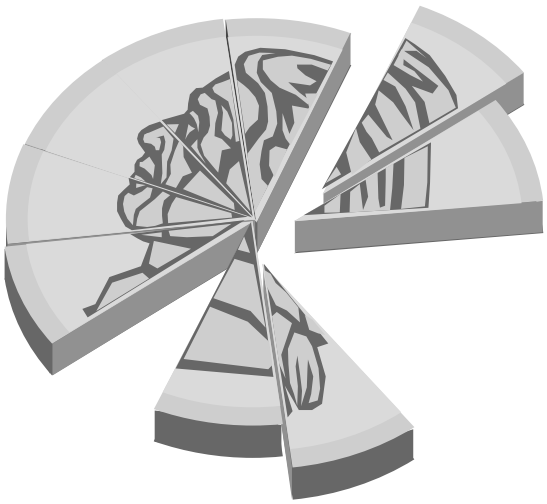
- Configuration Interface
- Creating Pictures
- Making Dynamic
- Library

Date: 27.10.2003
File: SWINCC_04e.6

Upon completion of this chapter, the participant should be able to:

- identify configuration relationships
- start WinCC and create a project
- describe the WinCC start procedure
- be comfortable with the WinCC Explorer
- create system flow diagrams with the Graphics Editor and carry out colorations
- configure graphic screens with an input/output value, bar graph, slider objects, button, mouse operation, function key operation, option group, check box, switching actions, value input checks, screen changes, help dialogs, and display changes with objects and attributes
- know how a configuration is activated with the Runtime module
- carry out the necessary settings for the start dialog
- create a project library for symbols
- use the objects in the global library
- configure password protection
- identify the basic elements of C programming
- use action configuring to make objects dynamic
- create an action library.

Preparation for Configuration (1)



- Process Pictures
- Static Part
- Dynamic Part
- Action Configuration

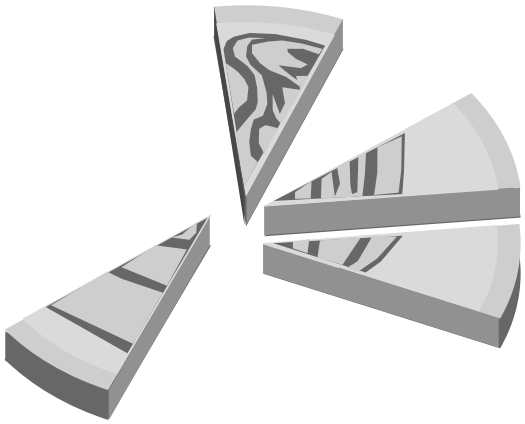
Date: 27.10.2003
File: SWINCC_04e.7

Configuration Preparation (1)

- Process Connection (see Chapter 3)
  - Specifying programmable logic controllers for the data exchange
  - Specifying the physical level for the data transfer
  - Specifying the number of input/output data for programmable logic controller communication
  - Specifying data areas for the individual send cycles

- Visualization (see Chapter 4)
  - Dividing the process into system screens
  - Specifying selection possibilities for the system screens
  - Specifying the static display per picture (text, line, circle, rectangle, color, etc.)
  - Specifying the active (dynamic) parts per picture (input/output value, bar graph output, switching operation, help box, trends, messages, etc.)
  - Specifying tags for the dynamics
  - Specifying the file name
  - Specifying operator devices (mouse, function keys, etc.)

Preparation for Configuration (2)



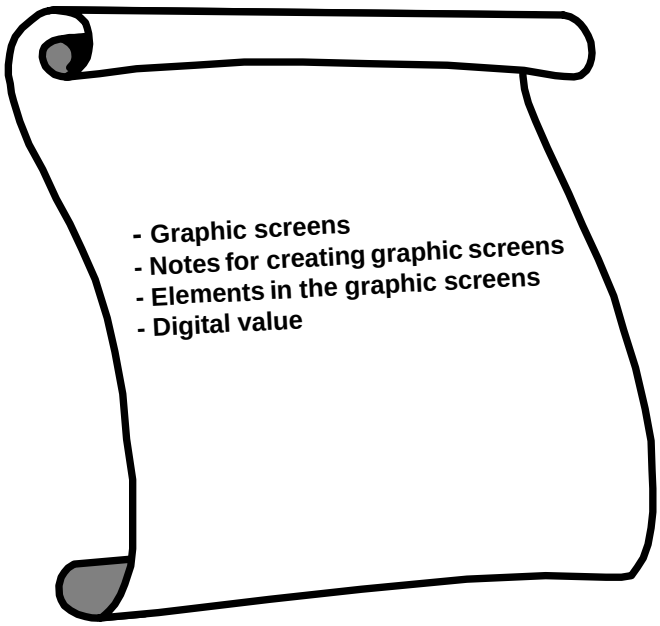
- Archiving Messages
- Acquiring Measurement Values
- Reporting
- Passwords

Date: 27.10.2003
File: SWINCC_04e.8

Configuration Preparation (2)

- Acquiring Measurement Values (see Chapters 6 and 7)
  - Specifying the measurement values to be archived (acquisition cycle, update cycle, length, etc.)
  - Specifying the type of display of measurement values (trends, tables, etc.)
- Archiving Messages (see Chapter 5)
  - Drafting messages (acknowledgement concept, assignment of message numbers, text, attributes, archiving concept, etc.)
- Reporting (see Chapter 8)
  - Specifying the system documentation
  - Drafting the report layout (static and dynamic component)
  - Specifying reporting components
  - Specifying which archives and process values the report accesses
  - Specifying the number of reports
- Password
  - (define) groups
  - (define) single users

Operating and Monitoring Terms



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.9

SITRAIN Training for Automation and Drives

Graphic Screens

Graphic screens are pictures you create freely yourself and that generally represent systems or system components. They make it possible to operate automated systems. The graphic screens are output in the graphic window. They consist of a static and a dynamic screen component. Both the static and the dynamic figure elements can be operated. Specific keysets belong to each graphic screen from which operations can also be triggered.

Notes for Creating Graphic Screens

- Several graphic screens can be called and displayed for configuring at the same time.
- The clipboard or drag-and-drop feature makes it possible to copy picture components from other graphic screens.
- The size of a graphic screen can be adapted to the monitor resolution.

Elements in the Graphic Screens

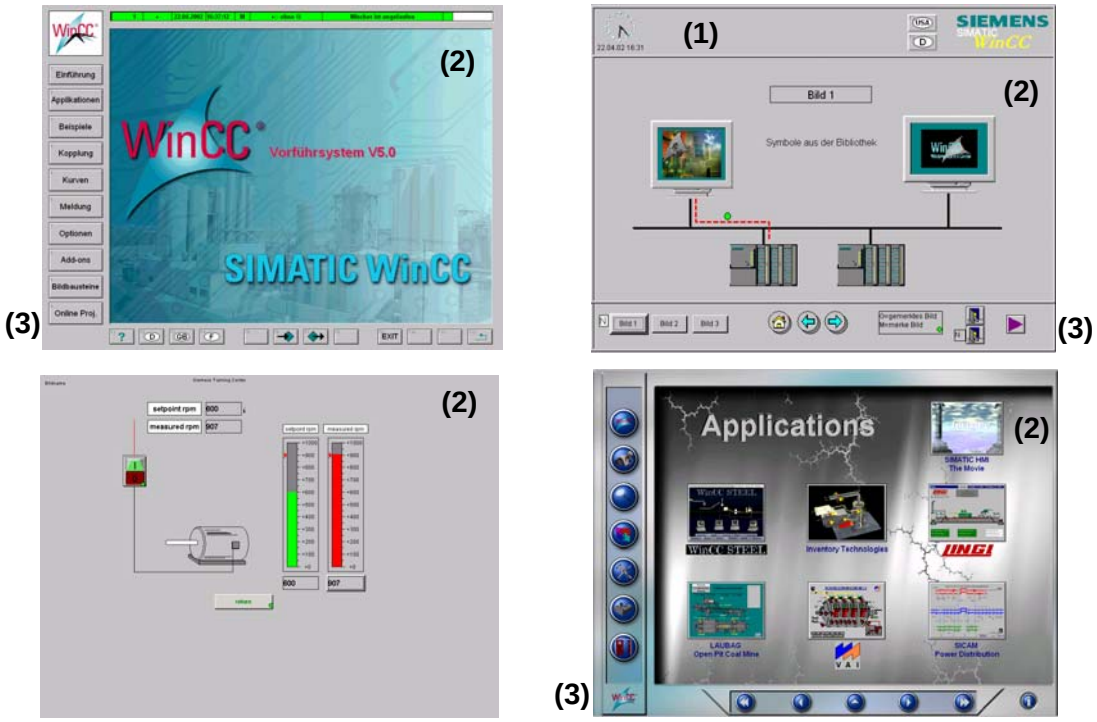
All the figure components that do not change either in form or color belong to the static picture component.

All picture elements that display process values or process states belong to the dynamic picture component. Among the dynamic picture components are digital values, bar graphs, output texts, attributes, and display changes.
Action configuration enables all objects (static or dynamic) to undergo changes in color, pattern changes, geometry, flashing, style, position, and level in process control.

Digital Values

Digital values display values from the process in the form of numbers, or they are used as input fields to change setpoints directly in the picture. The display of the value can be changed with attributes and formats, depending on the value that has to be displayed.

Examples of Dividing a Screen



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.10

SITRAIN Training for Automation and Drives

Dividing a Screen

In principle, each individual screen division can be configured. There are four examples of a screen division displayed in the slide above.

Overview Area (1)

The overview area contains generally applicable information, such as the date, time of day, messages, range selection operations, and system messages. The overview area is displayed in all pictures.

Work Area (2)

The work area displays the process events.

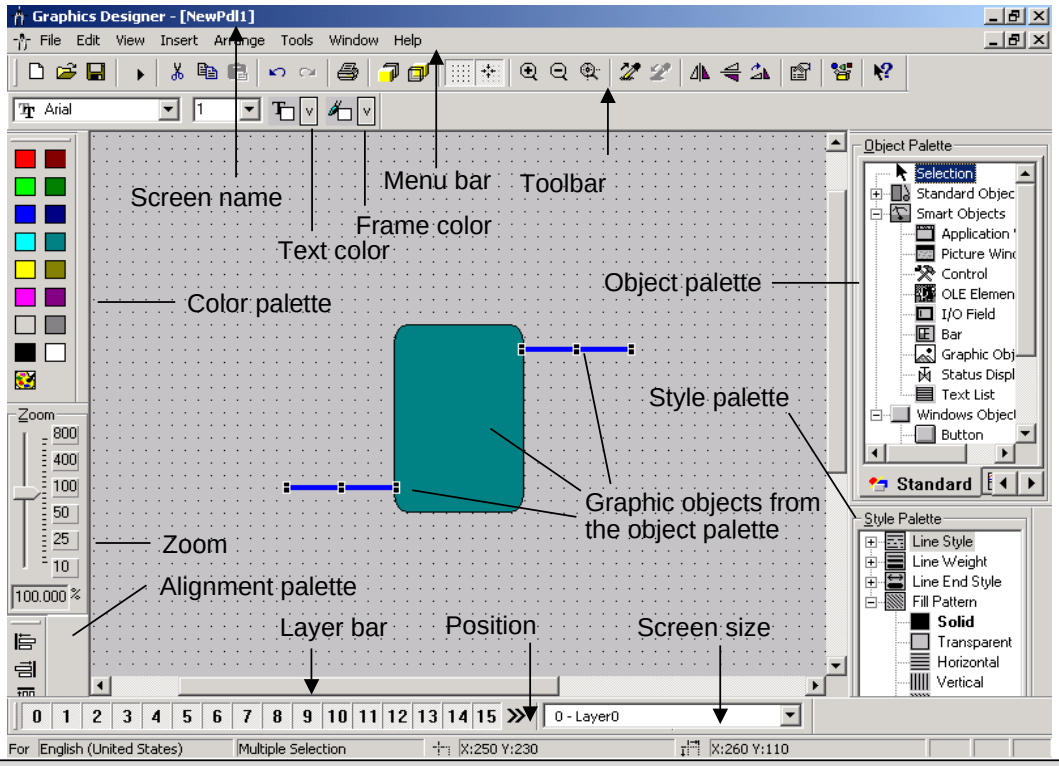
Keyset Area(3)

The keyset area displays buttons that can be operated, such as switching the workarea, calling information, and switching the keyset. The buttons can be used independent of the selected workarea. The mouse and/or function keys support operation.

Note

The screen division in the slide above, first line, right, can be configured with Basic Process Control. BPC is delivered with WinCC. It must be installed from the CD separately. BPC does not require a license.

Graphics Editor Configuration Interface



Date: 27.10.2003
File: SWINCC_04e.11

Objective

The Graphics Editor is used to carry out the necessary screen (picture) creation (static and dynamic components) according to the task definition.

Activating/ Deactivating a Palette

1. In the menu bar, click on **"View".**
2. In the drop-down menu, click on **"Toolbar..."**

Screen Size

You determine the size of the screen and set the size on the graphics driver you installed. The default setting is 800 x 600 pixels.

Setting the Screen Size

1. Call the shortcut menu on the empty screen area.
2. Select "Properties..." from the shortcut menu.
3. Use the "Properties..." tab in the "Object Properties" window to select "Geometry."
4. Double click on the "Height" or "Width" attribute to enter a new value in the displayed window.

Note

The picture selection time can be improved if the static objects are loaded via an *.emf file.

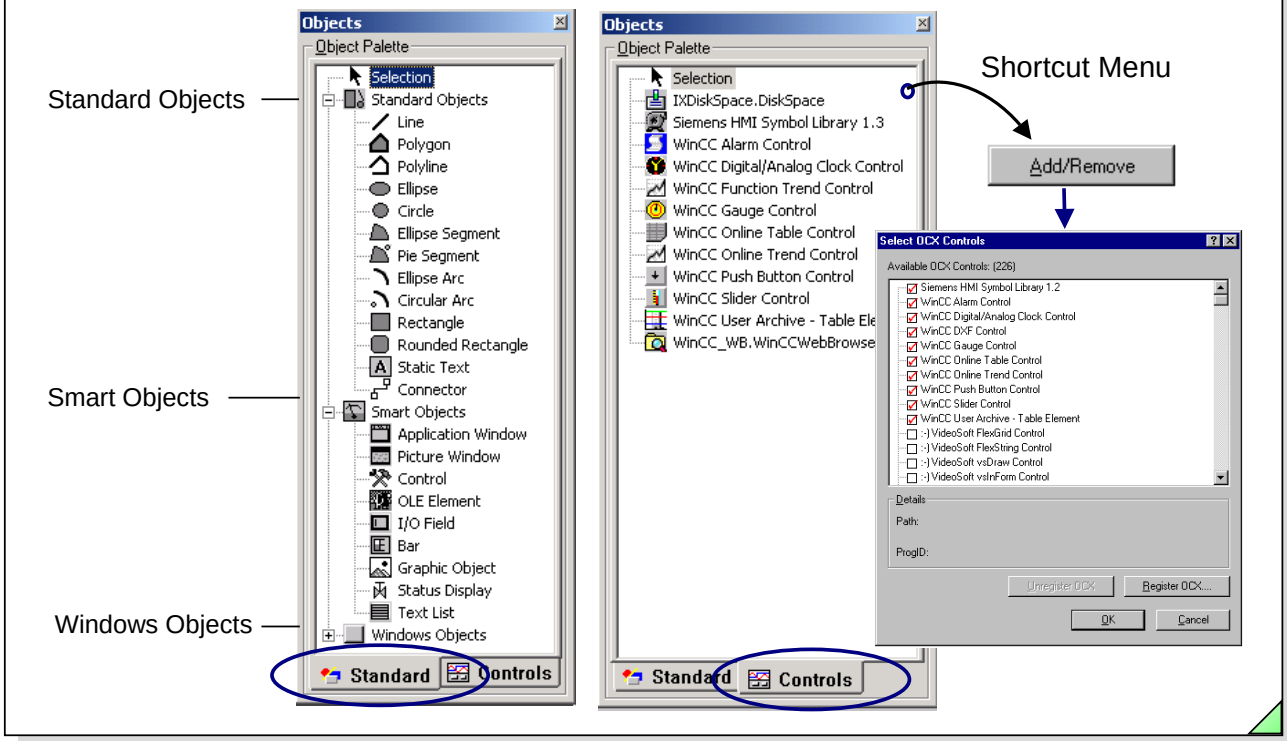
For this you select all static objects (they must all be located in the same layer) and save them in the GraCS directory by clicking on **File** in the menu bar and then **Export** in the drop-down menu.

Then, the object palette is used to edit the graphic object and the exported *.emf file is linked with the graphic object.

Toolbar

The toolbar can be configured by pressing ALT + double-clicking with the left mouse button. Please take into consideration the settings under the menu bar **Options and Settings in the drop down menu, Menu/Toolbar tab.**

Graphic Object Palette (1)



Configuring Objects

You use the left mouse button to select an object in the object palette. Click on the mouse to place the object on the configuration interface. Additional editing possibilities are described in the Help function.

Standard Objects

The standard objects are composed of graphic objects and static text. The connector is a line object whose ends can be connected with the connection points of other objects.

Smart Objects

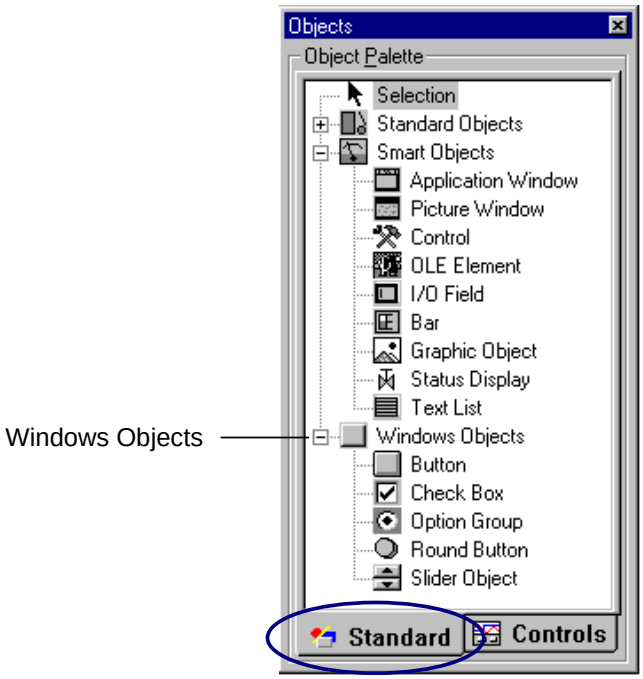
Application windows are objects that are supplied by the report system, the diagnostics system, or even external applications. The external properties (position, size, and external attributes) are configured in the Graphics Designer and are transferred to the external application at run time. The external application opens the application window and takes care of the display and operation.

Picture windows are objects that are configured in their position, size, and their other properties and that can be made dynamic. An important property is, for example, the reference to another screen (picture) that is to be displayed in the picture window. The window contents can be dynamically changed through dynamic changes in the "picture name" attribute at run time.

Controls are used to implement Windows elements (such as a slider or an analog clock). An OLE control has attributes displayed in the "Object Properties" window in the "Properties" tab and which can also be edited in this window.

OLE elements can be integrated in a graphics screen. You double click (for example, a Microsoft Paint screen) to edit this object with the respective OLE application. Other object types, such as audio or video objects, are played when you double click on them. Their source application does not open up.

Graphic Object Palette (2)



I/O Fields can be used as input field, output field, or as combined I/O field.

Binary, hexadecimal, decimal, or string data formats are possible. A specification of limit values, a hidden entry, or a transfer when an entry is complete is possible.

Bar (Graphs) display process values in an analog form. An area displays the current value. A typical bar graph application is displaying the level in a container. Changes in color can indicate limit value violations.

Graphic Objects are used to integrate external Graphic objects **(.wmf, .emf, .bmp, .gif, .jpg, .jpeg, .dib**) in a picture in the Graphics Designer.

Status Display for dynamic display changes of Graphic Objects **(.wmf, .emf, .bmp, .gif, .jpg, .jpeg, .dib**).

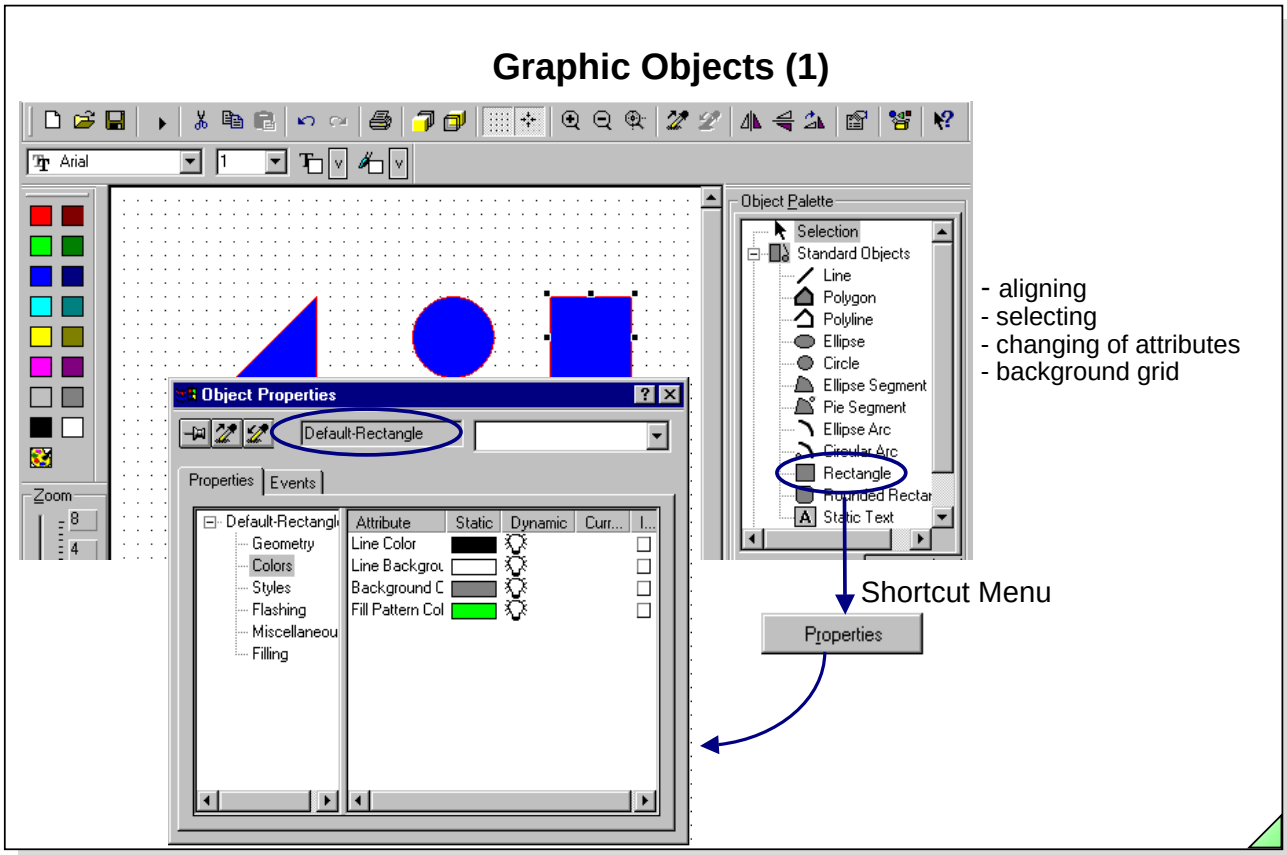
EMF/WMF Objects (Enhanced and Windows Meta File) are generated elsewhere and are integrated in graphics screens. Integrate EMF or WMF objects with:

1. In the menu bar, click on **"Insert"** .
2. In the drop-down menu, click on **"Import..."**

Text list can be used as input, output or an in-and output text list.

Windows Objects

Buttons are used to operate process events.
Check Boxes are needed for multiple selection of operations.
Option Groups are similar to the check box, but only one operation can be executed.
Round Buttons are used to operate process events.
Sliders are used as linear regulators for analog process value adjustment.



Aligning Objects

This function can be handled via the alignment palette.

Tip

If you turn on the grid points before you begin configuring a graphic, you can align at the same time as you move objects.

Selecting Objects

1. You use the left mouse button to select an object.
2. When you hold down the left mouse button and drag across several objects, all objects in the rectangular shaped frame are selected (lasso function).
3. When you hold down the Shift key and select individual objects with the left mouse button, objects can be "collected" .

This function is used for deleting, aligning, duplicating, moving, and object editing. There are other settings:

1. In the menu bar, click on the **"Tools"** menu.
2. In the drop-down menu, click on **"Settings...“**

Changing Attributes

Select an object and use the right mouse button to call the shortcut menu. All objects properties can be changed under "Properties". This way you have your own defaults which can be stored in a **Default.pdd** file. In the menu bar, click on **Tools** and in the drop-down menu, click on **Settings; Default Object Configuration tab**.

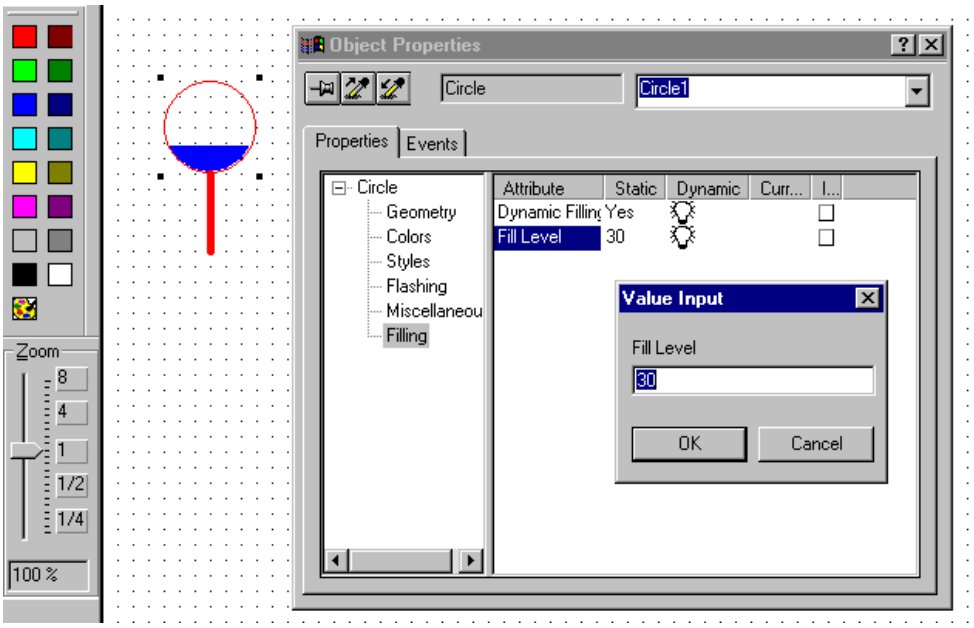
Changing an Object’s Default Settings

1. Call the "Object Properties" window.
2. Select an object in the object palette.
3. Set the attributes you want and drag the object onto the screen.

Background Grid

1. In the menu bar, click on the **"Tools"** menu.
2. In the drop-down menu, click on **"Settings...,"** where you enter the settings for the background grid, width, and height.

Graphic Objectes (2)



-deleting

- changing

- shifting

- duplicating

Deleting Objects

Select objects and remove the object from the configuration interface by
1. In the menu bar, click on the **"Edit"** menu.
2. In the drop-down menu, click on **"Delete"** .

Tip

It is faster if you use the DEL key.

Changing the Object Display

Depending on the object you select, you can change the size, width, length, and angle. You select an object, hold down the left mouse button, and shift the configuration nodes in the direction you want.

Changing Object Properties via the Shortcut Menu

1. After you select an object, you use the right mouse button to call the shortcut menu.
2. Use "Properties" to call the "Object Properties" window. You can now change the object properties.

Object Properties

1. In the menu bar, click on the **"View"** menu.
2. In the drop-down menu, click on **"Properties..."** to call the object properties window.

Moving Objects

Moving takes place with the drag-and-drop feature. You select an object and you move the object while holding down the left mouse button.

Tip

1. Select an object and use the cursor keys to position it pixel by pixel.
2. When you depress the Shift and Cursor keys simultaneously, you can shift the object across a larger area.

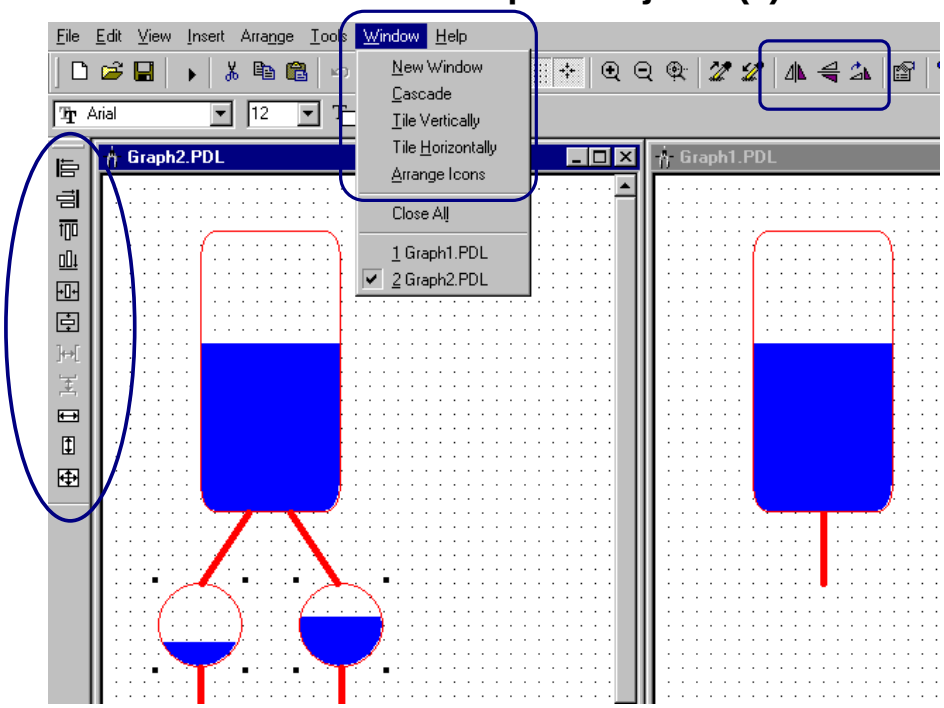
Duplicating Objects

Select an object and create a duplicate by
1. In the menu bar, click on **"Edit".**
2. In the drop-down menu, click on **"Duplicate"** .

Tip

It is faster when you hold down the Ctrl key and use the drag-and-drop feature at the same time.

Graphic Objects (3)



- rotate
- mirror
- arrange

Rotating

Rotating objects in a circle

1. In the menu bar, click on **"Arrange"** .
2. In the drop-down menu, click on **"Rotate".** Or, use the button in the toolbar.

Mirroring

Mirroring objects on the horizontal or vertical axis

1. In the menu bar, click on **"Arrange"**.
2. In the drop-down menu, click on **"Mirror"**. Or, use the button in the toolbar.

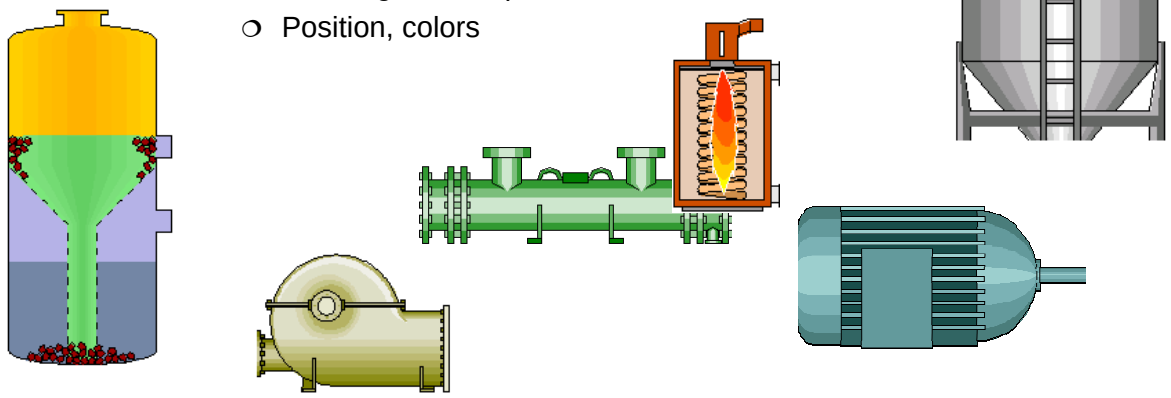
Arranging Windows

If two windows are arranged next to each other, you can use the drag-and-drop feature to move an object.
Settings for arranging windows:

1. In the menu bar, click on **"Window"**.
2. In the drop-down menu, click on **"Tile Vertically"**, for example**.**

Symbol Library ´Siemens HMI Symbol Library 1.3´ 1/2

- New symbol library ´Siemens HMI Symbol Library 1.3´
- They are not purely graphic objects
- rather Controls with dynamic Properties:
  - Appearance
  - Flashing mode, speed
  - Position, colors



SIMATIC HMI

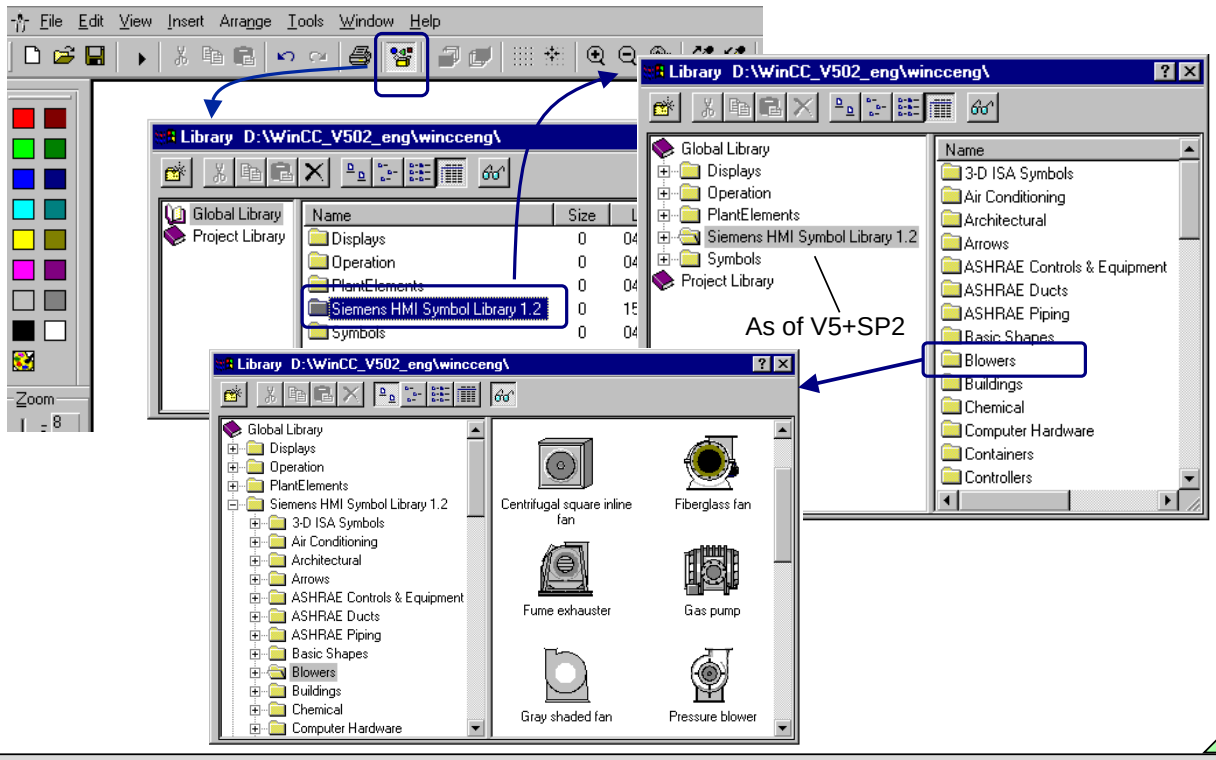
Date: 27.10.2003
File: SWINCC_04e.17

SITRAIN Training for Automation and Drives

Library for Symbols

You can activate the library function using the menu option **View** and the drop down menu **Library** or by clicking on the icon. A differentiation is made between the global library and the project library. Customized symbols can be stored in both areas.
The global library contains the default symbols with 2D representation and the expanded library (HMI Symbol Library 1.3, as of >=WinCC V5.0+SP2) with 3D representation.

Symbol Library ´Siemens HMI Symbol Library 1.3´ 2/2

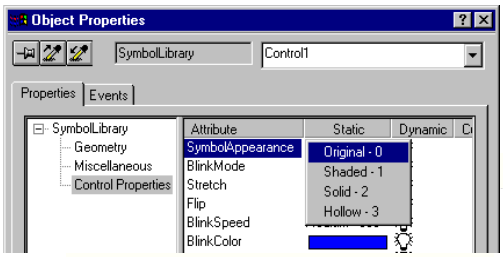


Date: 27.10.2003
File: SWINCC_04e.18

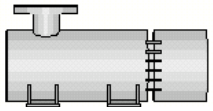
Library

As of >=WinCC V5.0+SP2, you have an extensive HMI Symbol Library 1.3. The individual symbols consist of controls with dynamic properties.

Control Properties

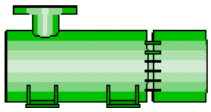


- **Original - 0** (0):



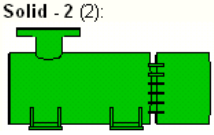
The surface color is gray

- **Shaded - 1** (1):



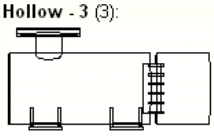
The surface color corresponds to the setting of the Foreground Color attribute

- **Solid - 2** (2):



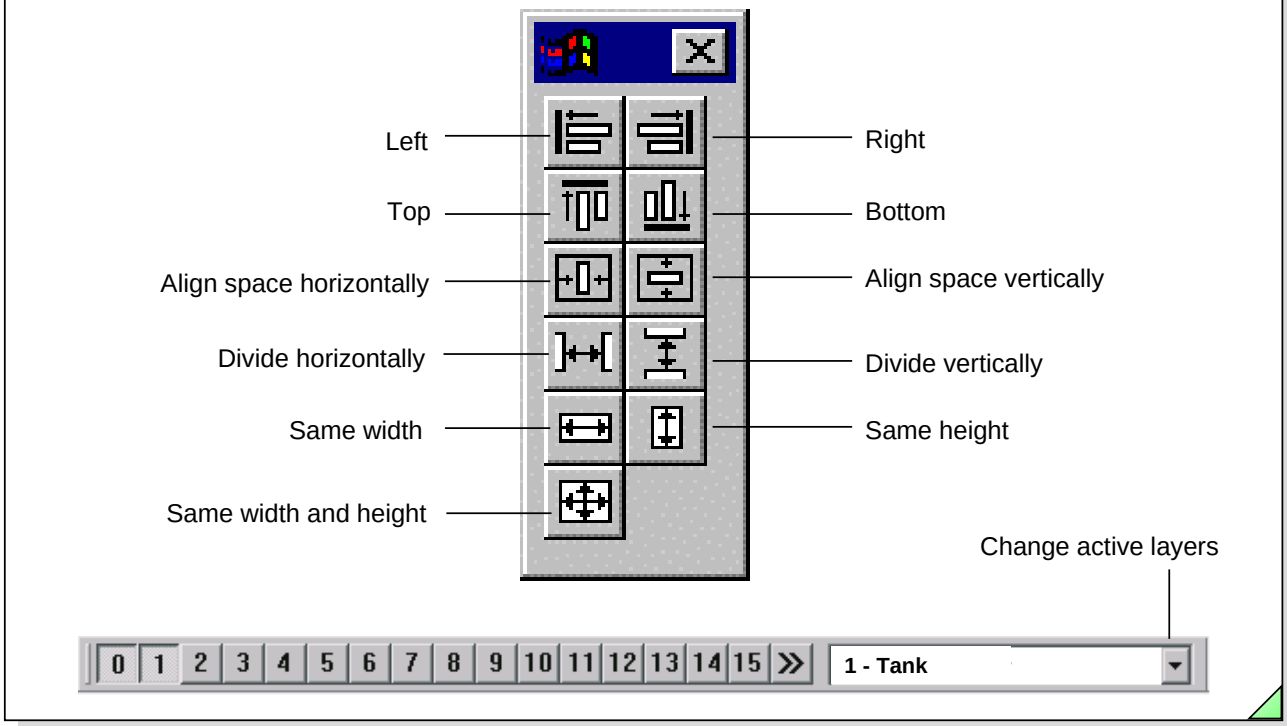
The surface color corresponds to the setting of the Foreground Color attribute

- **Hollow - 3** (3):



The surface is transparent

Alignment Palette and Layer Palette



Alignment Palette

With the alignment palette functions, you can edit several objects at the same time. You can also activate these functions with
1. In the menu bar, click on the **Arrange** menu
2. In the drop-down menu, click on **Align**.
Prerequisite: At least two objects have to be selected.

Layers Palette

To simplify the editing of individual objects in complex process pictures, the Graphics Designer enables you to do the work in layers. In this way, the contents of a process picture can be divided in up to 32 layers. These layers can be individually shown and hidden. In the default setting, all layers are visible. The active layer in the example is Layer 1 and cannot be switched off. When you press the buttons, the visible layers are displayed. You can also toggle the layers in RT mode.

Renaming Layers

You can rename the layers with
1. In the menu bar, click on the **Tools** menu
2. In the drop-down menu, click on **Settings…**
3. In the "Visible Layers" tab make the change.
Double-click on a layer, and you can enter a new name, such as Compressor.

Changing the Layer Assignment

The assignment of an object to a layer can be changed in the "Object Properties" window.

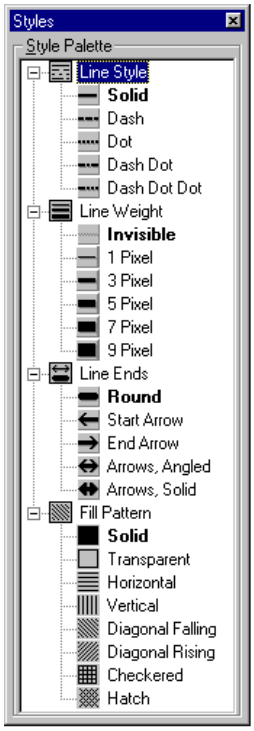
Style and Tag Palette

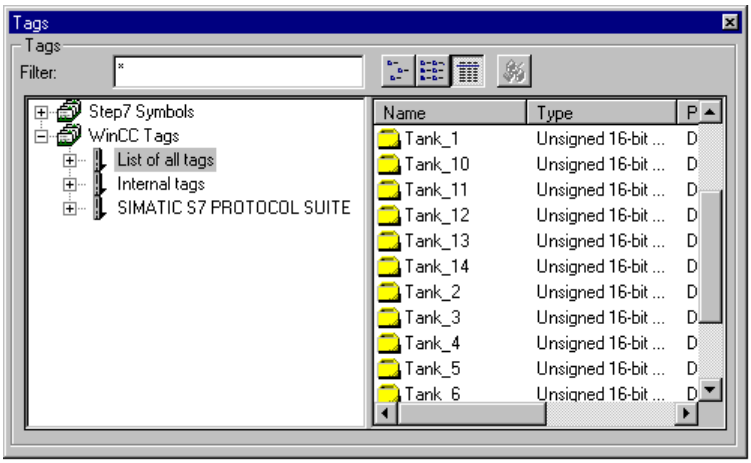
Line Style

Line Width

Line Ends

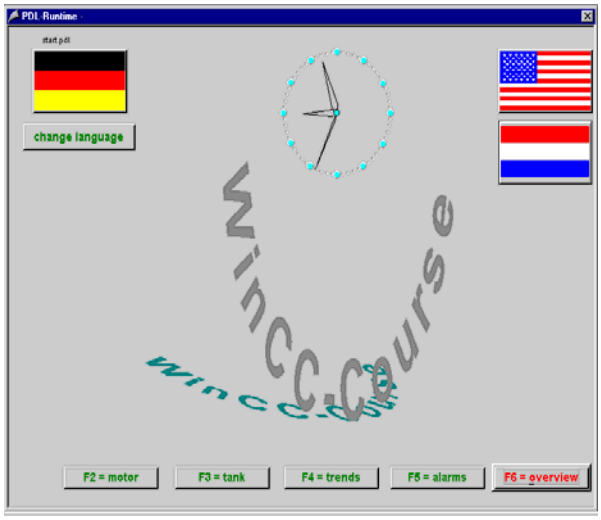
Fill Pattern

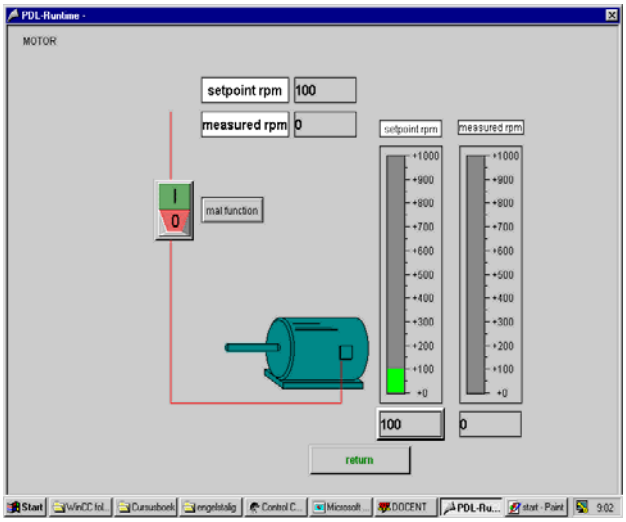




Date: 27.10.2003
File: SWINCC_04e.20

My First Project





SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.21

SITRAIN Training for Automation and Drives

Objective

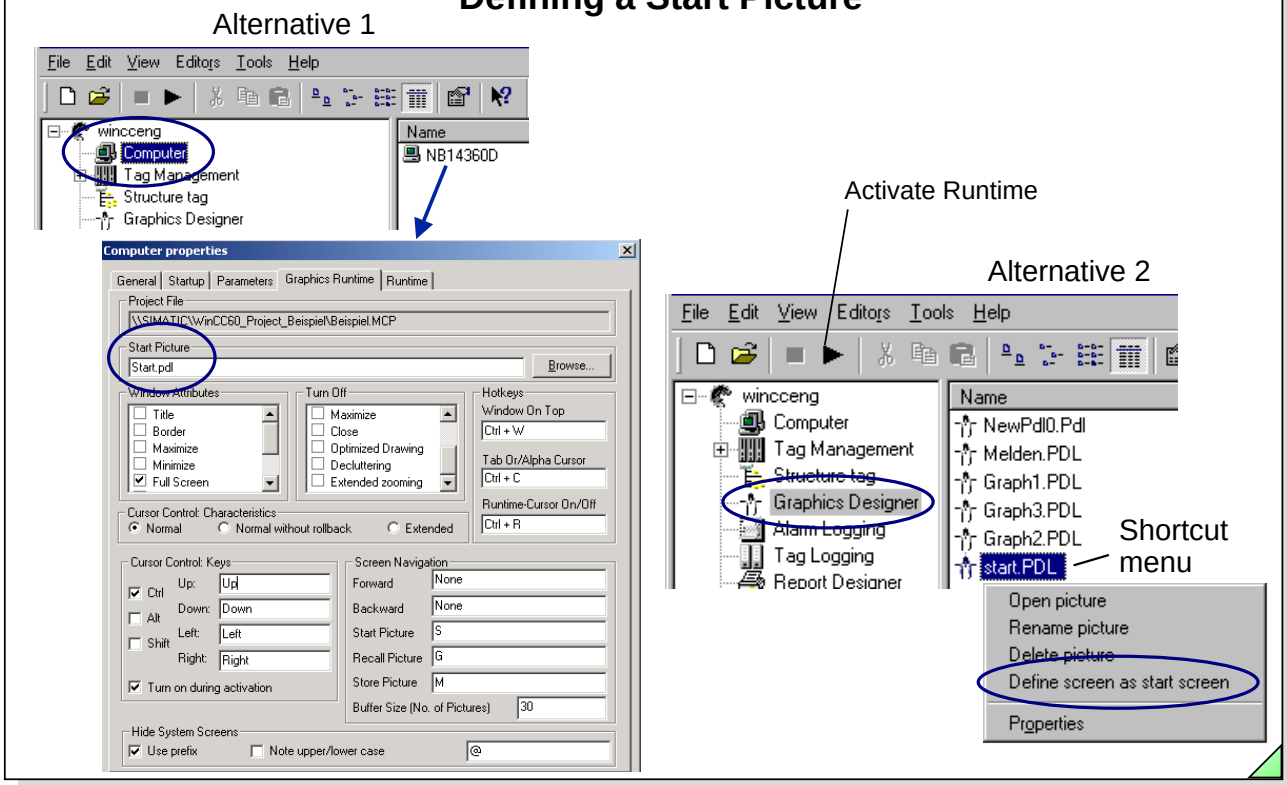
You will now configure a mini-project in small steps. The mini-project contains the static screen elements and the most important elements to be made dynamic. The following pages contain configuration solutions that take various WinCC philosophies into consideration. After adaptation (of such items as process tags, texts, and sizes) to your requirements, the solutions can be used in your configuration.

Structure

The configuration solutions are divided into:

- Digital input and digital output
- Color change (attribute change)
- Switching operations
- Value tests
- Help possibilities
- Miscellaneous

Defining a Start Picture



Date: 27.10.2003
File: SWINCC_04e.22

General

To activate the Runtime module, a start picture must be defined for each project. The start picture is the basis for all pictures from which tree structures can be configured for operating philosophies.
Each project must provide an operating capability to end the RUNTIME mode, so that configuration changes can be carried out, especially if the Windows operations are disabled.

Defining the Start Picture

Refer to the slide above.

Activating Runtime

1. Call the WinCC Explorer.
2. In the menu bar, click on the **"File"** menu.
3. In the drop-down menu, click on **"Activate"** .

Starting Runtime Automatically

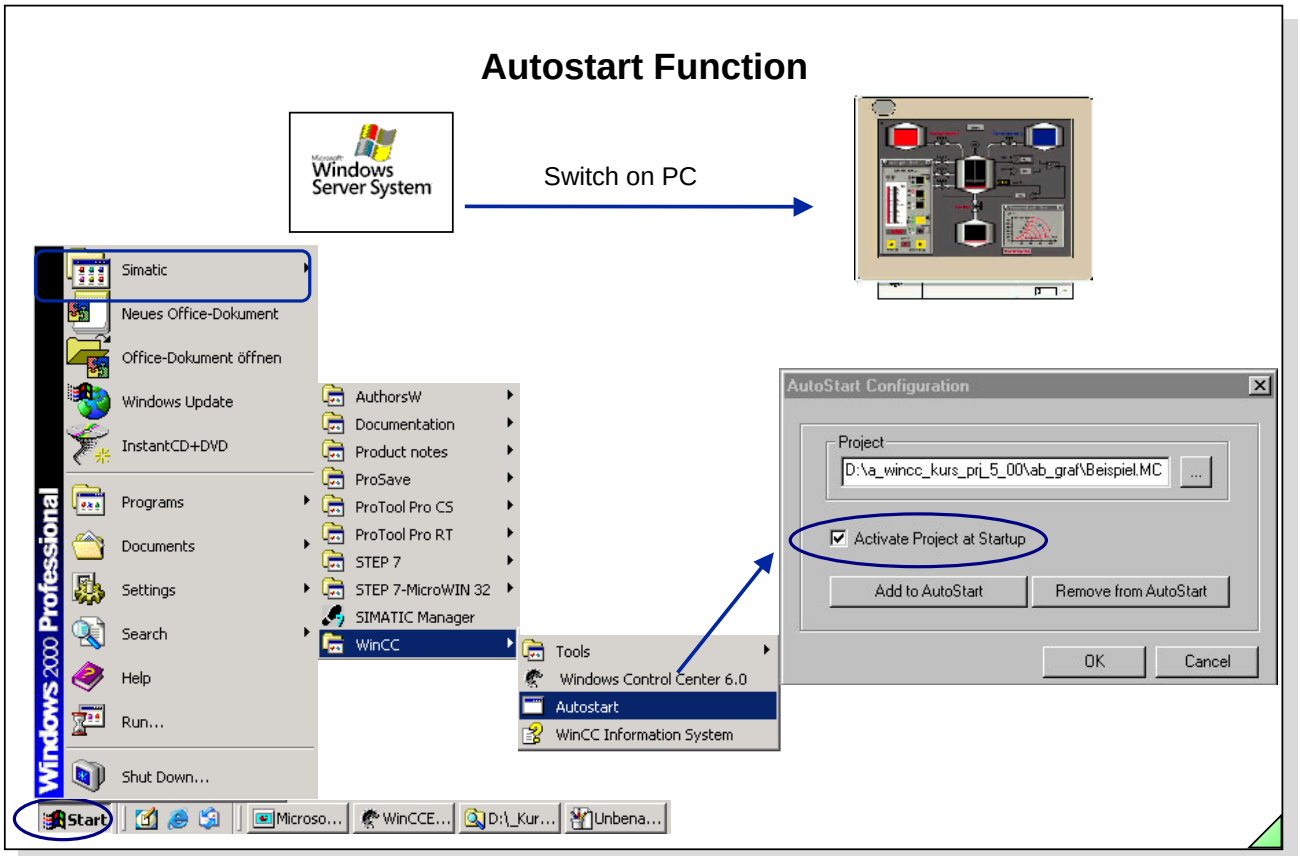
See next page.

Starting the WinCC Explorer

Create a link on the Desktop to the WinCCExplorer.exe (>=V5).

Exercise 1

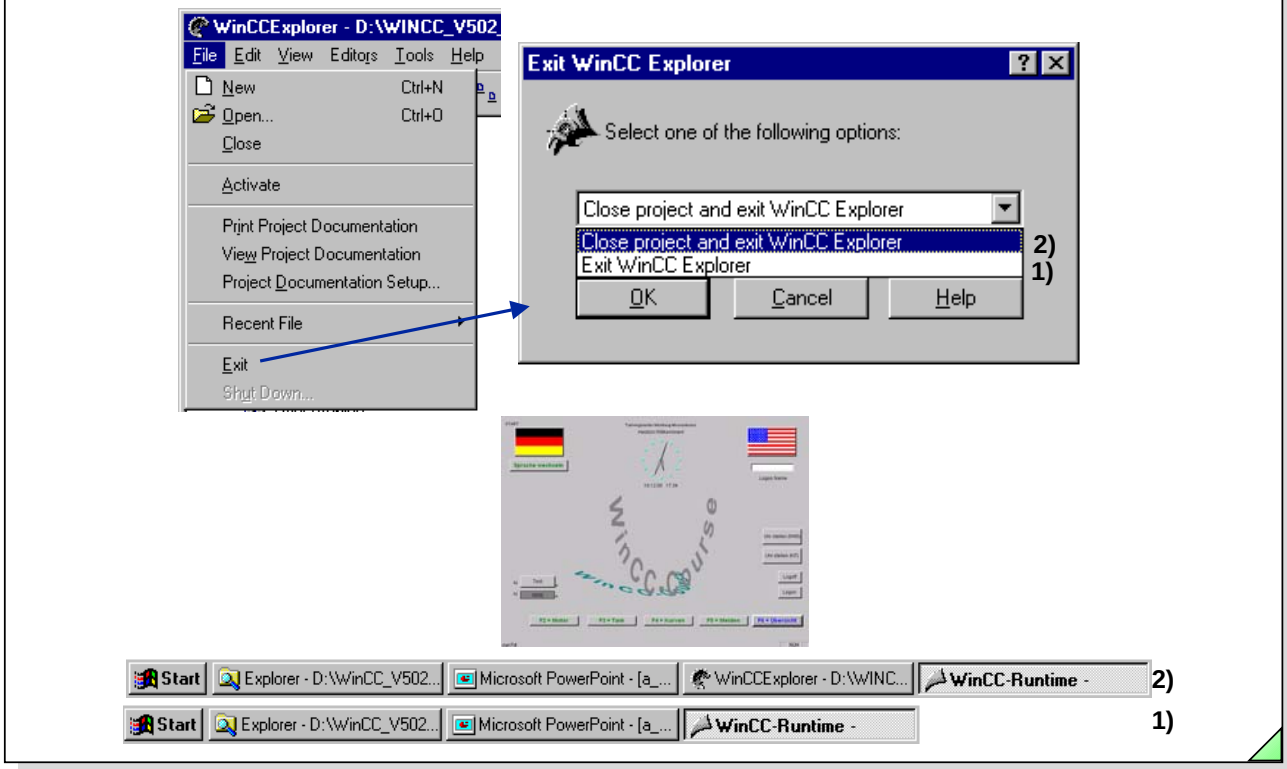
See the notes at the end of the chapter.



Note The WinCC Explorer must be exited with the Runtime System in active mode, so that the Runtime System automatically starts the next time the system boots up.

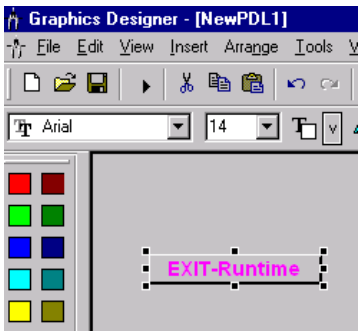
.

Exit WinCC Explorer



Exiting WinCC Explorer

When Runtime is activated, you can exit the WinCC Explorer task.
WinCC Runtime must then be exited using the system function "Exit Runtime" .

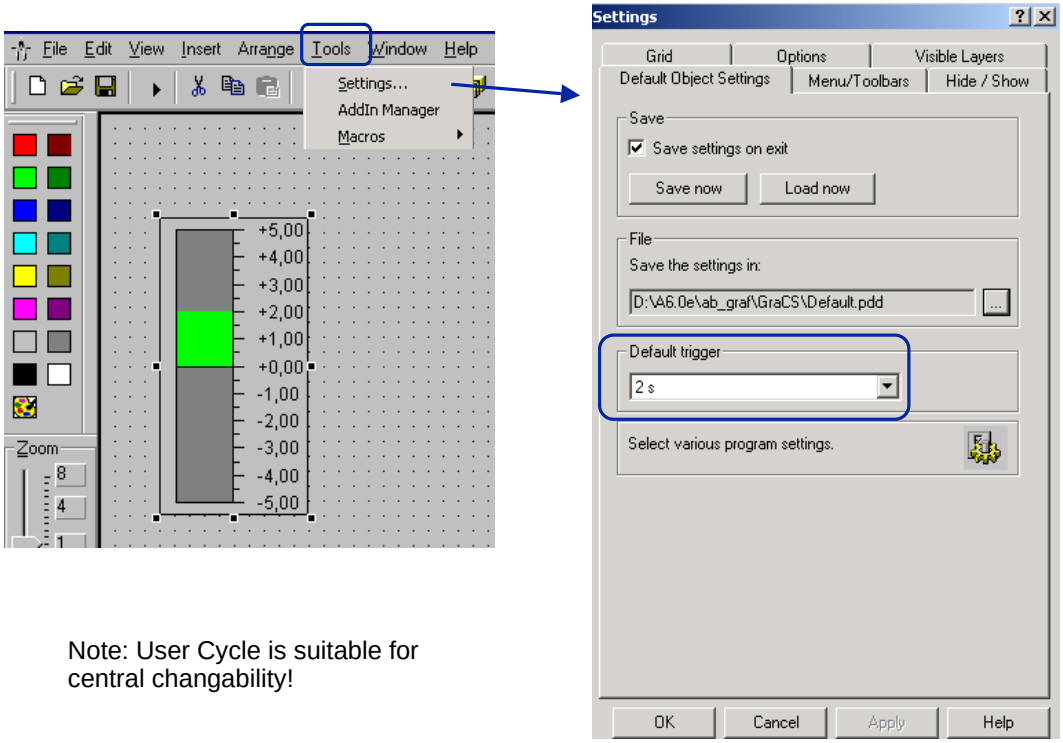


The next time you activate WinCC, the WinCC Explorer is then also activated.

Closing the Project and exiting the WinCC Explorer

WinCC is exited when Runtime is activated and you start the function "Close project and exit WinCC Explorer". The next time you activate the WinCC Explorer, the project is automatically started in Runtime

Default Trigger for Objects



Note: User Cycle is suitable for central changability!

Object Trigger

One of the reasons for achieving high performance picture selection and communication is to use the settting of a default trigger for the objects of a picture. In many cases it then makes sense to set a value of 2 seconds.
The important thing is to achieve a uniformity for the trigger for every picture.

Overview of Possibilities to make Graphic Screens Dynamic with Accessory Examples 1/2

- **Configuration Dialog (fast configuration) Type I/O**
- **Dynamic Wizard (configuration assistant) Type I/O**
- **Tag Connection Type O**
- **Dynamic Dialog Type O**
- Direct Connection Type I
- Property actions on the object corresponding to the ANSI C Syntax Type O
- Event actions on the object corresponding to the ANSI C Syntax Type I
- VBS Visual Basic Script Properties Type O
- VBS Visual Basic Script Event Type I/O

Configuration Dialog Fast method to configure standard applications (objects) e.g. picture change. Not all objects have this kind of dialog. The Configuration Dialog can be activated by clicking on **Tools** in the menu bar and then clicking on **Settings** in the drop-down menu. Then you click the **tab: Options**.

Dynamic Wizard The Dynamic Wizard makes it easier for you to use and configure complex objects, such as I/O fields and bar graphs. The Wizard prompts you to supply all required parameters. These parameters are then entered in the object properties. Later, you can even adapt the parameters individually. The Dynamic Wizard always generates a C-script. The Dynamic Wizard can be activated by clicking on **View** in the menu bar and then clicking on **Toolbars** in the drop-down menu.

Tag Connection If tags are to be displayed 1 to 1, direct tag connection of internal and process tags is available. These process tags can also be addressed indirectly.

Dynamic Dialog A user-friendly way for making objects dynamic is also to use the Dynamic Dialog. For example, it is very easy to configure color changes when various limits are exceeded. By using the Dynamic Dialog, you will achieve a high performance rate in Runtime.

Overview of Possibilities to make Graphic Screens Dynamic with Accessory Examples 2/2

❑ Configuration dialog (fast configuration)	Type I/O
❑ Dynamic Wizard (configuration assistant)	Type I/O
❑ Tag Connection	Type O
❑ Dynamic Dialog	Type O
❑ **Direct Connection**	**Type I/O**
❑ **Property actions on the object corresponding to the ANSI C Syntax**	**Type O**
❑ **Event actions on the object corresponding to the ANSI C Syntax**	**Type I**
❑ **VBS Visual Basic Script Properties**	**Type O**
❑ **VBS Visual Basic Script Event**	**Type I/O**

Direct Connection

The direct connection is a special type of action. Using the Direct Connection, you can very quickly pass fixed process sizes onto a tag, for example. In addition, individual object properties can be linked directly with tags here (for example, a string tag with the text property of a static text). By using the Direct Connection, you will achieve a high performance rate in Runtime.

Property Actions on the Object

There are various possibilities available to make graphic screens dynamic. Actions in C Syntax and VBS can be used on the object. Many properties can be dynamically created, such as color, object size, position, and font size, through internal tags or process tags. Triggering takes place through a timer or by making a tag change.

Event Actions on the Object

You can also use actions in the C syntax and VBS for extensive actions that are to be edited when triggered by an event. Complex calculations can, for example, be carried out in these actions. The results can then be displayed in WinCC pictures and objects. Triggering takes place when object properties are changed, such as color change, or events to an object, such as mouse click.

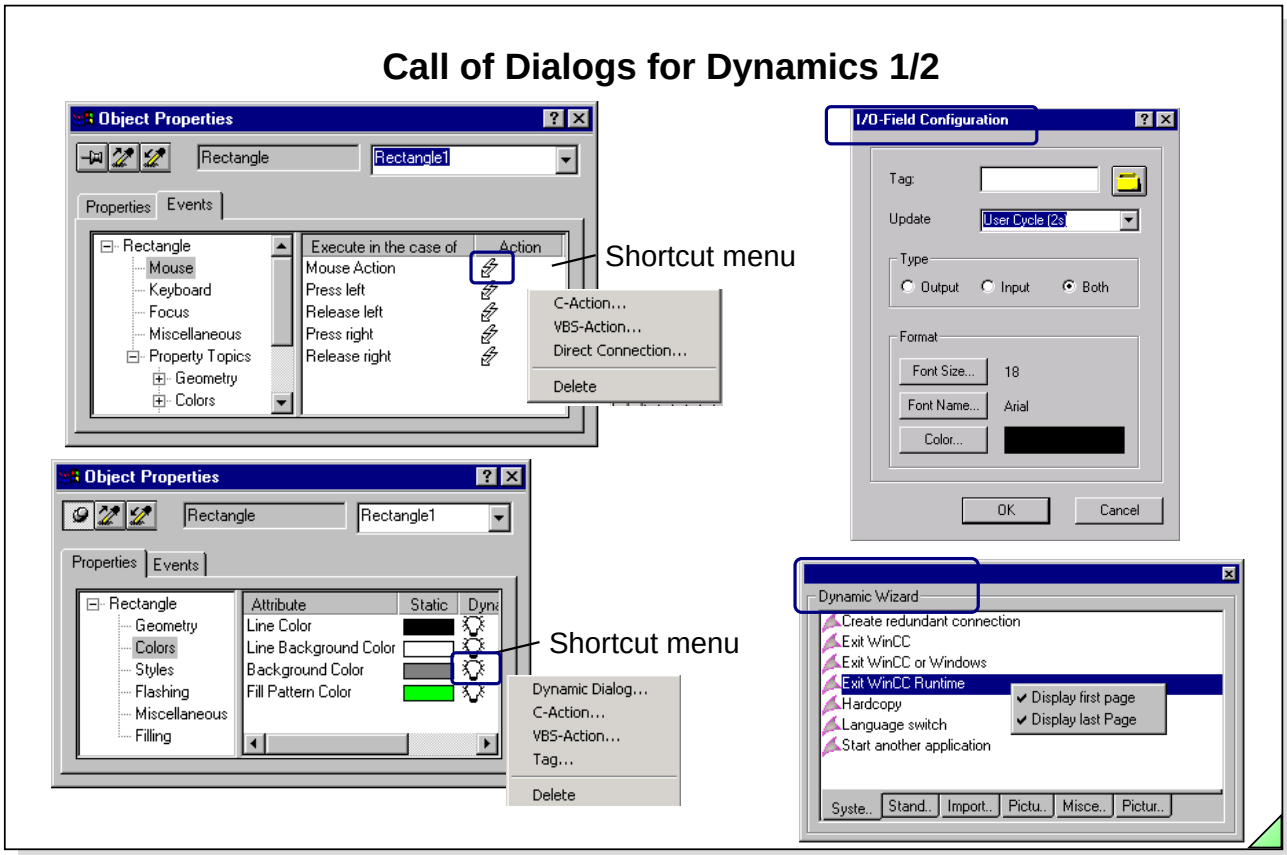
VBS

With VBScript (VBS), you have, in Runtime, access to tags and objects of the graphic Runtime system. Actions can be executed using objects.
This results in many possibilities of making the properties of graphic objects dynamic or of being able to react to events in Runtime. You can create picture-independent actions and procedures using the Global Script Editor. VBS supports the access using the **C**omponent **O**bject **M**odel.

VBS and C

In VBS, internal interfaces to tags and libraries are available, while you can also access other subsystems of WinCC while in the C environment (such as the Report System). Also see WinCC IS, **Performance**, action configuration.

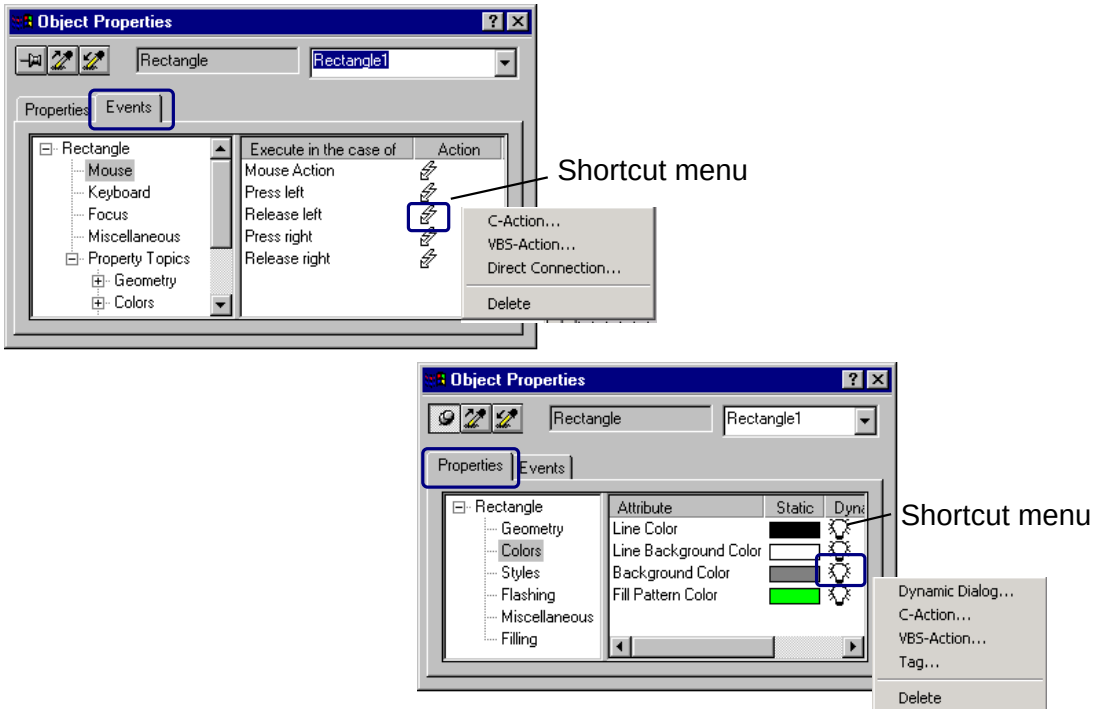
Call of Dialogs for Dynamics 1/2



Date: 27.10.2003
File: SWINCC_04e.28

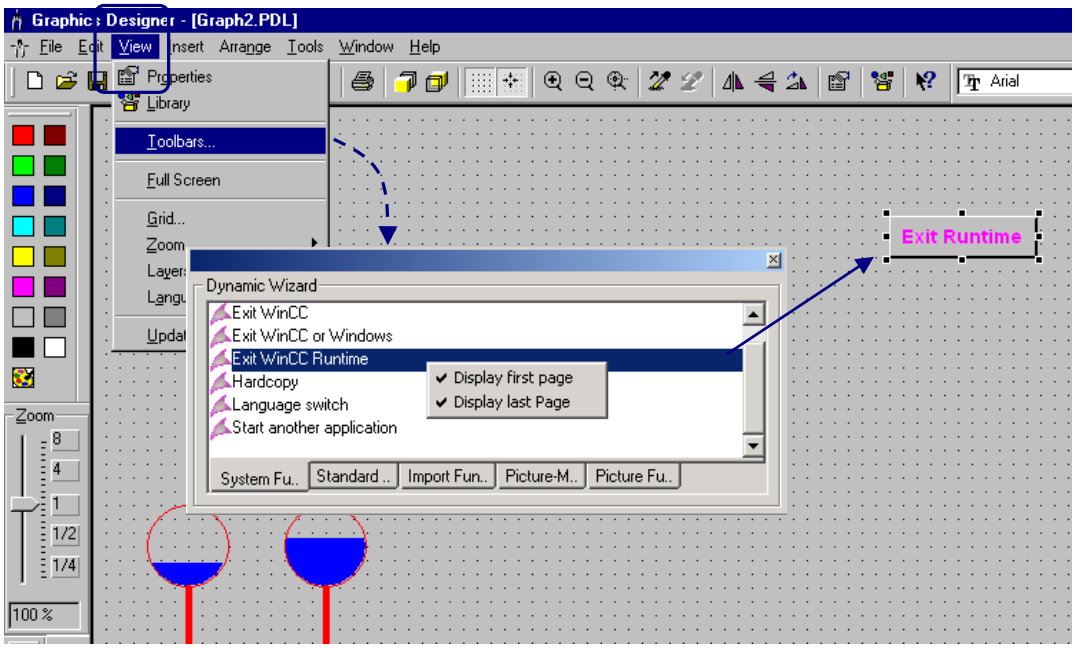
Dialog	Call
Configuration Dialog	Not all objects have such a dialog. It appears automatically when you generate these objects. You can show or hide (activate or deactivate) the Configuration Dialog by clicking **Tools** in the menu bar and then clicking **Settings** in the drop-down menu. Select an object in the Object Palette and position it in the graphic picture. For changes: open the shortcut menu with the right mouse button and start the Configuration Dialog.
Dynamic Wizard	Select an object in the picture --> select Property or Event --> choose the required Wizard and start Wizard with double-click. The Dynamic Wizard must be chosen by clicking **View** in the menu bar and then clicking **Toolbars** in the drop-down menu.
Direct Connection	Select an object in the picture --> display Object Properties --> choose the Events tab --> in the Action column, open the shortcut menu with the right mouse button --> select Direct Connection.
Tag Connection	Select an object in the picture --> display Object Properties --> choose the Properties tab --> in the Dynamic column, open the shortcut menu with the right mouse button --> choose Tag --> in the following dialog, select the required tag and accept it.
Dynamic Dialog	Select an object in the picture --> display Object Properties --> choose the Properties tab --> in the Dynamic column, open the shortcut menu with the right mouse button --> select Dynamic Dialog --> in the following dialog, configure the required dynamics and accept it.
C-action	Select an object in the picture --> display Object Properties --> choose the Properties or Events tab --> in the Dynamic or Action column, open the shortcut menu with the right mouse button --> select C-action --> configure the required C-action and push the "create action" button.

Call of Dialogs for Dynamics 2/2



Dialog	Call
VBS	Select object in picture --> Display Object Properties --> choose the Properties or Event tab -->in the Dynamic or Action column, open the shortcut menu with the right mouse button --> select VBS Action... --> configure the required script and push the "create action" button. Through the project window of the WinCC Explorer, select Global Script and Open the VBS Dialog with the shortcut menu command.

Dynamic Wizard Overview 1/2



The Wizard generates a C-script in the background that can be used for further applications.

Date: 27.10.2003
File: SWINCC_04e.30

Dynamic Wizard

The Dynamic Wizard enables you to create certain frequently required dynamic actions or even operations on various objects and object properties by means of program prompting. The Dynamic Wizard is called several times in succession to create multiple dynamic actions on the same object. This makes it possible to activate certain properties in a very targeted way without, for example, having to formulate the actions.

You can also make additions at a later time when you use the property pages for the respective object.

Note

The Dynamic Wizard can be activated and de-activated by clicking **View** in the menu bar and then clicking **Toolbars** in the drop-down menu.

Selection

This wizard supports the frequently used dynamics for the selected object. These dynamics take place in three steps:

Step 1 **Selecting the desired dynamic**

Step 2 **Selecting a trigger for this dynamic and**

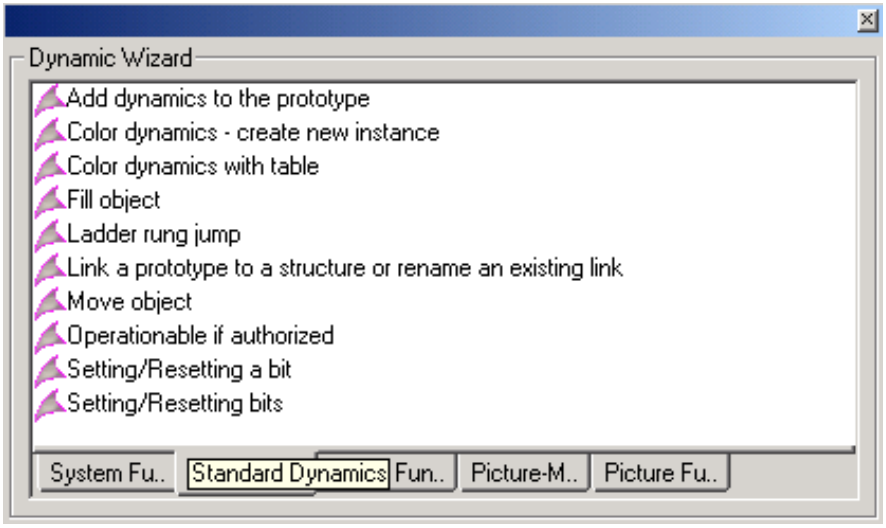
Step 3 **Settings for options**

Then a C script is automatically generated, and the required dynamics are assigned to the object.

Picture Functions

- *Screen Navigtion*
- *Picture Change in the Window* (changes the picture in a certain picture-window);
- *Single Picture change* (changes the current picture);
- *Display Error Box*
- *Display WinCCDialogbox* (inserts a PDL that can be used as a dialogbox)

Dynamic Wizard Overview 2/2



The Wizard generates a C-script in the background that can be used for further applications.

Import Functions
- Import Messages (file exported from COROS LS-B in TXT format).
- Import S5 Only Tag (file exported from COROS LS-B in TXT format).
- Import S7 S5 - Assignment list (variable import from S5 or S7 assignment list).

Standard Dynamics
- See slide above.

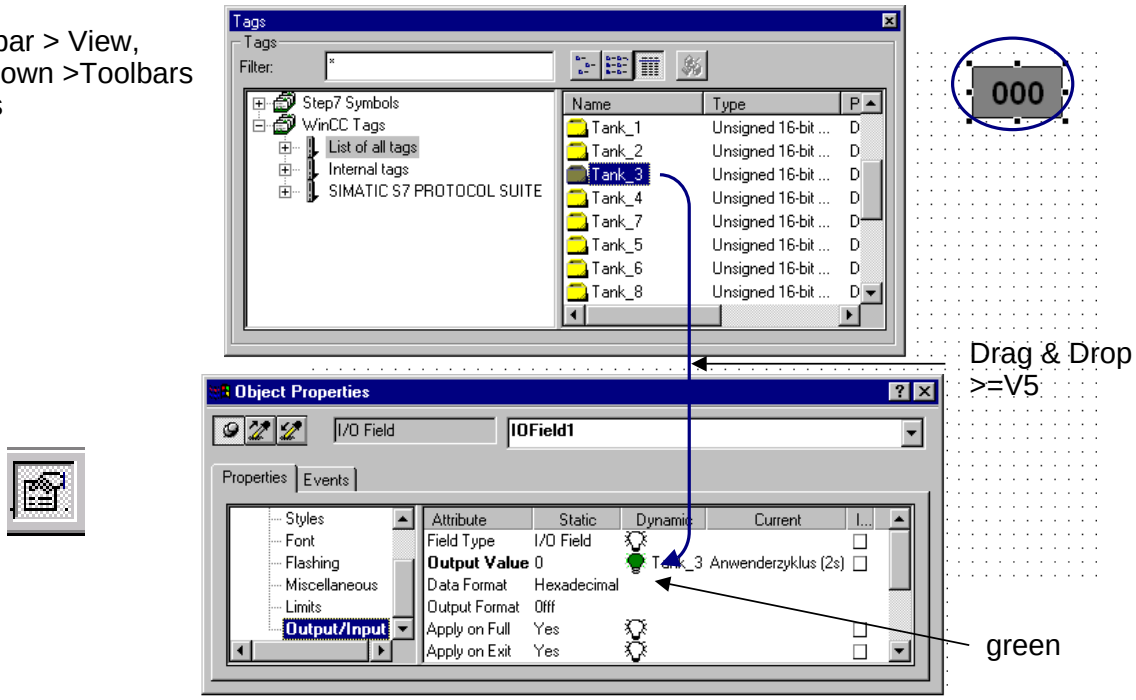
System Functions
- Starting another application (starts any other Windows application)
- Hardcopy (makes a hardcopy of the current screen on the standard printer)
- Setup redundant connection
- Language switch-over
- Exit WinCC (exits WinCC and shuts down all WinCC applications)
- Exit WinCC or Windows (shows several possibilities to exit, reboot and logon)
- Exit WinCC Runtime (exits WinCC Runtime and switches to the Design Mode)

Picture Module
- *Generate Picture module - Template .*
- *Generate Picture module - Instance.*

.

Tag Connecting Overview
Decimal Display

Menu bar > View,
Drop-down >Toolbars
->Tags



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.32

SITRAIN Training for Automation and Drives

Connecting Tags

Process tags or even internal tags can make various properties dynamic, either directly or indirectly. For the output of a process value, the "output value" property of an I/O field is connected with the desired tag, for example. This connection takes place through the tag browser or with the toolbar's "List of all tags". With it you have access to all the tags present or configured in the system.

Direct/Indirect

It is also possible to access indirect tag addressing to supply an I/O field. A tag (address tag) is defined that contains the tag name of the target tag. By changing the contents of the "address tag", it is possible to access various tags.
This also changes the linking of the tag to the I/O field.
Indirect addressing is distinguished by the following characteristic: In the case of "indirect", a check mark is entered on the properties side in addition to the tag by means of a double-click.
For indirect addressing, the tag entered there must then be a text tag.

Updating

The default value entered for the update time is 2 s. This value can be changed within broad limits. In the menu bar click on **Tools, in the pop-down menu select Settings, then the Default Objects Configuration tab**. In addition, there are other update possibilities available:
picture cycle, window cycle, or even a tag change.

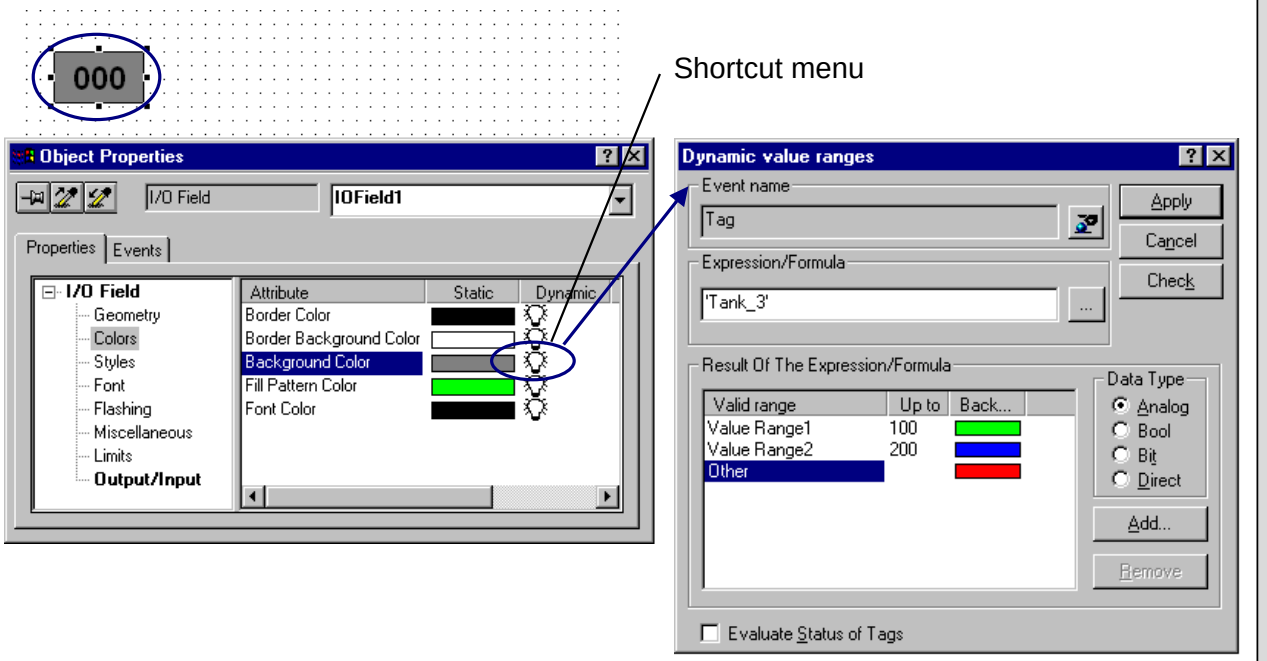
Note

A green light bulb indicates an object that has been made dynamic and is connected to a tag.
You can increase the performance by using the tag connection. It should always be used for an optimized configuration.

Exercise 2

See the notes at the end of the chapter.

Dynamic Dialog Overview
Color Attribute Change



Date: 27.10.2003
File: SWINCC_04e.33

Dynamic Dialog

The dynamic dialog can also be used in place of actions on the object. Using the dynamic dialog significantly simplifies configuring limit value checks, for example. It is not necessary to create an action with the respective scans. For the configuration, it is sufficient to simply enter the limits and the parameters to be set corresponding to the value.

Configuring

Use the right mouse button to click on the light bulb symbol to trigger the dynamic dialog configuration (pop-up menu for the object property). The scan results are immediately linked with the connected property.

Changes

If changes are necessary, you can integrate them very easily in the existing scan (edit dynamic dialog).

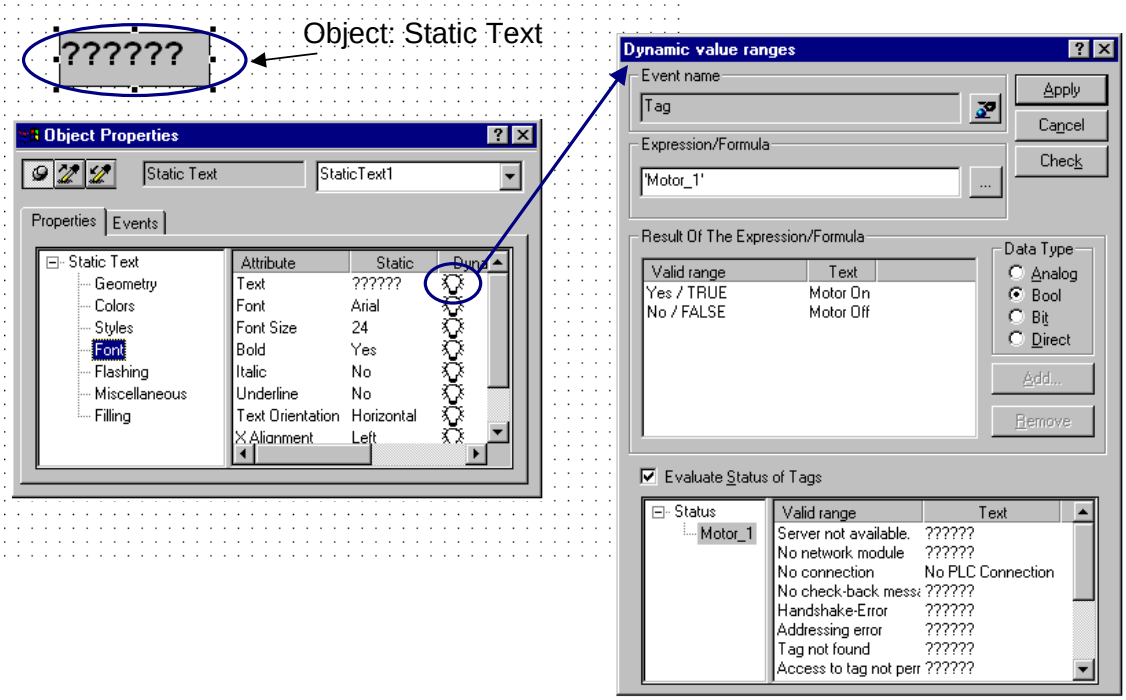
Note

If the attribute was made dynamic by means of a dynamic dialog, a red lightning bolt symbol replaces the light bulb symbol.

You can increase the performance by using the Dynamic Dialog. This is only possible, however, if the tag trigger is used. The tags are all called for once during picture selection. After that, the system optimizes itself by only processing the function when the value of a tag changes.

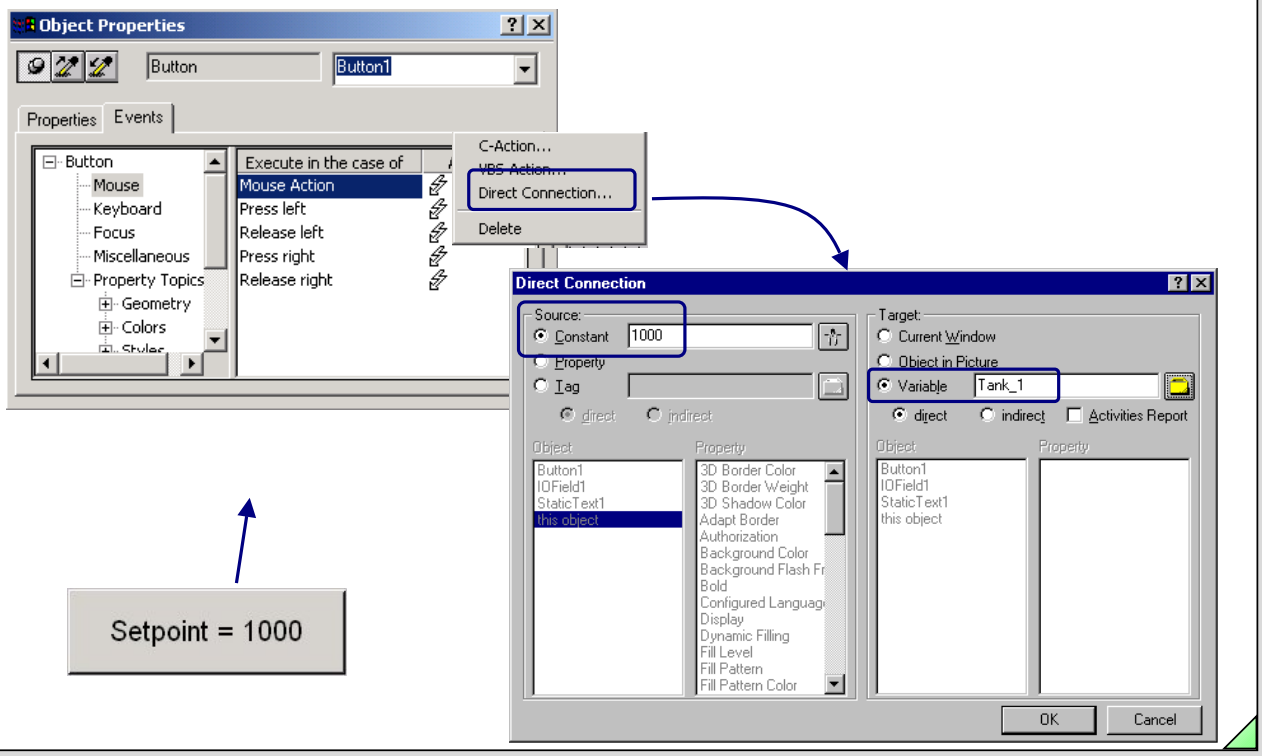
The status of a process tag can be evaluated with the help of the Dynamic Dialog, e.g. no connection to the PLC.

Dynamic Dialog Overview
Status Evaluation for Process Tags



Date: 27.10.2003
File: SWINCC_04e.34

Direct Connection Overview



Date: 27.10.2003
File: SWINCC_04e.35

Direct Connection

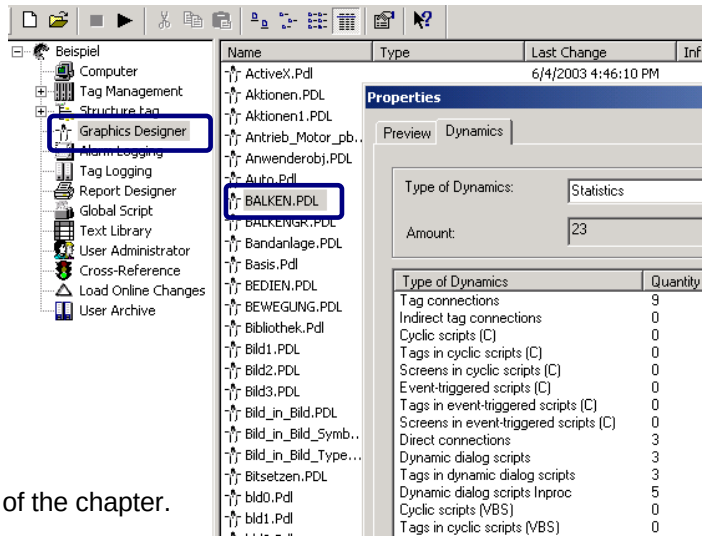
The direct connection makes it possible for you to connect certain properties and actions with each other, without formulating the action as C script. This is how a very fast connection between individual properties and the respective supply can be created.

The screen in the slide above displays the direct connection between the value of a mouse operation and the description of a tag.
It is just as easy to configure a direct picture change, for example.
As of WinCC V5, the direct connection at the target object will be shown through the object properties and will be recognizable by the Italic type.

Note

If a direct connection was configured, the lightning bolt symbol is displayed in blue.

The references of the direct connection are displayed through the picture properties.



Exercise 3

See the notes at the end of the chapter.

The Programming Language C in General

From the Program Text to the Executable Program

Source Text

↓

- ❑ **Editing** (With a compiler that contains a text editor)
- ❑ **Compiling** (The C compiler creates an object file with a machine code.)
- ❑ **Linking** (The Linker connects several objects into an executable program.)
- ❑ **Executing** (The compiled and fully connected program can then be executed.)

Historical

Dennis M. Ritchie at Bell Laboratories developed C at the beginning of the 70's. This development is closely connected with the development of the UNIX operating system, of which over 95% is written in C. Over time, C has developed into an independent programming language, independent of the operating system.
The ANSI Committee X3J11 published a language description for C in 1988. ANSI published the standardization for C in December of 1989.

Advantages

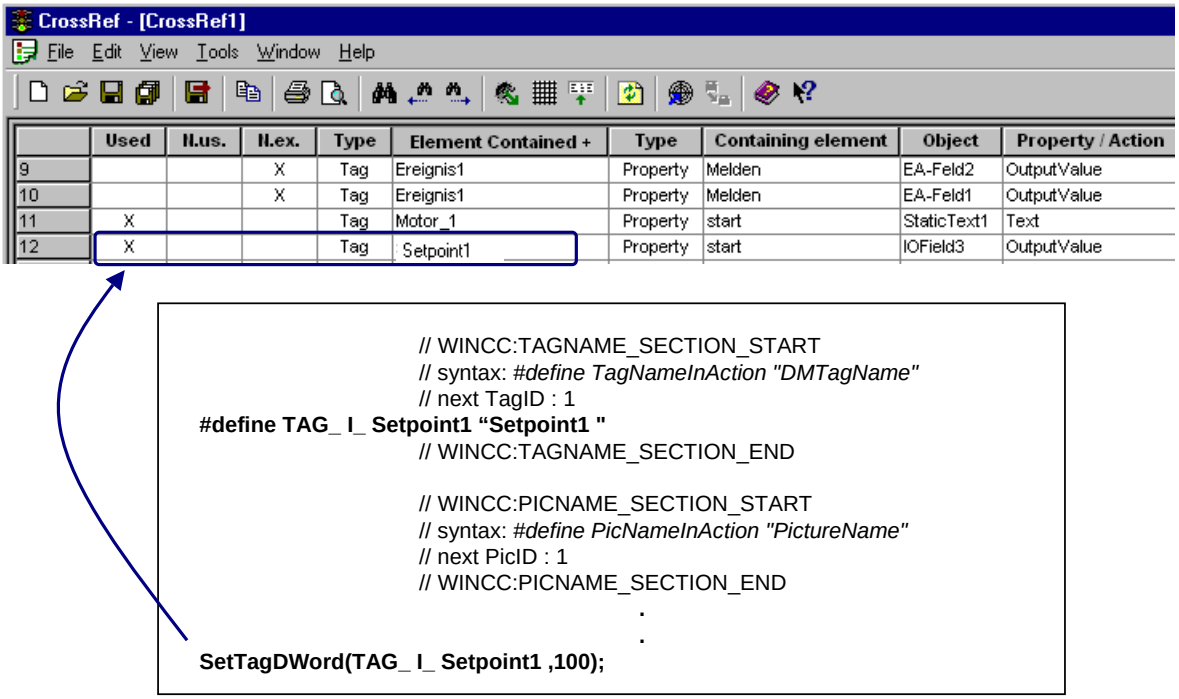
C combines the performance capability of an assembler language with the convenience of a higher programming language.

- Contains the memory properties and constructs of other programming languages, such as block structures, and pointers, etc.
- Related hardware programming is possible.
- Appropriate for time critical applications
- Appropriate for system programming
- Universally applicable language
- Small language
- Functionally simple to expand

Variable Types in C

int	-2 147 483 648 to 2 147 483 647
unsigned int in WinCC also **DWORD**	0 to 4 294 967 295
short int in WinCC also **SHORT**	-32 768 to 32 767
long int	-2 147 483 648 to 2 147 483 647
unsigned short int	0 to 65 535
unsigned long int	0 to 4 294 967 295
char	all ASCII characters
unsigned char	all ASCII characters
float	-10"38 to 10"38
double	-10"308 to 10"308

Configuration Rules in C Actions



Tags in Actions

Tags in actions can only then be found and replaced by the Cross Reference if the **configuration rules for tags and screen names** predefined in the C-Scripts are adhered to.

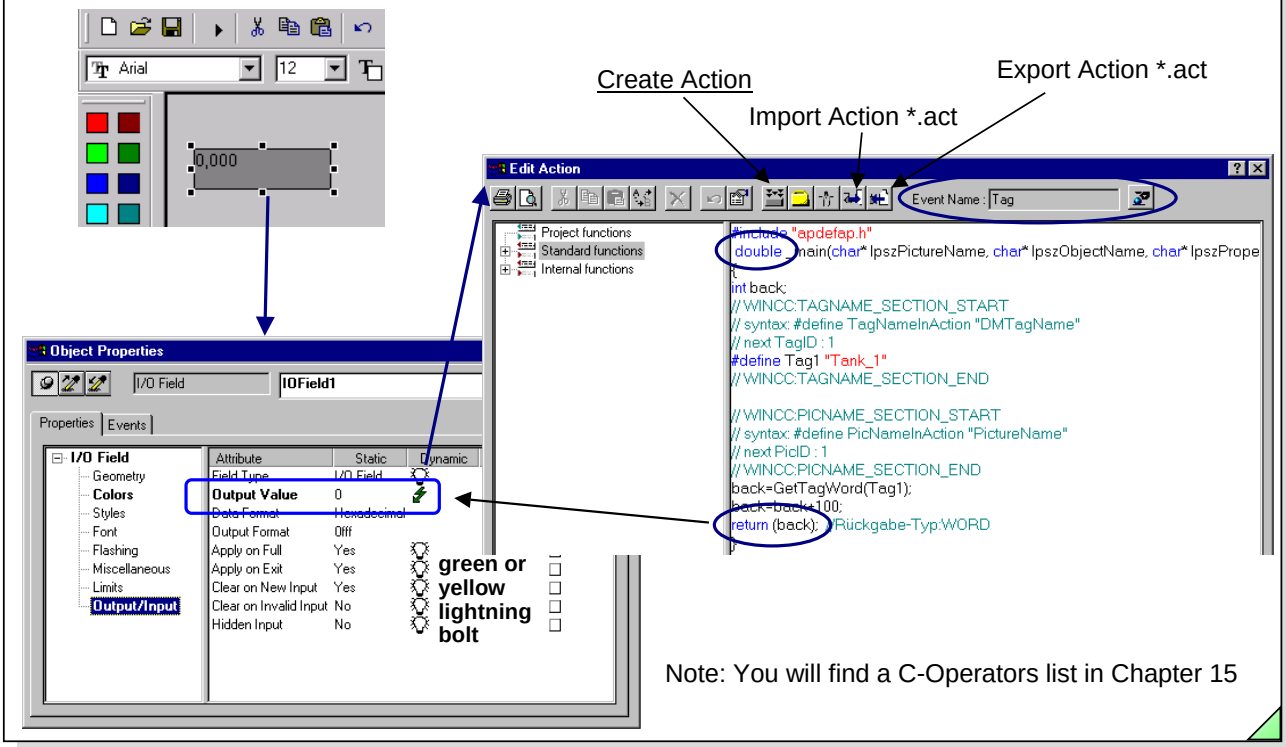
Note

The configuration rules are already taken into consideration by the Dynamic Wizards during generation of the scripts.
When you set up C-Scripts, a comment recognizes the configuration rules.

You can find further information in the **Online Help** from WinCC Information System, Index tab, Keyword: CrossReference.

As of the WinCC V5.0+SP1 version, you can use the CrossReferenceAssistant to convert the V4.02 projects to V5.0 projects. You will find this tool on the WinCC CD.

Overview of Property Actions with C Syntax
Calculation of an Output Value



Date: 27.10.2003
File: SWINCC_04e.38

Actions on the Object

Actions on the object make it possible to manipulate individual picture object properties. These actions can be used to make even static objects become dynamic objects. Internal tags or even external tags (process tags) can trigger the changes.

Properties

The type and number of properties an object has is dependent on the object. The "static text" object contains properties such as the following:

- Geometry (position X, position Y, width, and height)
- Colors (border color, border background, background color, fill color, and font color)
- Styles (border weight, border type, and fill pattern)
- Font (text, font type, font size, bold, italics, underline, text orientation, X alignment, and Y alignment)
- Flashing (flashing border active, border flashing color off, border flashing color on, border flashing frequency, flashing active, etc.)
- Miscellaneous (operate enable, password, and display)
- Filling (dynamic filling and fill level)

Example

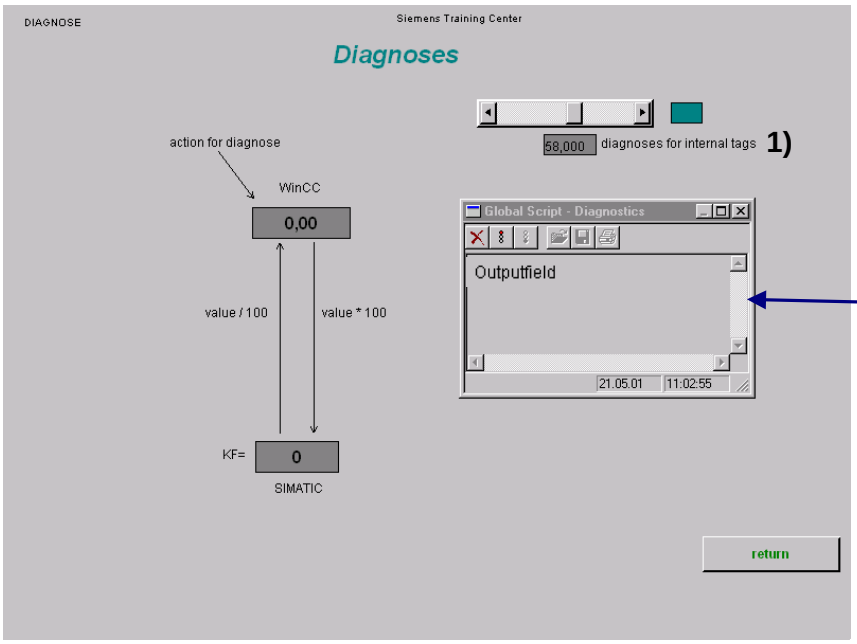
Output/Input: When you evaluate the "**Tank_1**" tag, the **output value** is determined through a scaling (calculation).

Note

A **green lightning bolt** indicates that an object that is linked to an action has been made dynamic. A **yellow symbol** indicates a C script has not been compiled. The frequency to run the action is determined by the Event Name (e.g. 500 ms), see picture above.

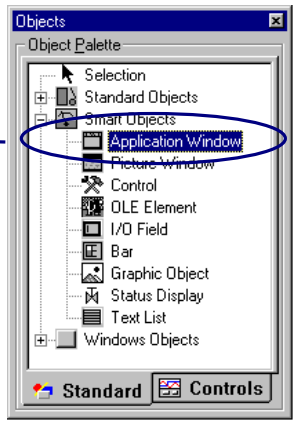
The tag trigger setting makes sure that the tags are all called for once during picture selection. After that, the system optimizes itself by only processing the function when the value of a tag changes.

Diagnostic Means for C-Scripts



1)

```
{
WORD back;
back=GetTagWord("wert");
printf("Outputfield \r\n");
return back;
}
```



Troubleshooting in C-Scripts

WinCC provides an **Application Window-> Global Script->GSC Diagnosis** for testing C-Scripts . It is inserted in a picture. From this point on, all printf functions from all C-Scripts are displayed in the application window.
The printf instruction is included in the ANSI-C library and has the following syntax, for example printf("\r\n Wert1: %d", sum);

Troubleshooting can be carried out simply and easily with this method.

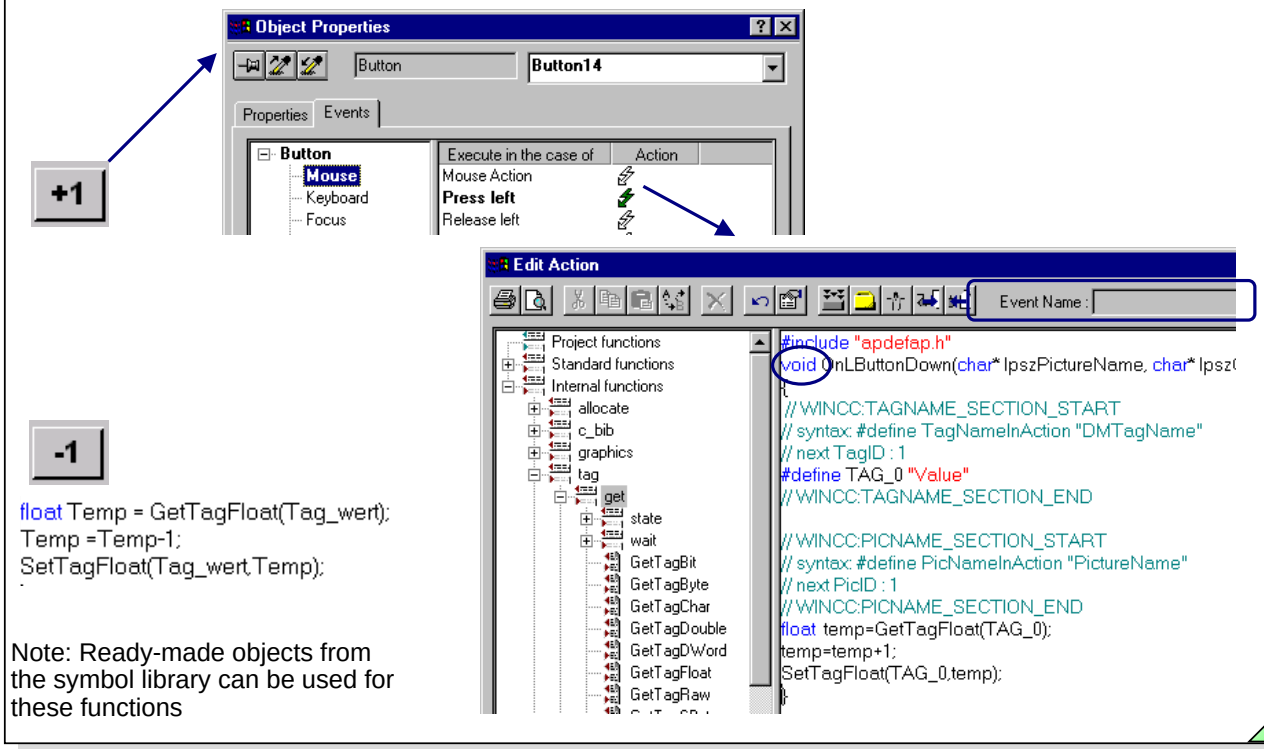
Note

In order to give emphasis to the new printf function, we dispensed with the comment and define lines.

Exercise 4

See the notes at the end of the chapter.

Overview of Event Actions with C Syntax Increment/ Decrement



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.40

SITRAIN Training for Automation and Drives

C Functions

Very extensive actions, checks, and scans can also be processed with C functions. In addition to the standard C functions (ANSI C), there are, of course, still the WinCC-specific functions available for reading and writing tags and for processing all possible objects.

Configuring

C functions can be used where the direct connection to individual tags is not sufficient, or even where several properties are to be changed simultaneously. With the help of the C functions, you obtain an extensive comprehension of all the object properties available, and of all the WinCC tag contents.

Event

If a C action is configured for an event, a **green lightning bolt** symbol represents the action that is loaded.
If the C action has not yet been compiled, the **lightning bolt** symbol is displayed in **yellow**.

If the condition of a bit is used in an IF statement, the condition is either TRUE or FALSE (FALSE is equal to zero, TRUE is not equal to zero).

The type and number of properties of the object depends on the selected object.

The action is executed when the selected property changes, or by mouse click/keyboard.

Example

A setpoint (value) is to be changed in fixed steps using two buttons. A range limit for upper and lower value can also be set with the library objects.

Visual Basic

Microsoft Visual Basic for Applications ← V6.3

Note: From the WinCC V6 CD “Additional Software“, install the Microsoft Script Debugger

Introduction

What is Visual Basic? The "Visual" in the name refers to the procedure for creating graphic user interfaces. Instead of writing innumerable program lines that specify what the interface elements are to look like and how they are positioned, you simply move prefabricated objects to the desired position using drag & drop. The "Basic" in the name refers to the BASIC language (Beginners All-Purpose Symbolic Instruction Code), a language that is used worldwide by more programmers than any other language. Visual Basic developed from the original BASIC and today contains several hundred instructions, functions and keywords, many of which refer directly to the Windows interface. Those just starting out can create practical applications just by learning a few basic keywords. At the same time, the language, through its performance, allows the pros to do everything that can be developed with all the other Windows programming languages.

Area of Use

The Visual Basic programming language is not just used for Visual Basic. The Visual Basic for Applications in Microsoft Excel, Microsoft Access and in many other Windows applications uses this same language. VBA was developed for the automation of software applications. The Visual Basic Programming System Scripting Edition (VBScript) for Internet programming.
VBA and VBS is a subset of the Visual Basic language.

VB in WinCC

The VBS Scripts are based on the Microsoft Scripting host (run-time environment), that is installed in every Windows 2000, Windows XP and Windows ME System. WinCC has its own Editor.

The Microsoft Visual Basic Editor is used for VBA scripts. .

Note

http://www.microsoft.com/germany/ms/windows2000/magazin/artikel135.htm

VBS, VBA Data Types

VBS Data type: VBS uses the Variant data type. The Variant data type automatically adapts itself to the assigned value.

VBA Data types

Name	Type	Value Range	Memory
Byte	**Integer**	**0-255**	**1 Byte**
Boolean	**Integer**	**0,1**	**2 Byte**
Integer	**Integer**	**-32,768 to 32767**	**2 Byte**
Long	**Integer**	**-2,147,483,648 to 2,147,483,647**	**2 Byte**
Single	**Floating point**	**-3,37E38, to 3,37E38 (7 digits)**	**4 Byte**
Double	**Floating point**	**-1.7...E308, 1.7...E308 (15 digits)**	**8 Byte**
Date	**Date**	**1.1.1000 (-657434) to 31.12.9999 (+2958465)**	**8 Byte**
String	**String**	**0 to 65535 characters**	**Number of characters**
Variant	**any**	**Depends on the assigned data type**	**at least 16 Byte**

Example: Dim Setpoint As Integer

You will find a VB Operator list in Chapter 15.

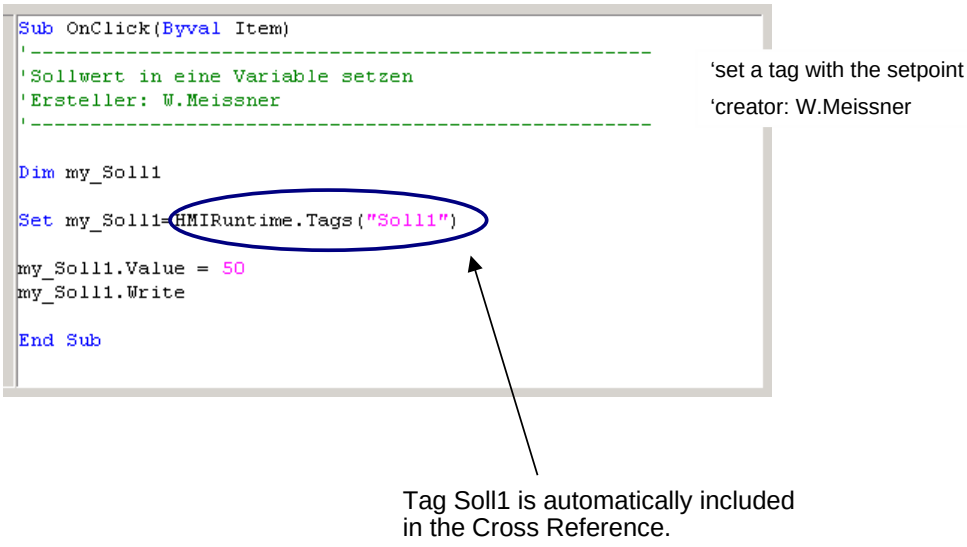
VBS(RT) and VBA(CS) You use VBA in WinCC Configuration during configuration to adapt the Graphics Designer to your individual requirements and to simplify the configuration and to automate it. VBA programs only operate in the WinCC configuration environment.
Unlike VBA, VB Scripts only run in WinCC Runtime and enable you to access graphic objects and tags. With VBS, you can neither create objects and pictures nor can you permanently change them, unlike VBA.

Essential differences between VBA and VBS are, for example:

+VBS was developed for use on the Internet, VBA for the automation of software applications.

+ The data type of VBS tags is always VARIANT. VBA, on the other hand, makes a distinction between the individual data types such as INT, DOUBLE, STRING etc.
+ Certain language constructs of VBA were removed or supplemented in VBS.
+ Troubleshooting in VBS is handled differently than in VBA.

Configuration Rules in VBS Actions



Tags in VBS Actions

With the WinCC CrossReferences, you can also quickly find all Places of Use of tags and screen names in VBS Actions. You can "rewire" tags from actions in the Graphics Designer with CrossReference. In other words, you can replace them with other tags in all or in selected positions.

Configuration Rules for VBS Scripts

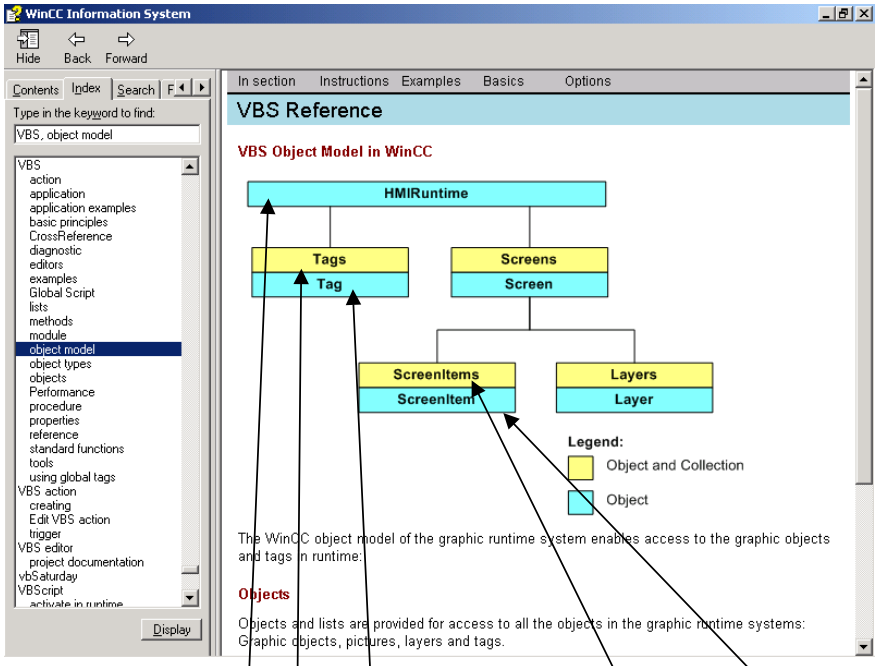
All tags that you address with the standard format
HMIRuntime.Tags("Tagname")
are automatically recorded by the WinCC CrossReference and are listed in the screen properties.
Should you address tags with other formats in your code, you can access them through the following section of the CrossReference:
'WINCC:TAGNAME_SECTION_START
'Const TagNameInAction = "TagName"
'WINCC:TAGNAME_SECTION_END

All screens that you address with the standard format
HMIRuntime.BaseScreenName("Screenname")
are automatically recorded by the WinCC CrossReference and are listed in the screen properties.
Should you address screens with other formats in your code, you can access them through the following section of the CrossReference:
'WINCC:SCREENNAME_SECTION_START
'Const ScreenNameInAction = "ScreenName"
'WINCC:SCREENNAME_SECTION_END

Note

Screen names are to be written without the file extension ".PDL".

VBScript Object Model in WinCC



Example: Set my_Soll1=HMIRuntime.Tags("Soll1")

Example: Set Rect = ScreenItems("RoundedRectangle1")

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.44

SITRAIN Training for Automation and Drives

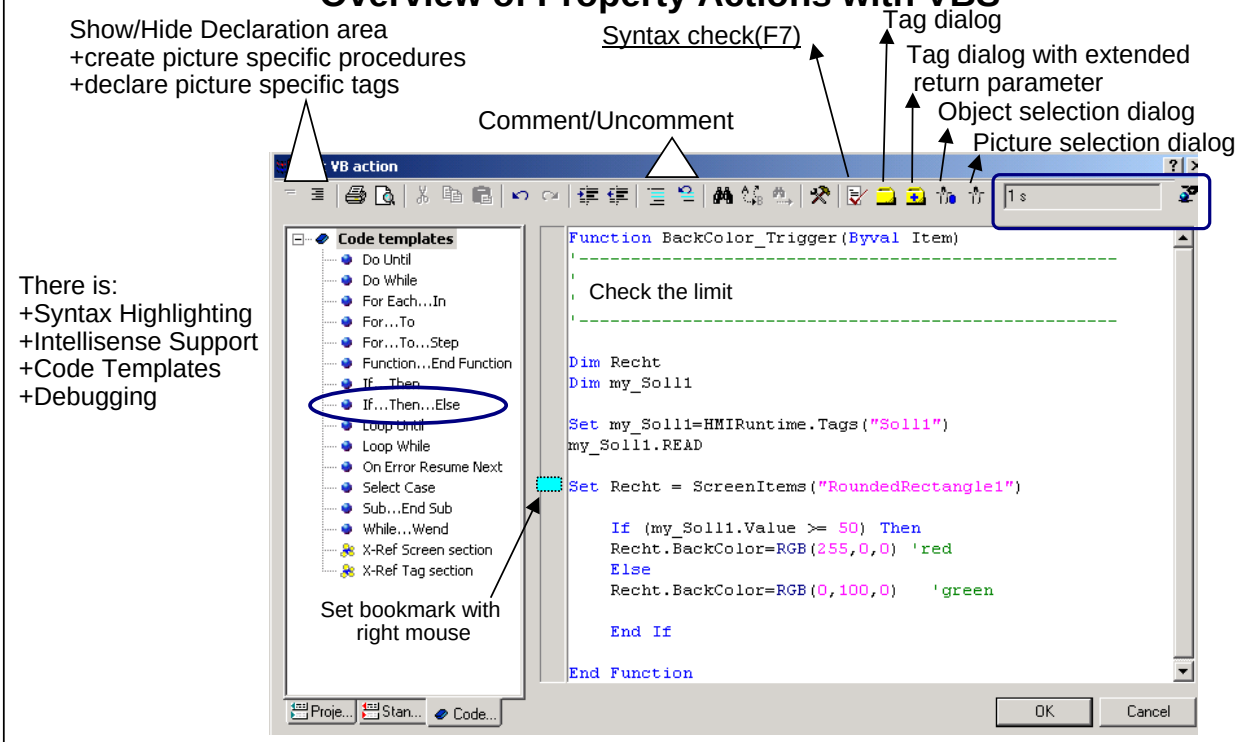
Objects Through the objects and the listings, you can access all objects of the graphic Runtime system: graphic objects, screens, layers and tags.

Properties Through the properties of the individual objects, you can targetedly change the graphic objects and tags in Runtime. For example, enable a control element by mouse click, or by changing a tag value, trigger a color change.

Methods With the methods that you use on the individual objects you can, for example, read out tag values for further processing, or display diagnostic messages in Runtime.
For example:
+ HMIRuntime.Stop
+ HMIRuntime.Trace "Customized error message"
+ Term.Write [Value],[Writemode]
  Writemode=0 or empty->asynchronous, 1=synchronous
+ Term.Read([Readmode]) Mode like Writemode
+ etc.

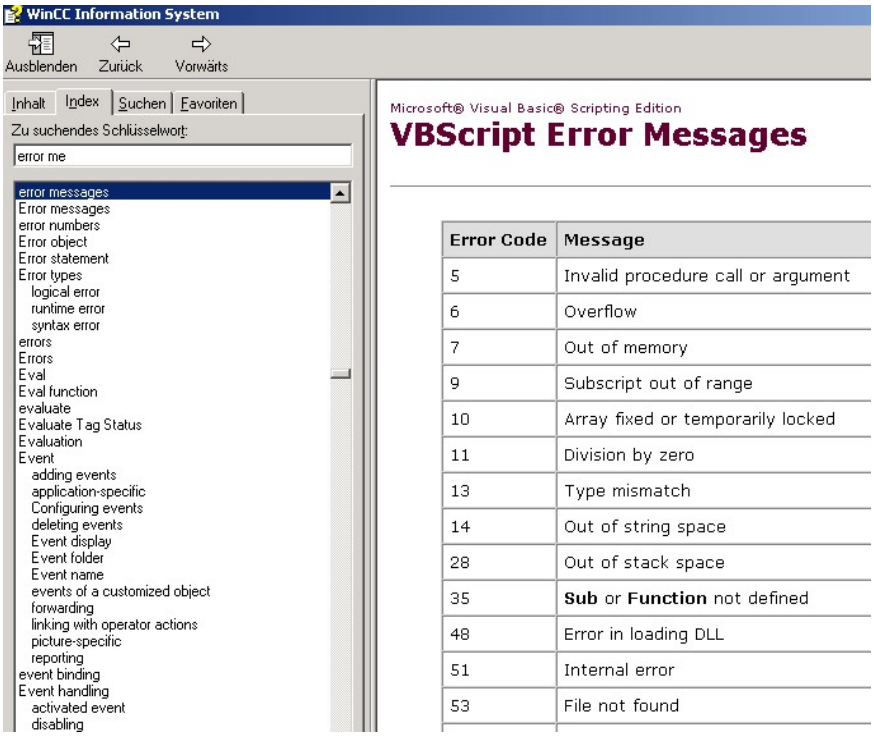
Examples - VBS
+ Access to MS Office applications (Excel, Word, Access)
+ Access to any SQL database
+ Send an e-mail using Outlook
+ Access to methods of ActiveX-Controls
+ Read and Write values in the controller (RT)
+ etc.

Overview of Property Actions with VBS



Note: Only syntax error-free Scripts can be adopted-> take out faulty lines

VBS Error Messages You can display a list of error messages from the Online help, under the Index tab, keyword “Error Messages“



Error Code	Message
5	Invalid procedure call or argument
6	Overflow
7	Out of memory
9	Subscript out of range
10	Array fixed or temporarily locked
11	Division by zero
13	Type mismatch
14	Out of string space
28	Out of stack space
35	**Sub** or **Function** not defined
48	Error in loading DLL
51	Internal error
53	File not found

.

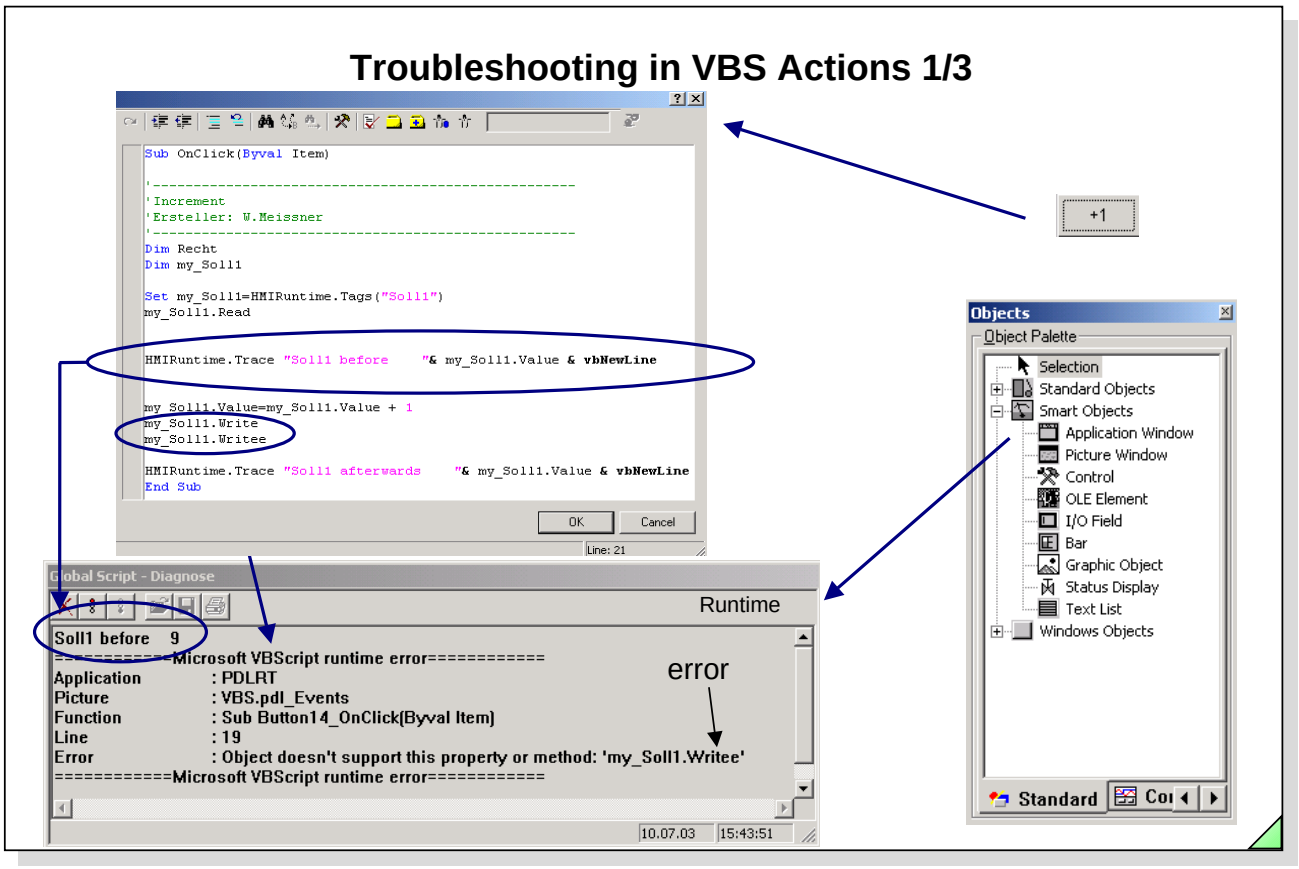
Syntax Highlighting Highlights code sections in color.

~~Intellisense Support~~ ~~Shows a list field for selecting properties, methods and objects~~

~~Code Templates~~ Can be pulled directly into the action window with, for example, drag & drop.

~~Debugging~~ ~~With MS Script Debugger V1.0~~

Troubleshooting in VBS Actions 1/3



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.46

SITRAIN Training for Automation and Drives

General

GSC Diagnostics outputs the trace methods contained in the VBS Actions in the sequence that they are called. This is also true for trace instructions in procedures that are called in actions. Through targeted use of trace instructions, for example to output tag values and customized texts, you can follow the sequence of actions and the procedures called in them. You specify the trace instruction in the format "HMIRuntime.Trace <Variable>.Value".

Note

For performance reasons, you should delete the trace instructions after you have tested the scripts.

Troubleshooting in VBS Actions

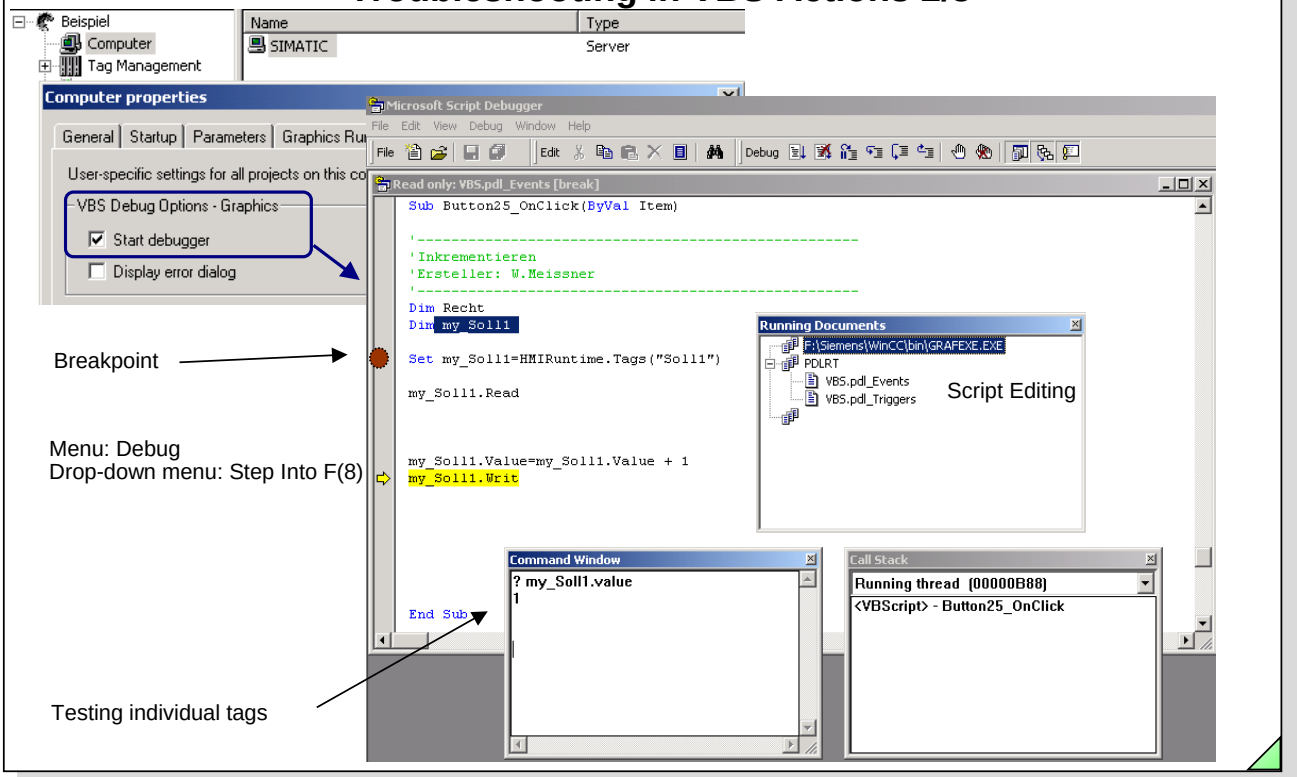
To test VBS actions, WinCC offers an Application Window-> Global Script->GSC Diagnostics. This is inserted in a picture. From this point on, all trace instructions from all scripts and actions are displayed in the Application Window.

Debugger

In addition to the "Microsoft Script Debugger" delivered with WinCC (included in Windows 2000 or Internet Explorer as of version 6) you can also use other Microsoft Debuggers to test, such as:

+ Debugger "InterDev." (included in the installation environment of Developer Studio)
+ Script Debugger (included with Windows XP)

Troubleshooting in VBS Actions 2/3



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.47

SITRAIN Training for Automation and Drives

General

The procedure for finding and eliminating errors in an application is called troubleshooting or debugging. The debug functions support breakpoints, breakpoint comments, monitoring comments and the possibility of moving instruction by instruction or procedure by procedure through the code and displaying the values of tags and properties.

Debugger

You can enable the Standard Debugger in the computer properties.

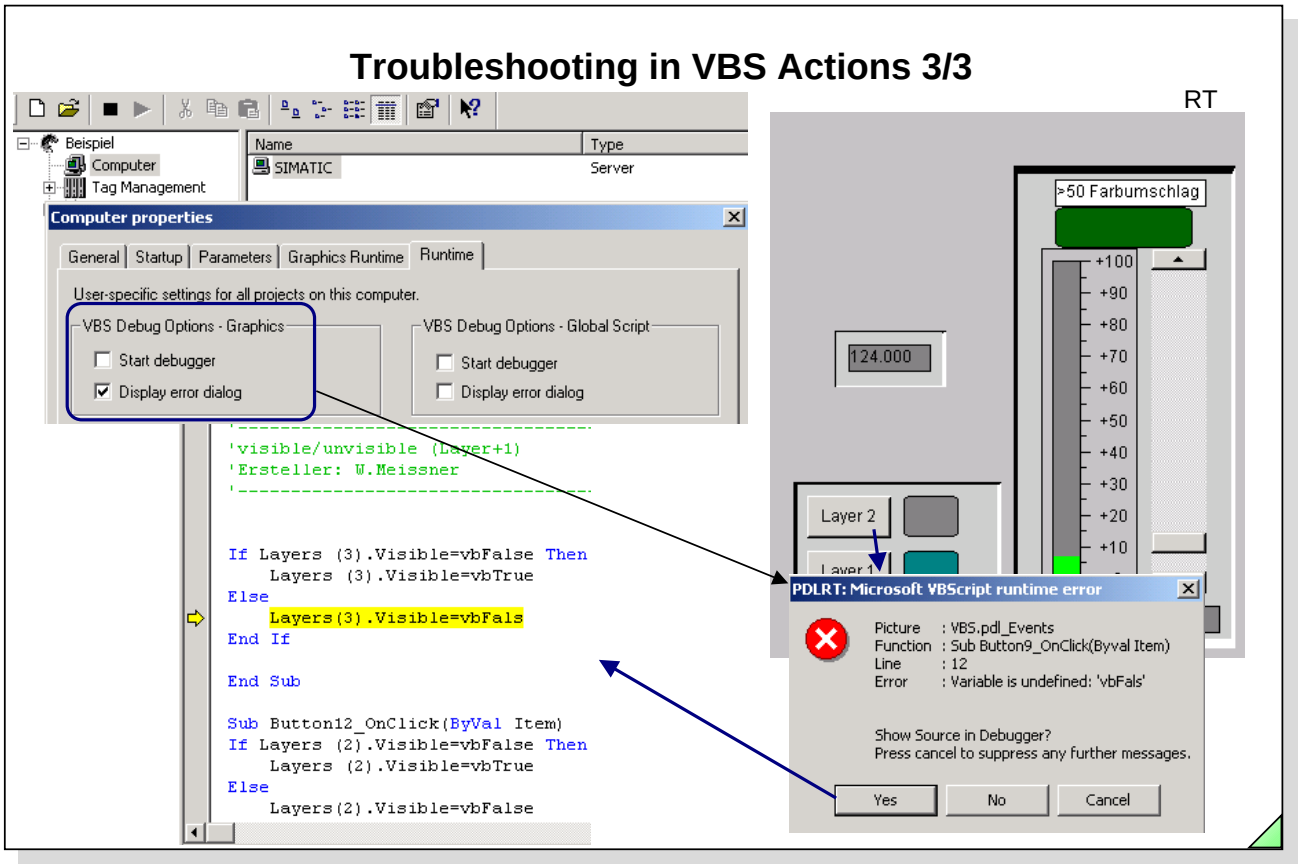
Help

In the Help menu, you will find an extensive description of the MS Script Debugger.

Note

For performance reasons, the MS Script Debugger should be exited after troubleshooting.

The Debugger mode stops all scripts in a Tread.



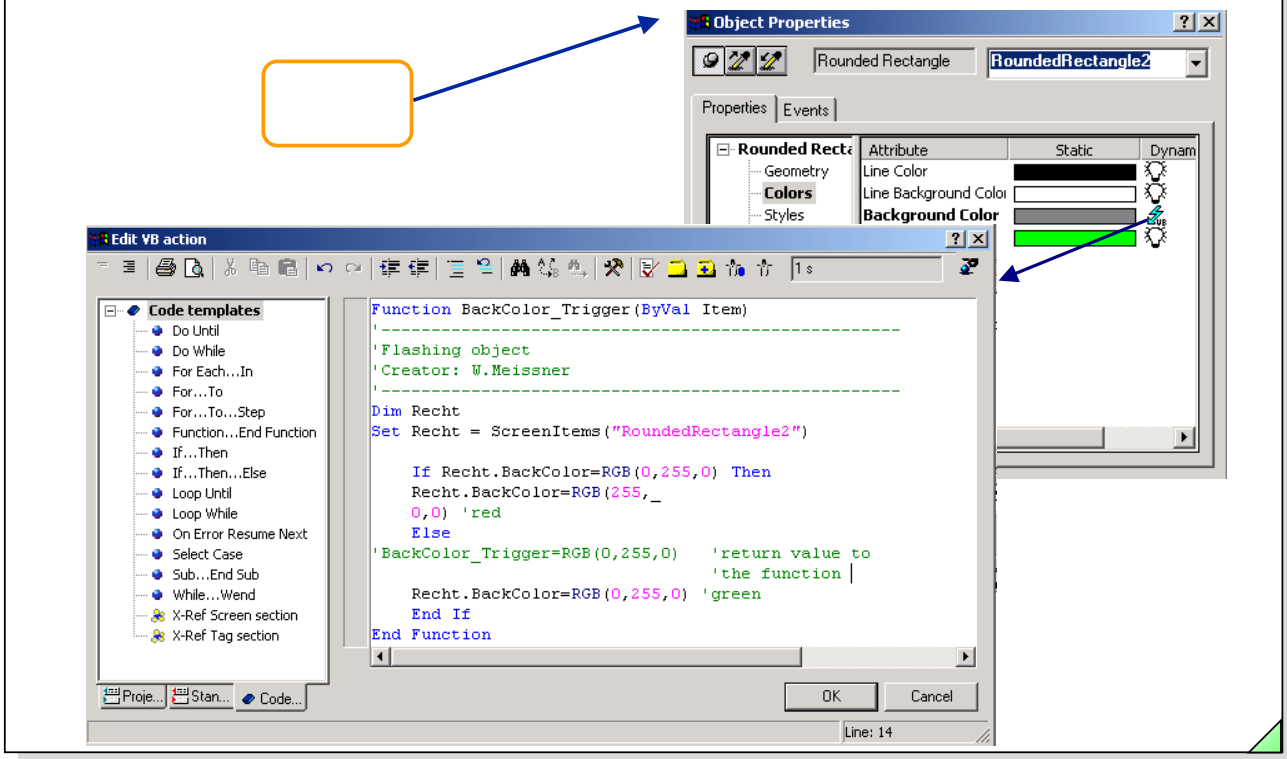
Error Dialog

You can enable the error dialog in the computer properties.
You can also activate the debugger in the run-time window.

Exercise 5

See the notes at the end of the chapter.

Overview of Property Actions with VBS



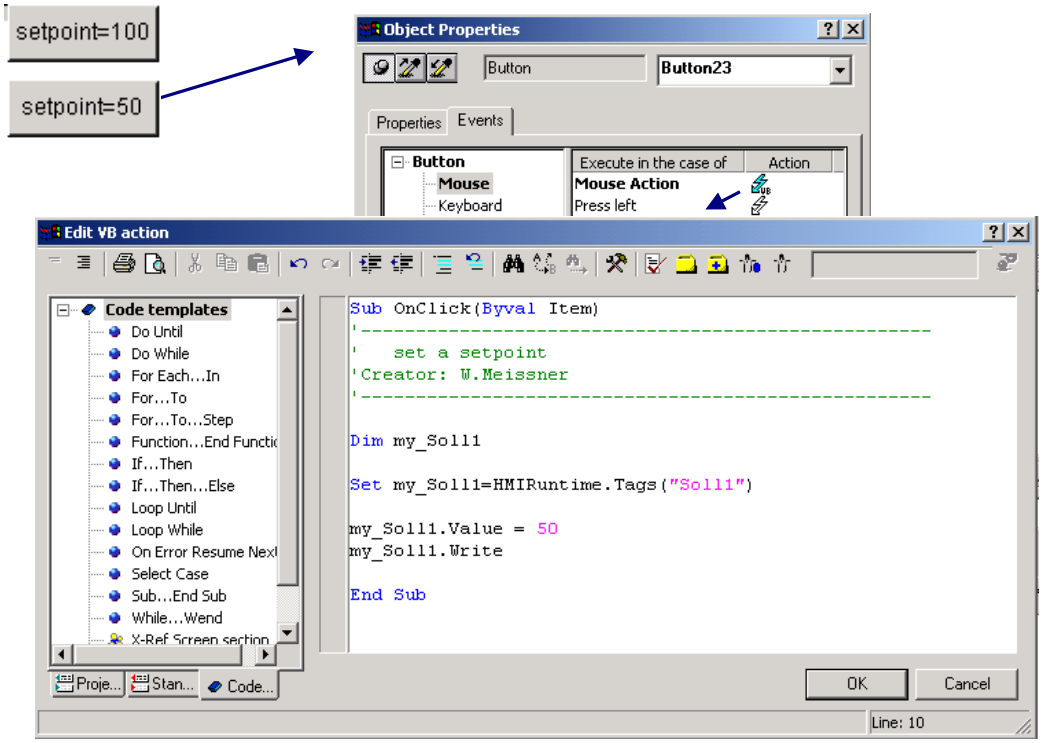
Date: 27.10.2003
File: SWINCC_04e.49

Task

The rounded rectangle is to flash alternately red and green. The flashing frequency is 1 second. The function can be supplied with the return value or you can supply the object through the property “BackColor“.

.

Overview of Event Actions with VBS



Task The temperature for a boiler is to be adjustable in fixed steps using the setpoint button.

.

Examples with VBS Actions

```
Example deactivate RT
Dim my_Soll1
Set my_Soll1=HMIRuntime.Tags("Soll1")
my_Soll1.READ
If (my_Soll1.Value >= 50) Then
        HMIRuntime.Stop
End If
End Sub
```

```
Read a tag from the Process Image
Dim myTag
Set myTag = HMIRuntime.Tags(“istwert")
value = myTag.Read

write
myTag.Write
```

```
Example to display a Helptext
Dim Recht
Set Recht = ScreenItems("StaticText2")
If Recht.Visible=0 Then
        Recht.Visible=1 ‘Visibile
Else
        Recht.Visible=0  ‘Invisible
End if
End Sub
```

```
In this example a value is written in Soll1
Dim my_Soll1
Set my_Soll1=HMIRuntime.Tags("Soll1")
my_Soll1.Value = 0
my_Soll1.Write (vbFalse)

or

Dim my_Soll1
my_Soll1.Write (5, vbFalse)

The Parameter "vbFalse" means, that the tag is written
synchron, this is directly in the process.
```

```
Examle for collecting Status Istwert_1
Dim myTag,last_err
Set myTag = HMIRuntime.Tags(“Istwert_1")
last_err = myTag.LastError
If (last_err > 0) Then
Msgbox myTag.QualityCode
End If
```

SIMATIC HMI

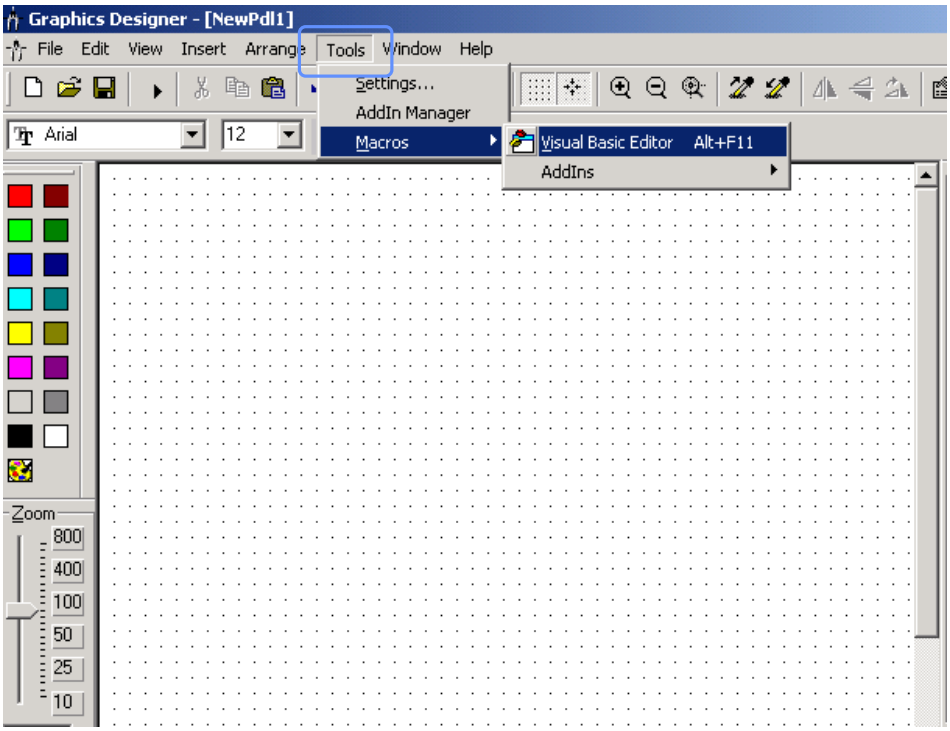
Date: 27.10.2003
File: SWINCC_04e.51

SITRAIN Training for Automation and Drives

VBS Actions	You can edit very extensive actions, checks and queries with VBS actions.
Configuration	VBS actions can be used everywhere where the direct connection to individual tags is not sufficient, or if several properties are to be modified at the same time. With the help of VBS actions, the user gains an extensive access to all available object properties and even the tag contents of WinCC.
Procedures	You can use procedures to consolidate repeating or frequently used functions.
Sub Procedures	A sub procedure is a code block that is executed as a result of an event but doesn‘t deliver a return value.
Local Script Tags	Local script tags are to be defined with the Dim, Private, or Public instruction and are only valid within the VBS action.
Picture-specific Script Tags	You can declare picture-specific tags in the declaration area of the action window.

.

Activating the VBA Editor (Development Environment)



VBA

With VBA you expand the functionality of the Graphics Designer and automate the configuration. The Graphics Designer has a VBA Editor with which you can automate the configuration of pictures. The VBA Editor is identical to the one in the products of the Microsoft Office family.

Starting the VBA

1. In the menu bar, click on the Tools menu
2. In the drop-down menu, click on Macros- ->VisualBasic Editor

Overview VBA

```
'In this example, a short message is to be displayed when a picture is opened. The "Opened" event is
used for this, after that a circle is to be created:

        Private Sub Document_Opened(CancelForwarding As Boolean)

        Dim objVariableTrigger As HMIVariableTrigger
        Dim objCircle As HMICircle
        MsgBox ("Bild wurde geöffnet!")

        Set objCircle = ActiveDocument.HMIObjects.AddHMIObject("Circle1", "HMICircle")
'Create dynamic with type "direct Variableconnection" at the
'property "Radius":

        Set objVariableTrigger =
        objCircle.Radius.CreateDynamic(hmiDynamicCreationTypeVariableDirect, "'Soll1'")

'To complete dynamic, e.g. define cycle:

        With objVariableTrigger
        .CycleType = hmiVariableCycleType_2s
        End With

        End Sub
```


VBA Examples

+User-specific menus through which, for example, a VBA macro is called (similar to WinCC Dynamic Wizards).

+Customized fast configuration dialogs for your own objects, that are automatically shown when an object from the library is inserted.

+Automatic generation of tags, messages, archive tags and triggers when a user-specific object is inserted in the picture.

+Automatic plausibility check for tag existence when closing or saving a picture.

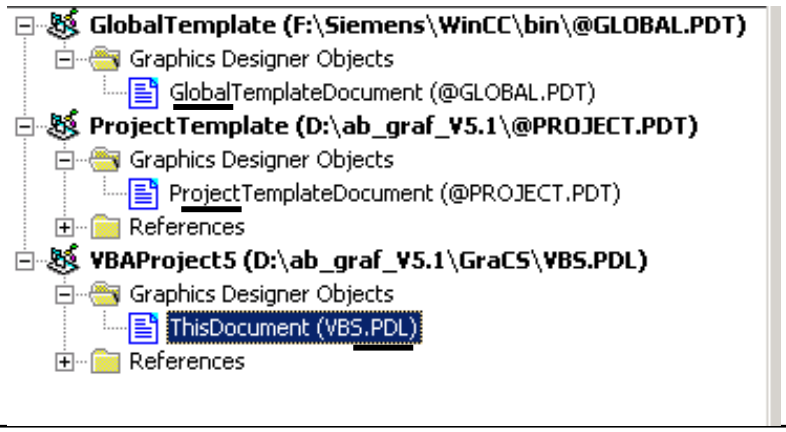
+Importation of external data (*.csv files, MS Excel, databases) from which pictures and picture objects are automatically created.

+Exportation of picture data in any format.

VBA Document

VBA Template

.



Comparison of VBA and VBS

	VBA	VBScripting
Language based on VB	Yes	Yes
Debugging	Yes	Yes
Simple Access to other application	Yes	Yes
Integrated functionality	Yes	Yes
Runtime Environment	WinCC CS: Graphics Designer	WinCC RT: Graphics Designer Global Script
Access to (COM Object Model)	WinCC CS: Graphics Designer, Tags, Alarms, Archiv, Text	WinCC RT: Graphics Designer, Tags
Alternative to	Dynamic Wizard and ODK	C-Scripting and ODK

-Standard Microsoft VBA 6.3 development environment
-VBScript is a subset of Visual Basic for Applications (VBA)

WinCC VBA

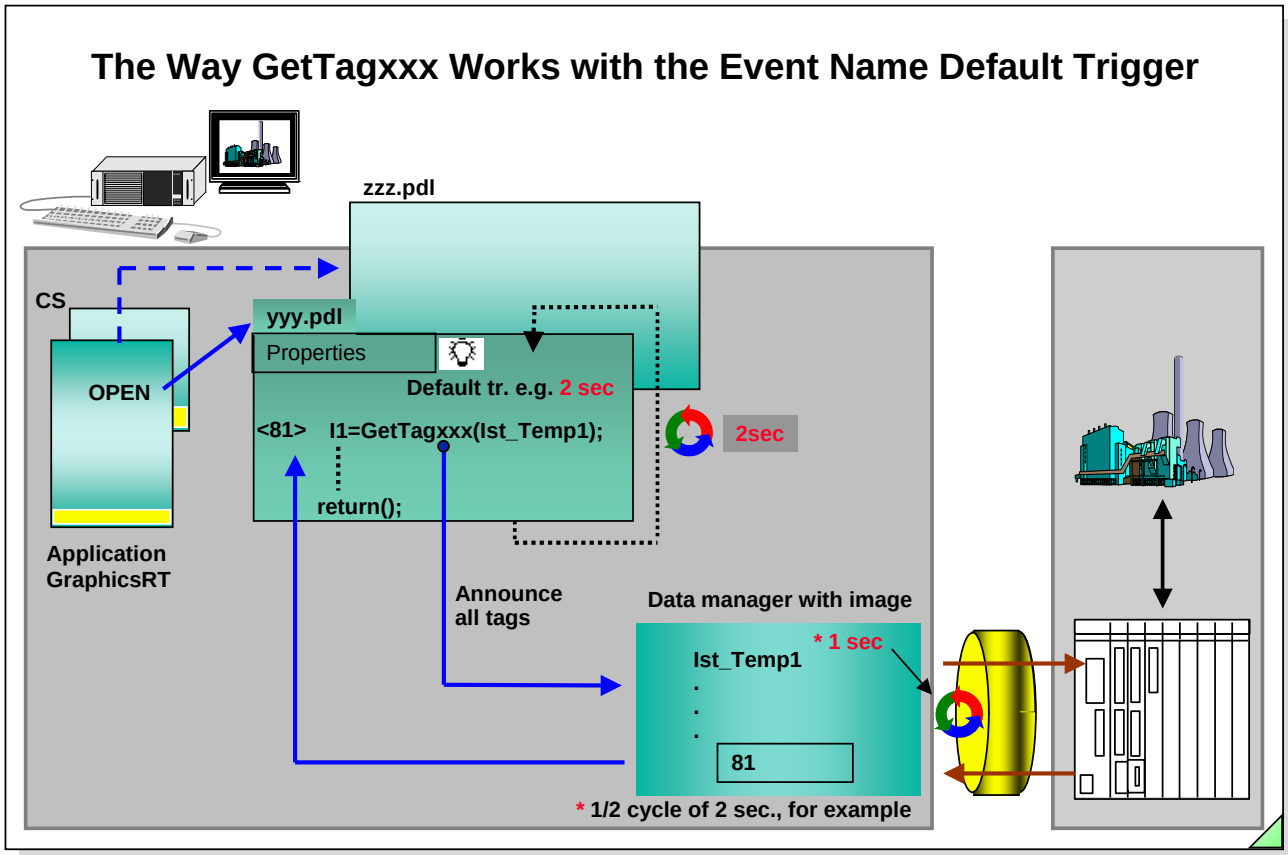
VBA provides the possibility of automating engineering tasks in the WinCC Graphics Designer, for example, creating user-defined menus and toobars; creating and editing Standard, Smart and Windows objects; accessing products that support VBA; action configuration for pictures and objects, etc.

WinCC VBS

VBS Scripts are based on the Microsoft Scripting host that is installed in every Windows 2000, Windows XP and Windows ME System.

For example, configuring setpoints for tags for the operation of a graphic object. Making the switching of the Runtime language dependent on the operation of a graphic object. Configuring color changes, for example, cyclically (flashing) or to display states (motor on). Transferring data to other applications (such as, Excel, Word, Access). Starting external applications from WinCC, etc.

.



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.55

SITRAIN Training for Automation and Drives

Objective

To understand the way different GetTagxxx functions work.

Task

In the Global Script of WinCC there are various functions for reading tag values. This gives the user greater flexibility for solving tasks.

GetTagxxx

The picture yyy.pdl is called using the Graphics Runtime application. In this picture, the GetTagxxx function is called using a graphics object.

Example: **I1 =GetTagxxx(Ist_Temp1);** Note: xxx=data type
if(I1 >100)

The calling function is executed every 2 seconds, for example. During the first call, the GetTagxxx function is introduced with half of the cycle value (here 1 second) to the data manager and from then on is cyclically fetched by the PLC. This makes sure that the value is available with every further call of the function. The tag is known in the data manager for the entire execution time (as long as the picture yyy.pdl is selected) and can be used for further evaluations. When the picture yyy.pdl is replaced with the picture zzz.pdl, all tags are logged-off with the data manager.

with Changes

When the default trigger is set for “change“, the trigger is switched to 1 second, for example. This default trigger is version and channel dependent.

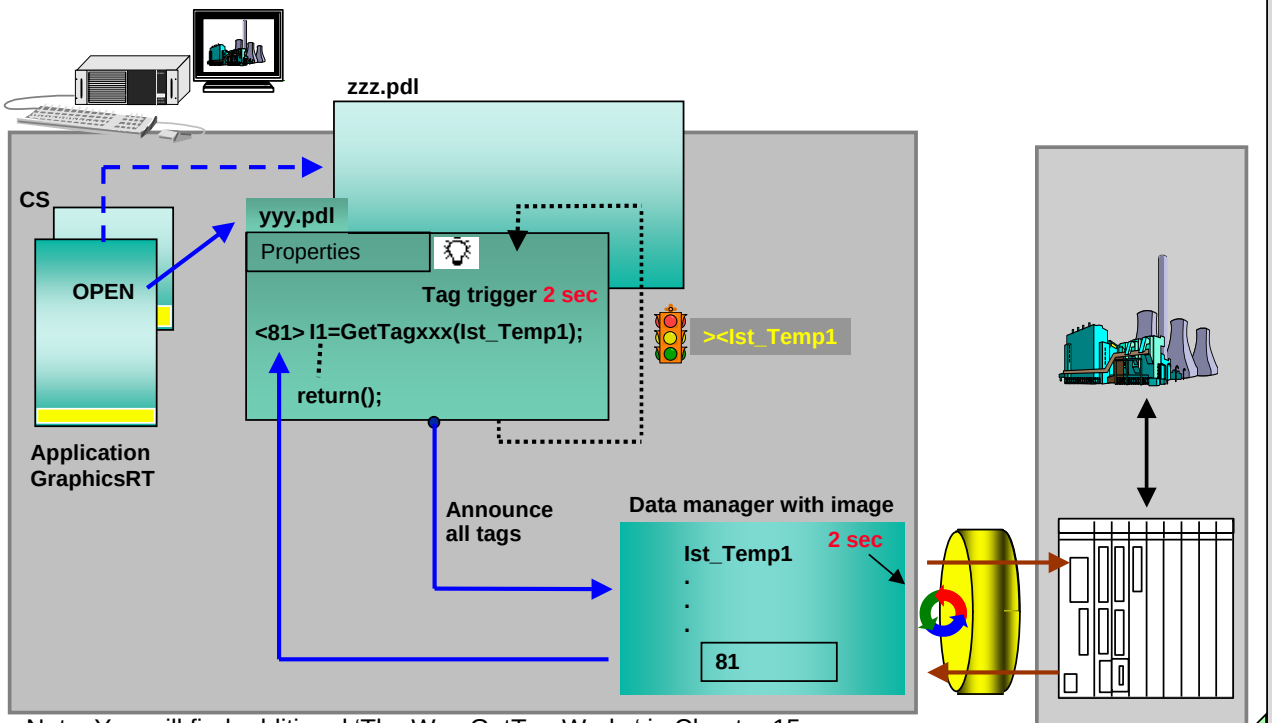
Properties

-The call has a return value that corresponds to the value from the data manager (**asynchronous reading**).
-The calling function is executed cyclically.
-The function does not provide any information on the status of the reading job.

GetTagxxxState

The function has the same characteristics as GetTagxxx. The GetTagxxxState function provides additional information about the status of the reading job. Since the status is always provided internally, there is no differentiation in performance to GetTagxxx.

The Way GetTagxxx Works with the Event Name Tag Trigger



Note: You will find additional 'The Way GetTag Works' in Chapter 15

Date: 27.10.2003
File: SWINCC_04e.56

Objective

To understand the way the GetTagxxx function works with tag triggers.

Task

In the Global Script of WinCC there are various functions for reading tag values. This gives the user greater flexibility for solving tasks.

GetTagxxx

The picture yyy.pdl is called using the Graphics Runtime application. All tags contained in the tag trigger are introduced during picture selection and from then on are fetched cyclically with the cyclic value (here 2 seconds) by the PLC. The GetTagxxx function is called in this picture using a graphics object.

Example: **I1 =GetTagxxx(Ist_Temp1);** Note: xxx=data type
if(I1 >100)

The calling function is only then processed every 2 seconds, for example, when the data manager has determined a change in the function value. The tag is known in the data manager for the entire execution time (as long as the picture yyy.pdl is selected) and can be used for further evaluations. When the picture yyy.pdl is replaced with the picture zzz.pdl, all tags are logged-off with the data manager.

Note

If a tag is called that is not in the tag trigger, then the behavior is the same as with the default trigger.

Properties

-The same as GetTagxxx.
-The calling function is processed when there are changes.

Configuration Notes

Making things dynamic with

Performance high

-Properties

>tag connection

>faster dynamic dialog with tag trigger and a tag that is assigned parameters

>slower dynamic dialog with tag trigger and several tags that are assigned parameters or C-actions that are called (Global Script Function)

Performance low

>C-actions, VBS actions

-Event

>a mouse click or object change (C-action or direct connection) has no effect on the picture selection time

Type of Update

-tag trigger (use the same cycle for all tags)
-select a uniform update of a picture
-avoid using Change

Functions

-SetTagMultiWait for several tags is better than many SetTagxxx
-GetTagxxx (tag is cyclically introduced to the data manager) ,SetTagxxx

Providing PLC Data

-For optimized communication, the data areas of the PLC are to be supplied for specific pictures.

SetTagMultiWait The values of several tags are set in a write job. The function is only completed after the PLC signals back that it has accepted the value.

Example

```
BOOL ok;

ok=SetTagMultiWait("%d%d%f",
“Sollwert1", 16,
“Sollwert2", “500",
“Sollwert3", 55.4711);
```

From the format writers, the following type is expected:

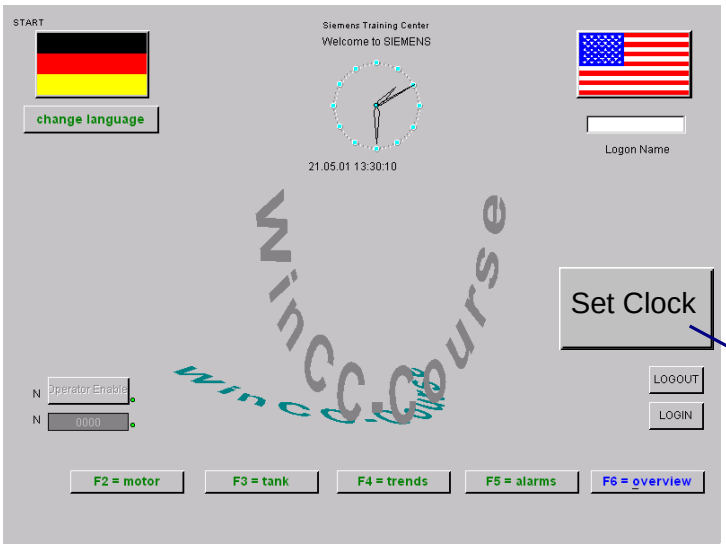
%d = DWORD / Int

%f = double

%s = char*

Note In the access to the peripherals of the inputs and outputs, you must expect a much longer response time.
Help: Addressing using bit memories or data blocks.

Setting and Displaying Date and Time



The ProgramExecute function is created with Dynamic Wizard, System Functions tab, Call: **Start another application**.

ProgramExecute("C:\\WINNT\\System32**Control.exe *timedate.cpl***");

Note: The directory where Control.exe is located depends on the installation of the Windows operating system.

Date: 27.10.2003
File: SWINCC_04e.58

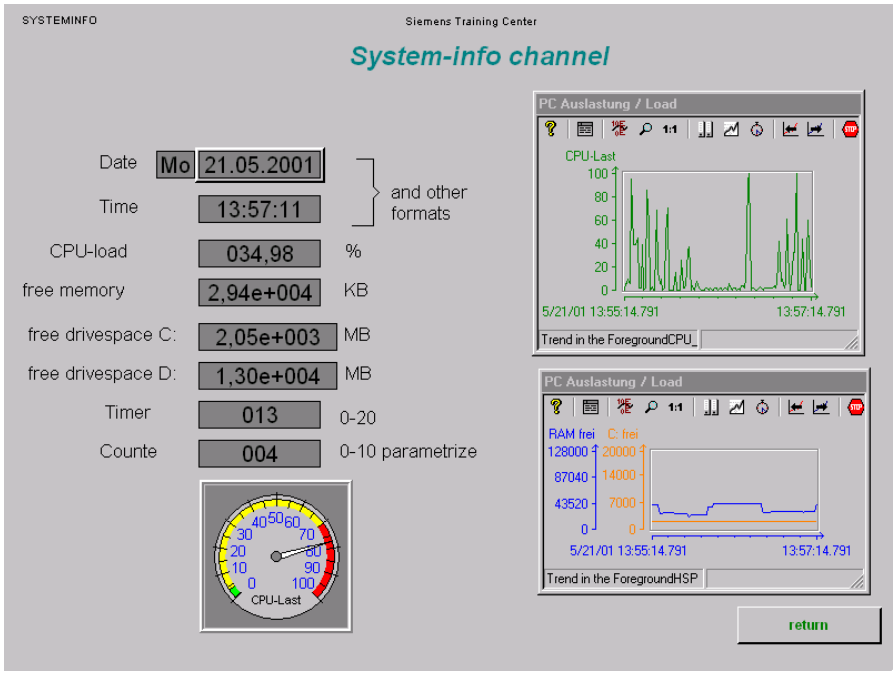
Dynamic Wizard

After the script **ProgramExecute ("C:** WINNT\\System32\\ **\\Control.exe");** is created with the Dynamic Wizard, the script has to be extended with the parameter **timedate.cpl**.

In Runtime

After activating the "Set Clock" button, the window to change the clock will be displayed.

System-info Channel 1/3



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.59

SITRAIN Training for Automation and Drives

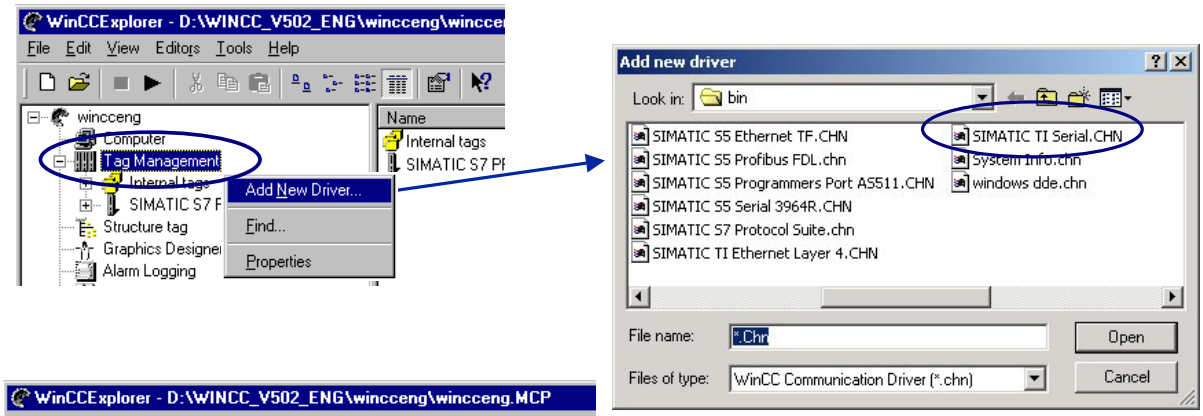
Objective

The Systeminfo-Channel gives you the opportunity to define, through a channel, different system information using internal tags and to link this information with input and output objects.

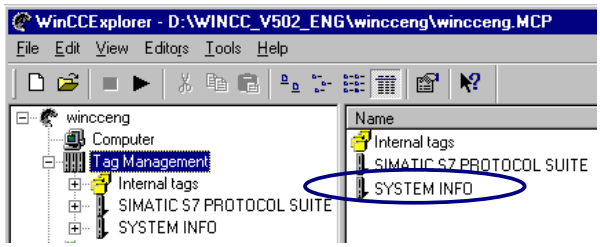
-Already included as of Version 5.0 SP1.

Call System-info Channel 2/3

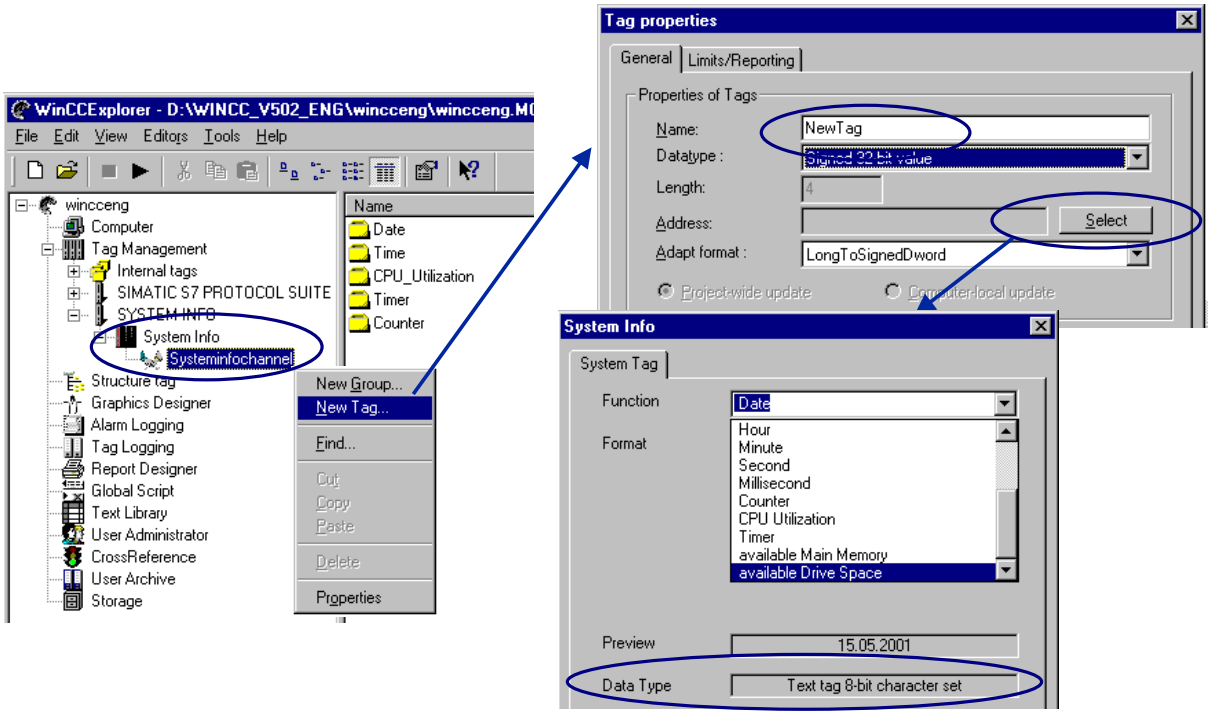
A)



B)



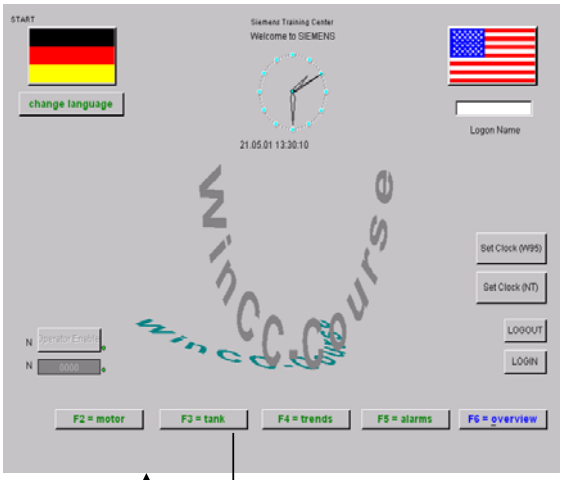
System-Info - Edit Variable 3/3



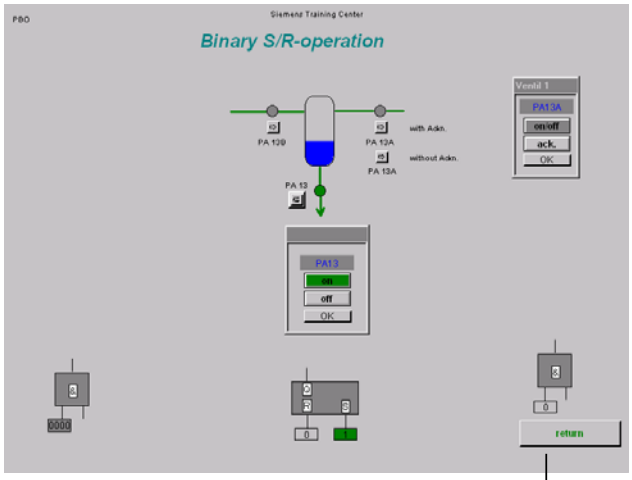
Note Every tag (variable) that is edited is counted as an external tag.

Configuring Picture Change (Configuration Dialog)

start.pdl



schalt.pdl



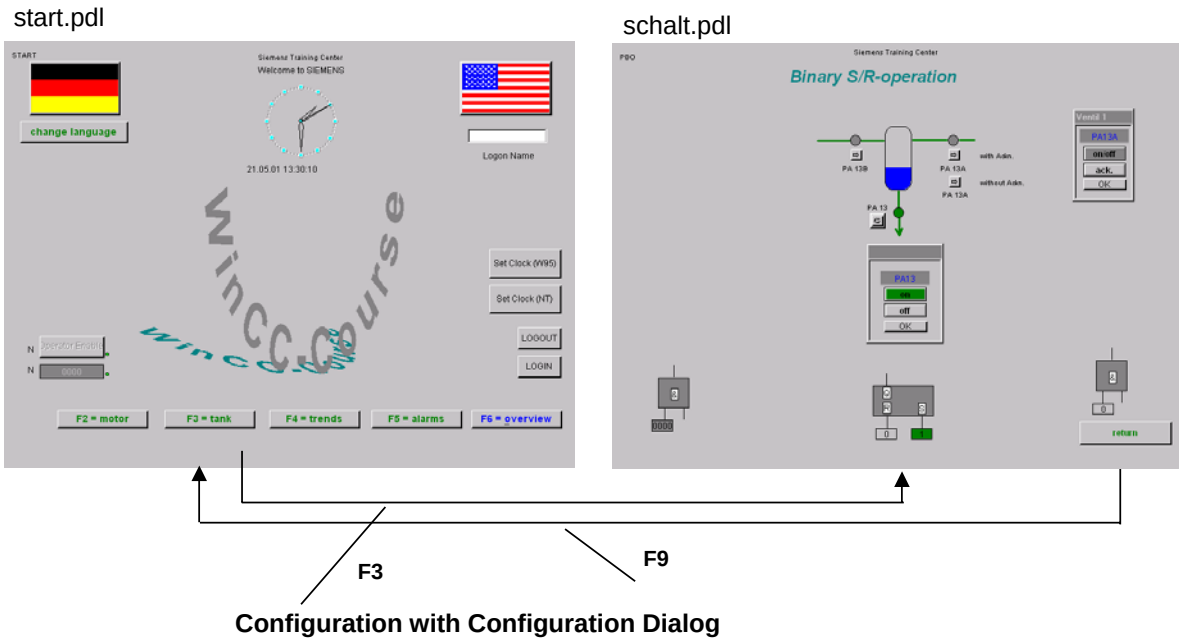
Configuration with Configuration Dialog

Date: 27.10.2003
File: SWINCC_04e.62

SITRAIN Training for Automation and Drives

Task	The mouse is to be used to change between two plant simulation diagrams.
Example	Operator elements (buttons) must be configured and correctly labeled in each picture.
Procedure	You can use, for example, the Configuration Dialog, the Dynamic Wizard, or action configuration for the Picture change. Parameters are set for the respective Picture Name in the configuration. If you use the Configuration Dialog, it generates a direct connection: With the Dynamic Wizard, a C-script is generated.
C-Script	1. Button -Event/ Button/ Mouse/ Press left/ Action: **OpenPicture("schalt.pdl");** 2. Button -Event/ Button/ Mouse/ Press left/ Action: **OpenPicture("start.pdl");**

Configuring Function Keys (Configuration Dialog)



Configuration with Configuration Dialog

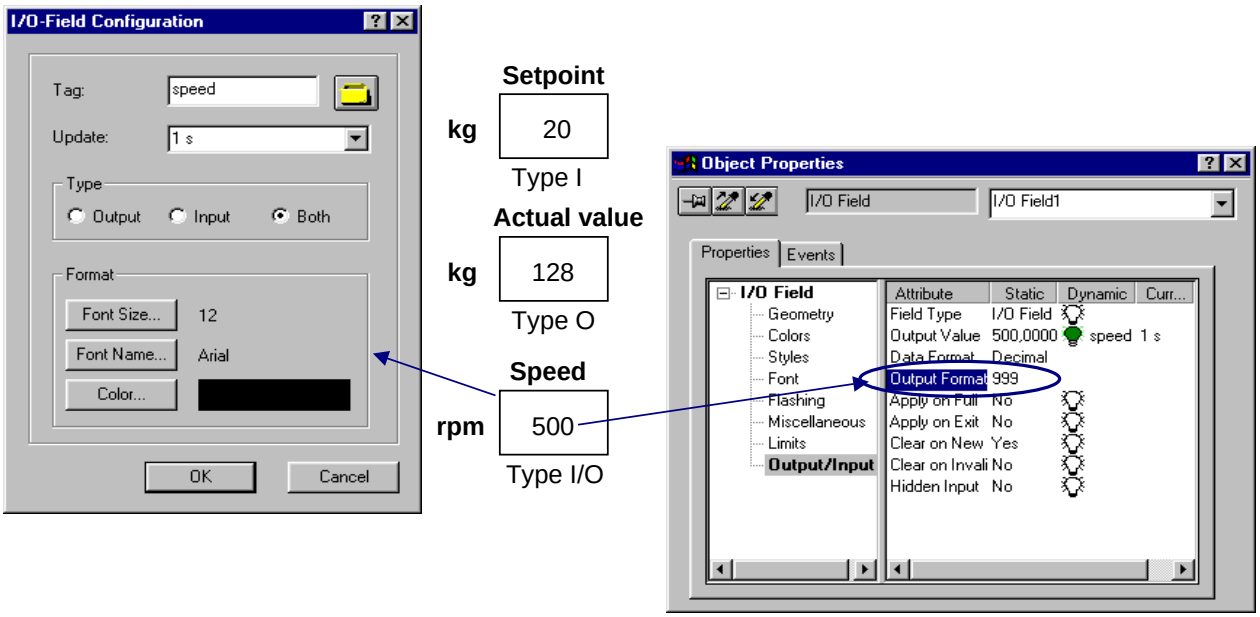
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.63

SITRAIN Training for Automation and Drives

Task	The mouse or keyboard is to be used to change between two plant simulation diagrams.
Example	Operator elements (buttons) must be configured and correctly labeled in each picture.
Procedure	The function keys and/or Hotkey are defined by using the Configuration Dialog. The above task can also be triggered using the object properties.
Object Properties	-Properties / Button/ Miscellaneous/Hotkey/ Static, such as **F3** -Properties / Button/ Miscellaneous/Hotey/ Static, such as **F9**
C-Script	-Event/ Button/ Mouse/ **Mouse-click**/ Action: **OpenPicture("start.pdl");** -Event/ Button/ Mouse/ **Mouse-click**/ Action: **OpenPicture("schalt.pdl");**
Note	The mouse action is only possible with the left mouse-click (depending on the Button configuration for left or right-handed mouse operation in Windows). The mouse-click is not suitable for jogging mode operation.

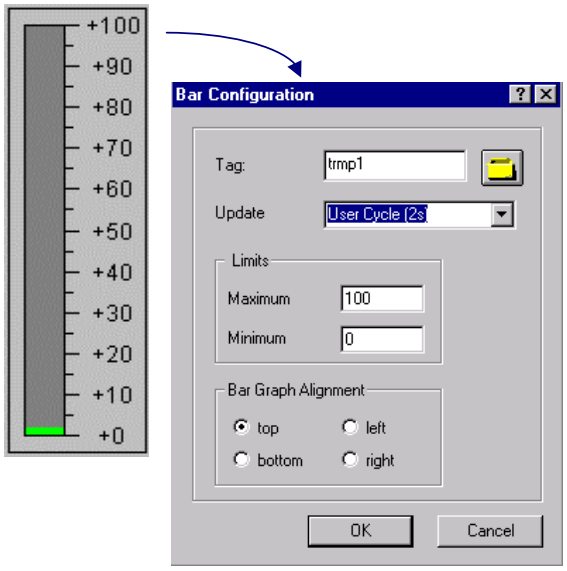
Digital Input/Output (Configuration Dialog)



Date: 27.10.2003
File: SWINCC_04e.64

Application	Input and combined input/output are used for process value adjustments, for example. Output objects and combined input/output objects are used to display process values, for example. All three field types can be defined with the **Configuration Dialog.**
Input-	Call the Configuration Dialog and set the Field Type: **Input** and the Tag: **Setpoint.** The setpoint, represented as Type I in the slide above, does not show the current process value (always 0) when a picture is selected.
Configuration	- Properties/ output and input / data format : **decimal** / output format: **999**
Output-	Call the Configuration Dialog and set the Field Type: **Output** and the Tag: **Actual Value**.
Configuration	- Properties/ output and input / data format: **decimal** / output format: **999**
Input/Output-Configuration	Call the Configuration Dialog and set the Field Type: **Both** and the Tag: **Speed**. - Properties/ output and input / data format: **decimal** / output format: **999**

Bar Graph Output (Configuration Dialog)



Configuring *additional* properties

- Properties/Miscellaneous/Process Driver Connection:		**temp1**
	/ Maximum Value:	**100**
	/ Minimum Value:	**0**
	/ Trend:	**no**
- Properties / Color	/ Bar Background Color:	**green**
	/ Trend Color:	**yellow**
- Properties / Font	/ Font:	**Arial**
	/ Font Size:	**12**
- Properties / Axis	/ Alignment:	**right**
	/ Bar Scaling:	**linear**
	/ Digits Left of Decimal Point:	**3**
	/ Digits Right of Decimal Point:	**0**

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.65

SITRAIN Training for Automation and Drives

Task

The temperature of a brewing procedure in a brewery is to be displayed graphically on a bar graph.
A trend display is to show whether the temperature value is rising or falling.

Example

The temperature display range lies between 0 and 100.
Configure the color green for the bar graph color.
The bar graph is connected with the tag, such as temp1, via the process connection.

Procedure

Select a bar graph in the object palette from the smart objects group. Configure the bar graph in your graphics screen with the help of the Configuration Dialog. Refer to the slide above for the other necessary settings.

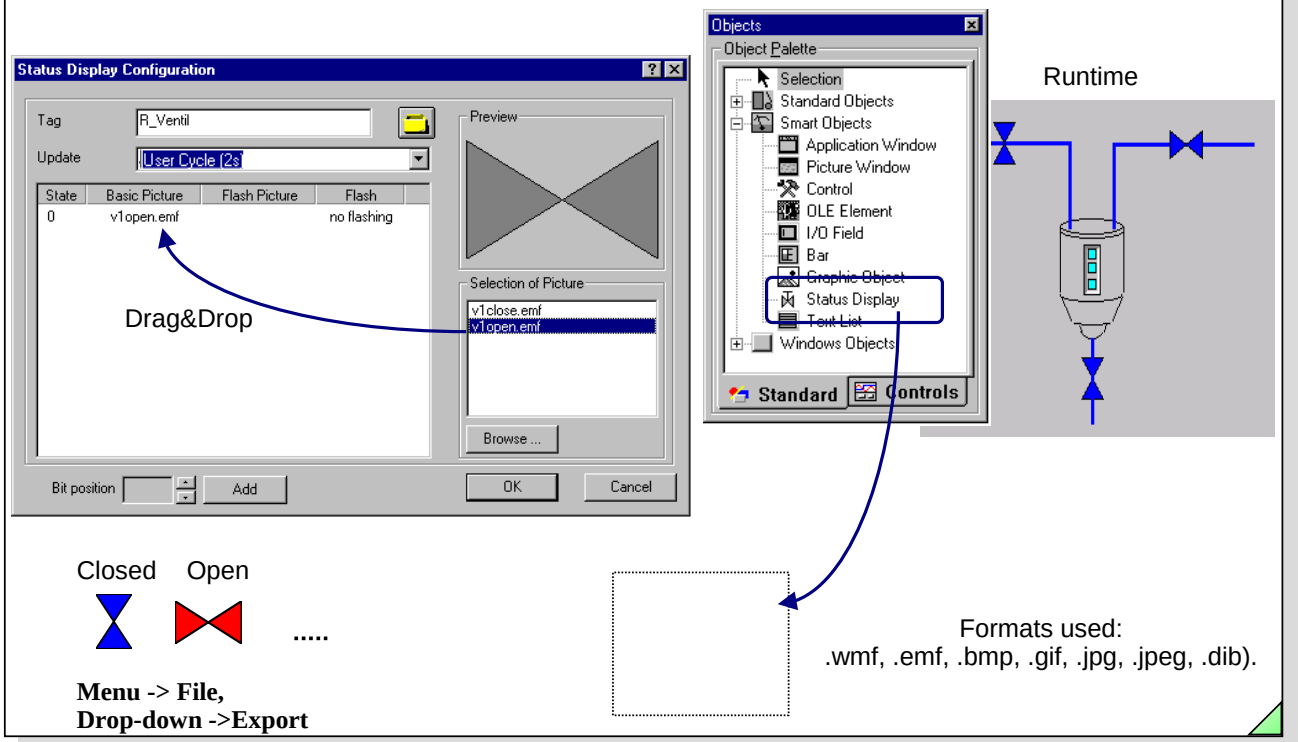
Color Switch-over

Color change, dependent on a digital value, can be configured via the "bar graph color" property and Dynamic Dialog. However it is better to use the bar limits to change the bar color totally or in segments.

Exercise 6

See the notes at the end of the chapter.

Status Display with Files



Task

The status of a valve, depending on the process, is to be displayed on the screen. The display change takes place through the object and attribute. There is always only one binary value available in a data byte for the scan in the status display.

Example

The status of a valve shows the operator the relevant system status: "closed" is blue, "open" is red, and "error" is flashing red.

Procedure

- Configure the three individual objects as displayed in the slide above, or, look for the appropriate icons in the library.

- Select (mark) the individual objects and export the files, using a logical name, into the GraCS directory. Use the **File** menu and the drop-down menu **Export**.

- Using the "Smart Objects" node, choose the status display from the object palette. .wmf, .emf, .bmp, .fig, .jpg, .jpeg, .dib files must first have been created as alternating objects.

These graphic files can also be created with other tools (for example, Paintshop Pro, MS Paint, etc.).

WMF files can be convertd with a tool that exists on the WinCC - CD. (**wmfdcode.exe**).

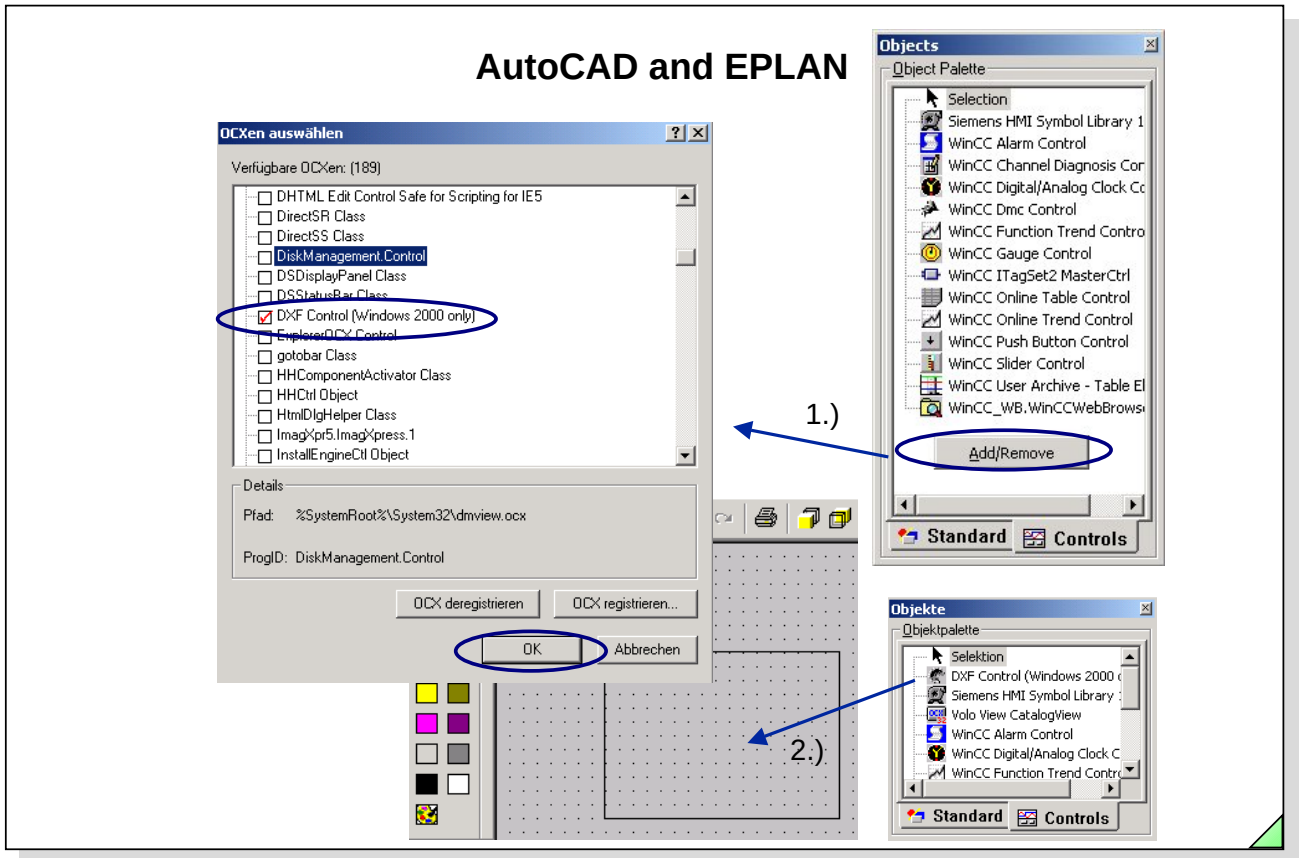
Note

By default, **yes** is set under **Properties/ Status/ Basic Picture Referenced**. That is, the individual status files (.emf, .wmf, .bmp) are loaded when there is a status change.

If you choose **Basic Picture Referenced: no**, then the status files are saved in the picture and are loaded when the picture is selected. That way, the updating time can be reduced. The system picture .pdl becomes larger, which may be noticed in the initial selection. This characteristic can be alleviated by hardware with a higher performance.

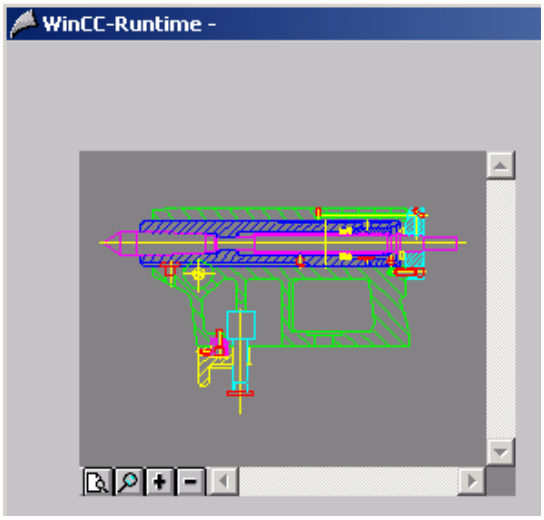
Exercise 7

See the notes at the end of the chapter.



Dxf Control

In order to be able to use the dxf Control, you have to install the Volo View CatalogView from the WinCC CD.



Overview of Configuration Possibilities for Color Changes with Accessory Examples

- Wizard, Color Dynamics with Table
- Dynamic Dialog
- C-Action
- VBS
- Tag Connection to the corresponding color property

Color Change

Various possibilities are available for changing object colors:
1. Dynamic Dialog
2. C action
3. Tag connection to the relevant color property (see page 70, Option Group)

Application

The color change of objects is used to signal states, such as:
- Circuit state
- Limit value overrange
- Process connection status

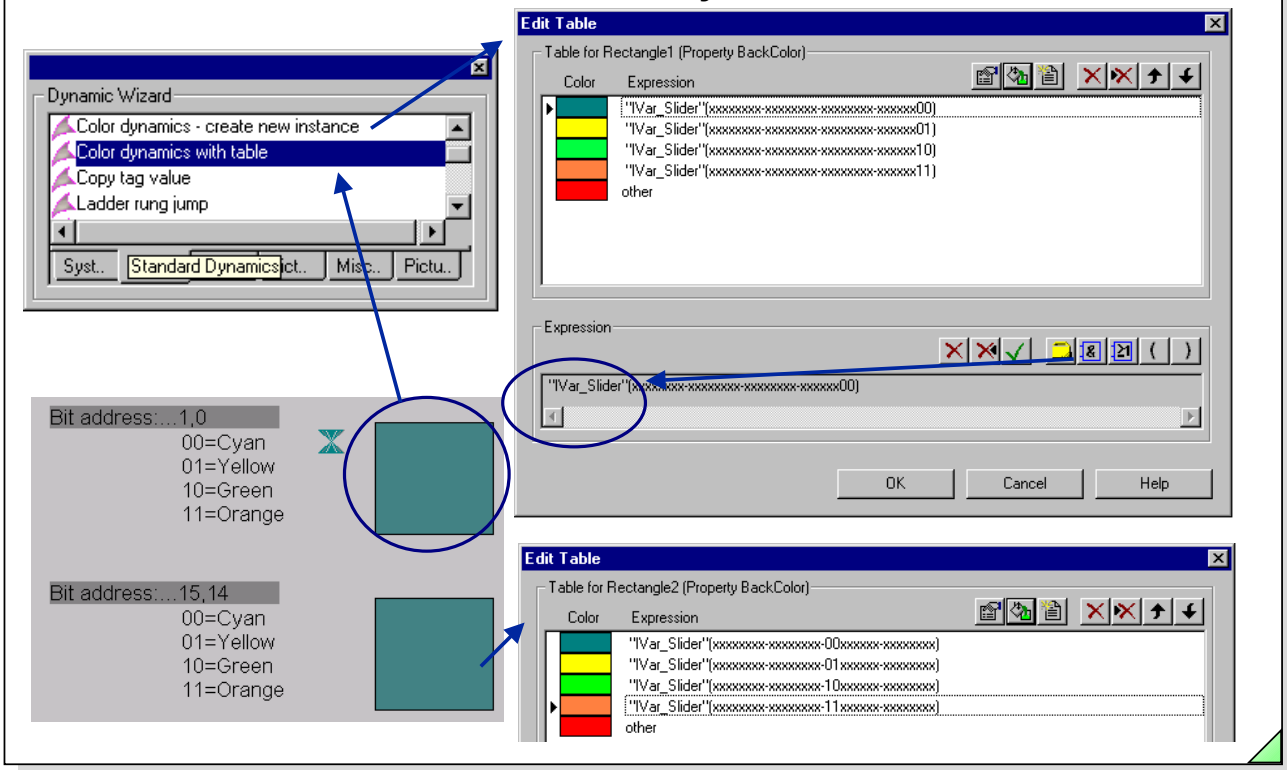
Color Values (Tones)

The color values are created from a large palette. The 16 basic colors are, for example:

Color	**Color Value**	**Symbolic Constants** (with #define ext. pre-defined)			
			Dark		
Red	0x000000FF:	CO_RED	Red	0x00000080:	CO_DKRED
Green	0x0000FF00:	CO_GREEN	Green	0x00008000:	CO_DKGREEN
Blue	0x00FF0000:	CO_BLUE	Blue	0x00800000:	CO_DKBLUE
Cyan	0x00FFFF00:	CO_CYAN	Cyan	0x00808000:	CO_DKCYAN
Yellow	0x0000FFFF:	CO_YELLOW	Yellow	0x00008080:	CO_DKYELLOW
Magenta	0x00FF00FF:	CO_MAGENTA	Magenta	0x00800080:	CO_DKMAGENTA
Lt. Gray	0x00C0C0C0:	CO_LTGRAY	Gray	0x00808080:	CO_DKGRAY
Black	0x00000000:	CO_BLACK	White	0x00FFFFFF:	CO_WHITE

Mixed colors are intermediate tones in the palette. The object colors are set by supplying the object properties with the corresponding values.

Wizard Overview, Color Dynamics with Table



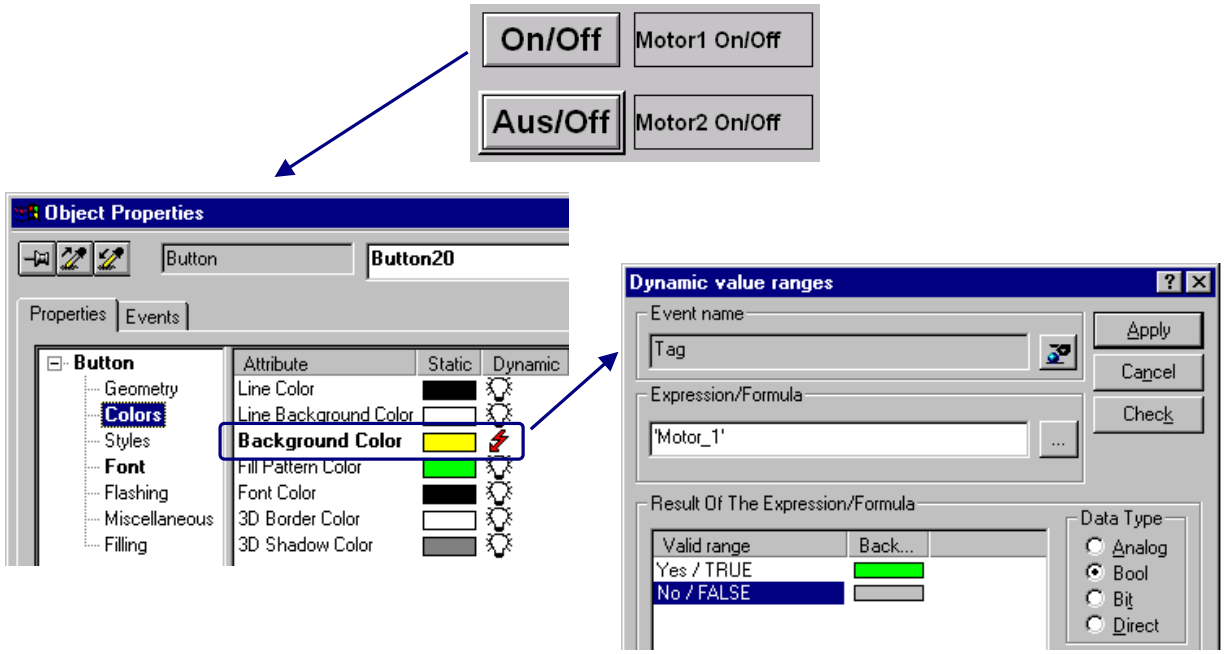
Date: 27.10.2003
File: SWINCC_04e.69

Objective Configuration of color changes from bit combinations of various tags.

Note

-The Script generated by the Wizard can also be used for other visualizations (object changes, for example).
-The Wizard can also be called again later for changes.
-The list entries are worked through from top to bottom.
-Maximum of 10 list entries.
-Already included in Version 5.0 SP1.

Overview of Color Change with the Dynamic Dialog



Date: 27.10.2003
File: SWINCC_04e.70

Task

Switching a motor on and off and the motor's circuit state are to be displayed.

Components

A button object is used for the momentary contact switch. This button object is located in the Windows objects for the object palette in the Graphics Designer. Operation is carried out when a key is pressed. The "off" state is identified in gray, and the "on" state is identified in green.

Button On/Off

Properties/ Colors/ **Background Color**/ Right Mouse button in the Dynamic column/ Dynamic Dialog...

Expression/ Formula: Ein_aus1 (on_off1)
Data type: Bool
Result of expression: yes/ TRUE: *green*
no/ FALSE: *gray*

The relevant text is also displayed in a second object. The Dynamic Dialog can also be used for this purpose.

Button Off

Background color - see On/Off button

Properties/Font/ **Text**/ Right Mouse button in the Dynamic column/ Dynamic Dialog...

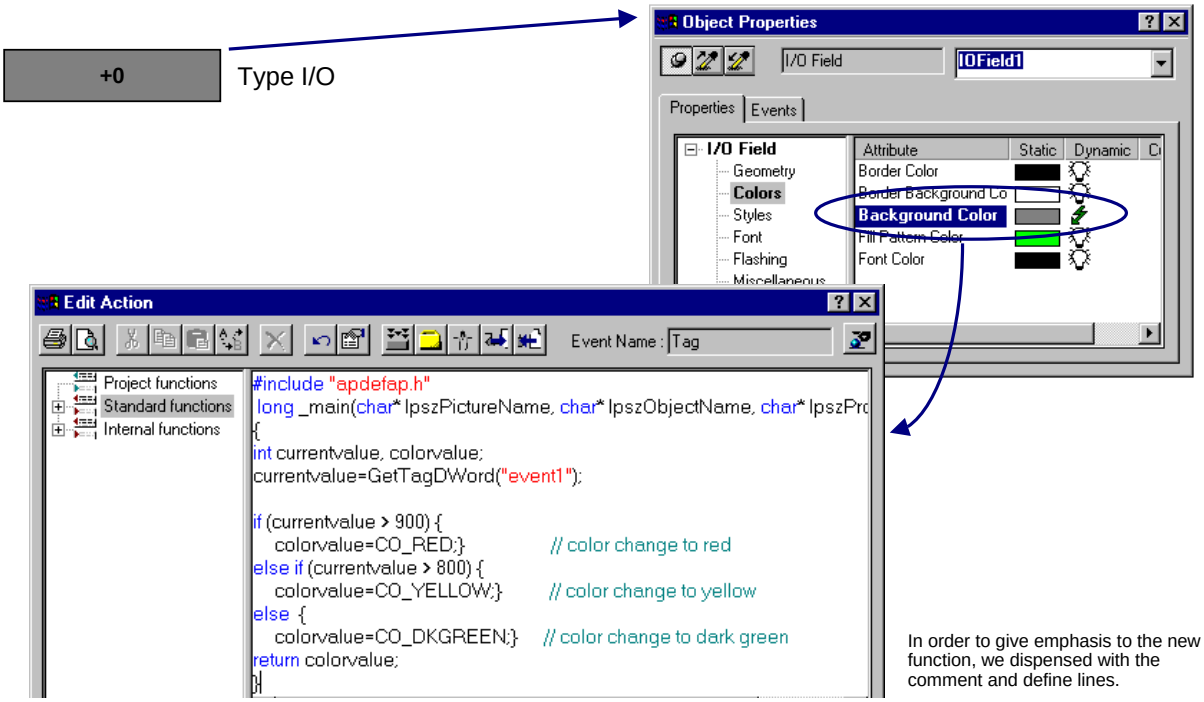
Expression/ Formula: Ein_aus2 (on_off2)
Data type: BOOL
Result of expression: Value range yes/TRUE *Off*
Value range no/FALSE *On*

Mouse Operation

Configuring:
Events/ Button/ Mouse/ Mouse-click/ C action..

```
BOOL value;
value = GetTagBit ("Ein_aus1");
value =! value;
SetTagBit("Ein_aus1 ",(WORD)value);
```

Overview of Color Change with a C Action



In order to give emphasis to the new function, we dispensed with the comment and define lines.

Date: 27.10.2003
File: SWINCC_04e.71

Task

A process tag is to be checked for limit violation.
The limit violation is to be made visible as a color change.
The tag accepts values between 0 and 1000.
The following color animations are to be configured:
1st. Value > 800: color animation from dark green to yellow
2nd. Value > 900: color animation from yellow to red

Procedure

An I/O field and a bar graph are required.
In addition, a scroll bar is configured for entering the value.
Both the I/O field and the bar graph are connected with the process tags.

I/O Field

The color change can be configured with the C action.
Properties/ I/O Field/ Colors/ **Background Color**/ Dynamic/ C action...

```
int currentvalue, colorvalue;
currentvalue = GetTagDWord("event1");
if (currentvalue > 900) {
  colorvalue = CO_RED; }                    // color change to red
else if (currentvalue > 800) {
  colorvalue = CO_YELLOW; }                 // color change to yellow
else {
  colorvalue = CO_DKGREEN; }                // color change to dark green
return colorvalue;
```

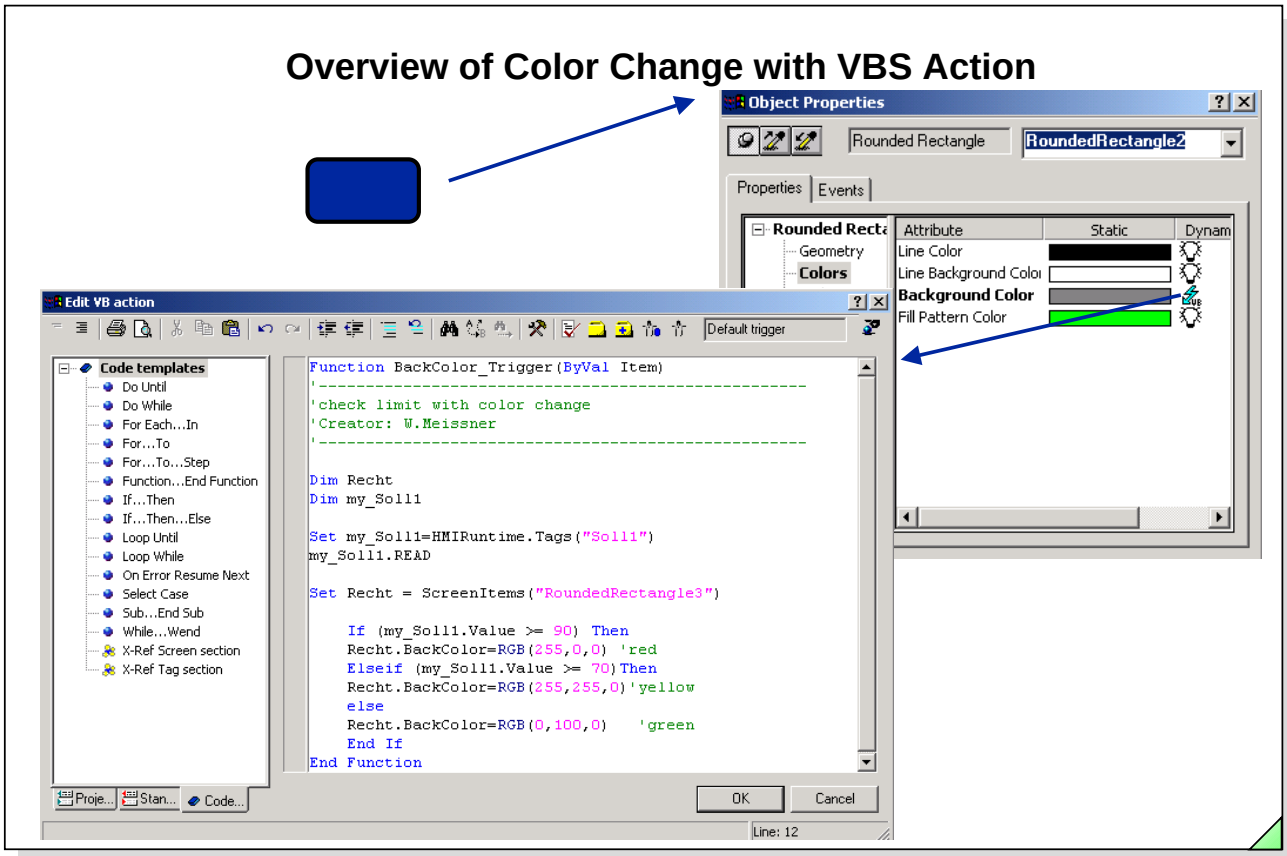
Bar Graphs

The same C action can be used for bar graphs.
The connection point is Properties/ Colors/ **Bar Color**/ Dynamic/ C action… With bar graphs, diverse limit values are also ready for use as an alternative color change.

Note

This task can also be solved with the Dynamic Dialog. The example in the slide above shows what you have to take into consideration when you work with a C-Program.

Overview of Color Change with VBS Action



Date: 27.10.2003
File: SWINCC_04e.72

Philosophy for On/Off Operations

Task		Solution
On/Off Ackn.	Toggle function with mouse-click with acknowledgement	C-action Direct Connection Dynamic Wizard C-action
On/Off	Toggle function with mouse-click without acknowledgement	C-action
On	Set Bit in var1 Reset Bit in var2	C-action
Off	Set Bit in var2 Reset Bit in var1	C-action
On Off	Set Toggle with mouse-click On Reset Toggle with mouse-click Off	Direct Connection Dynamic Wizard C-action
On/Off	Left / right mouse button for toggle	Direct Connection Dynamic Wizard C-action

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.73

SITRAIN Training for Automation and Drives

Switch Operation

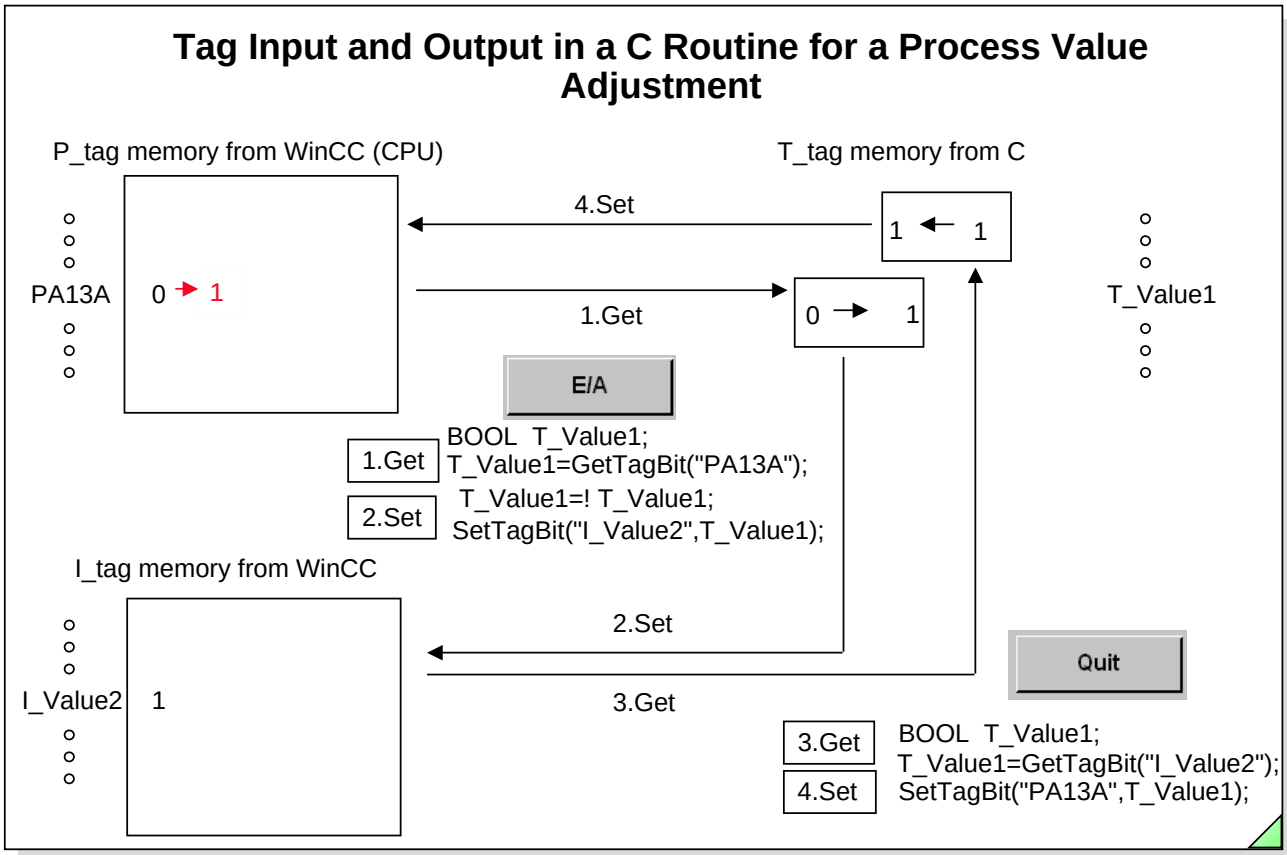
There are several possibilities available for switching on and off via objects.

1. Direct Connection
2. Dynamic Wizard
3. C-Action
4. Library Objects

The best solution for the system is the Direct Connection. Therefore this method should be used, if possible.

Example

The switching on and off of a unit should take place directly in the picture or in a two-step operation (picture window).



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.74

SITRAIN Training for Automation and Drives

Task

An input/output process value adjustment (bit value) is to be carried out using a button, if the operator provides an additional acknowledgement.

Procedure

Two buttons are configured. One button is used for the switching operation, and the other button is used for the acknowledgement (transfer). The action list for the switching operation is processed once for each mouse click (I/O button).
The current process value must be read with the Get function and saved in a temporary C tag.
Then the process value is negated and reloaded to the C tag.
Because the bit value for the process value adjustment is only written to the CPU after the acknowledgement, the value must be stored temporarily in an internal tag from WinCC using the Set Function so that it does not get lost. (The life-span of the C tag exists only during runtime.)

Reloading the process value from the internal tag takes place with the Set Function after the acknowledgement.

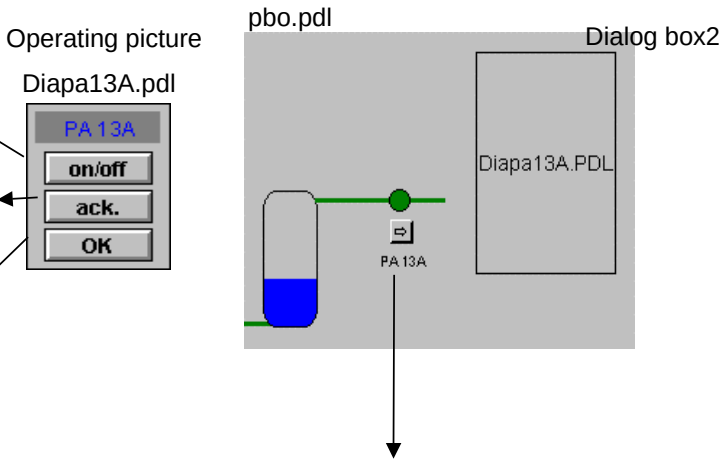
Advantages

A digital process value adjustment can be made with a few changes in the action list.
Digital adjustment and bit adjustment can take place directly in the picture or via a window that is temporarily displayed.
If the additional acknowledgement is not necessary in an application, then the temporary storing in an internal tag of WinCC is not necessary.
The bit value from the C tags can be written directly into the process tag with the Set Function.

Binary Switching Operation (Two-Step Operation) with Acknowledgement

Configuring with C-action:
Events/Mouse/Press Left/Action

```
BOOL  T_value1;
T_value1=GetTagBit("PA13A");
T_value1=!T_value1;
SetTagBit("I_value2",T_value1);
```



Configuration with Direct Connection:
Source: Tag *I_Value2*
Target: Tag *PA13A*

Configuration with Direct Connection:
Source: Constant *0*
Target: Current Window -> *Display*

Configuration with Dynamic Wizard:
Picture function
Display WCCDialog Box with *Diapa13A.PDL*

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.75

SITRAIN Training for Automation and Drives

Task

Switching a valve on and off is to take place with a button in a two-step operation. The switching operation **must** be acknowledged.

Example

A two-step operation (1. Select Button PA13A and 2. I/O button) is to be carried out for a switching condition 1-0-1...by means of a temporarily displayed picture.
The adjustment is not executed until the acknowledgement key is **also** operated.
The "OK" button is used to close the picture.
The switching operation corresponds to the binary value of a tag. The additional acknowledgement provides the operator with a greater switching safety.

Procedure

1. Configure the operating picture as shown in the screen in the slide above.
   The operating picture has three buttons.
   One of the buttons is for the switching operation, a second button is for the acknowledgement, and a third button is for closing the operating picture.
   When you create the Dialog box, the size of the operating picture (x and y dimension) and the dialog box must be the same.
2. Configure a button in the picture for calling the dialog box.
   The size of the dialog box (x and y dimension) must be the same as the operating picture. You use the Dynamic Wizard to generate the dialog box.

A binary type tag must be created for the value transfer.
If you want to have a title in the dialog box, the height of the dialog box must be 10 pixels more than the height of the .PDL file that is displayed in the dialog box.

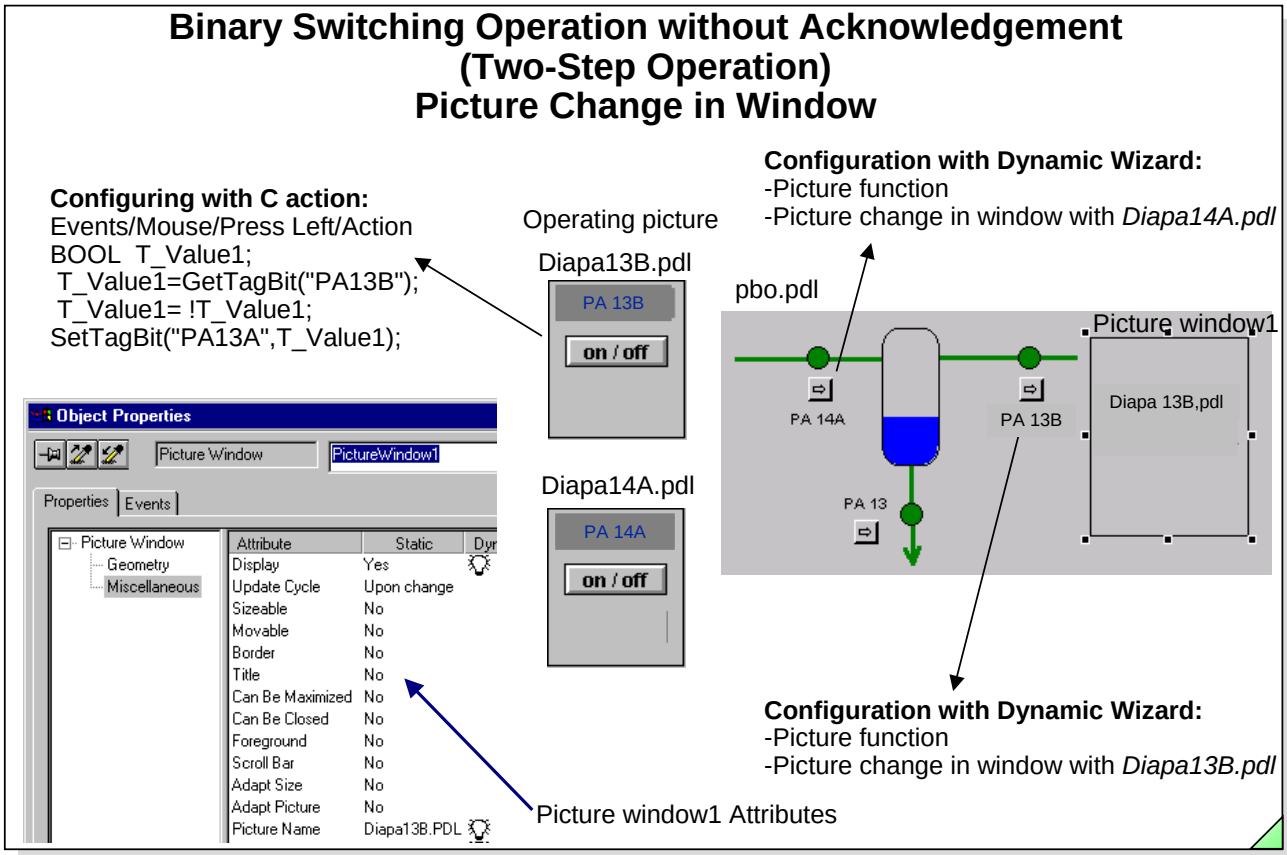
Note

The SetTagBit **cannot** be used for S5 switching operations since the handling blocks transfer the data word by word. The S7 coupling is an exception.
Remedy: Use bit setting in a word.

Exercise 8

See the notes at the end of the chapter.

Binary Switching Operation without Acknowledgement (Two-Step Operation) Picture Change in Window



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.76

SITRAIN Training for Automation and Drives

Task

Switching a valve on and off is to take place using a button in a two-step operation.

Example

A two-step operation through the Select Button PA13B is to be carried out for a switching condition 1-0-1...by means of a temporarily displayed picture.
The switching operation corresponds to the binary value of a tag.

Procedure

1. Configure the operating pictures as shown in the slide above. Every operating picture has one button for the switching operation. When you create the picture window, the size of the operating picture (x and y dimension) and the picture window must be the same.
2. Configure two buttons in the pbo.pdl picture for changing the contents in picture window 1. The size of the picture window (x and y dimension) must be the same as the operating picture. You use Object Palette --> Smart Objects to edit the picture window.

Two binary type tags must be created for the value transfer.
If you want to have a title in the dialog box, the height of the dialog box must be 10 pixels more than the height of the .PDL file that is displayed in the dialog box.

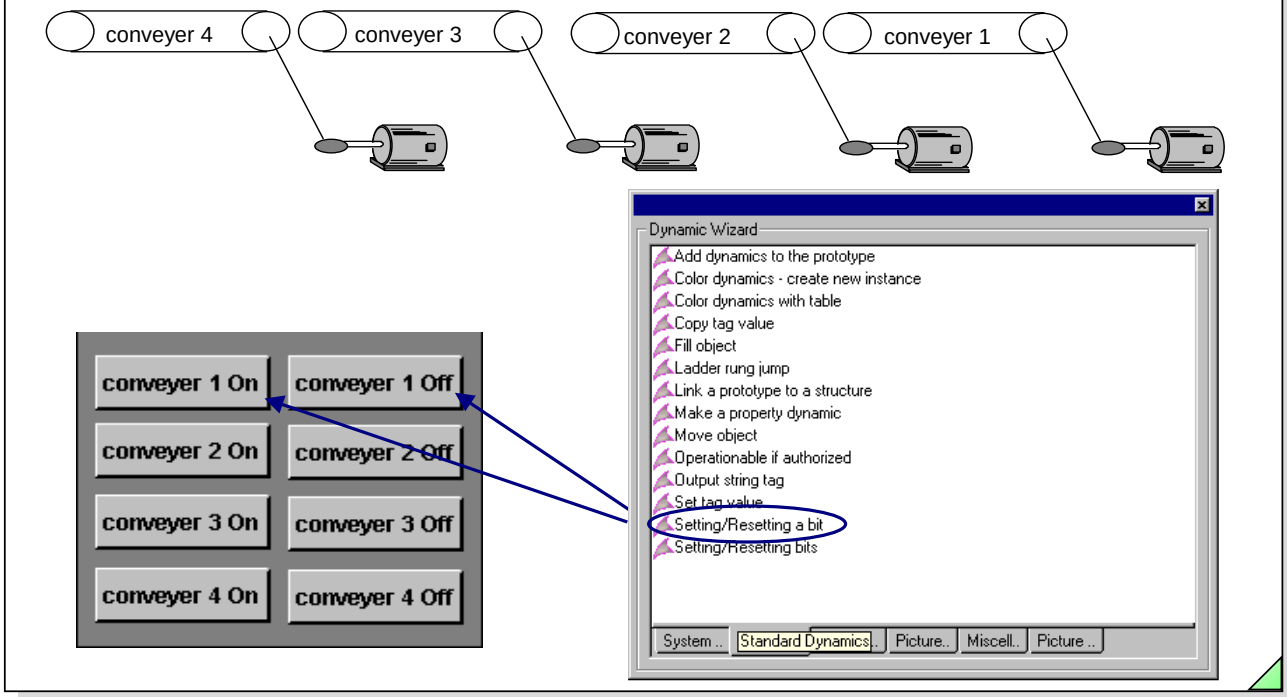
Further Application as Message Box

The graphic picture (Diapa13B.pdl) can be used as a message box, if the I/O button is deleted and suitable texts are entered (Limit values for adjustments 0-100!, for example).

Note

The SetTagBit **cannot** be used for S5 switching operations since the handling blocks transfer the data word by word. The S7 coupling is an exception.
Remedy: Use bit setting in a word.

Bit Setting in a Word with the Dynamic Wizard



Date: 27.10.2003
File: SWINCC_04e.77

Task

Four assembly conveyors are to be switched with 2 times four buttons via a bank of switches. An assembly conveyor motor is assigned to each button. Each motor can be switched on or off. The same process tag (conveyor) is to be used for the switching operation. This ensures that the memory locations in the PLC are optimally used.

Example

Each switching operation for a motor leads to setting and resetting a bit in a word. Bit addresses 0 to 3 are assigned. Bit addresses 4 to 15 can be used in the controller program for additional operations.

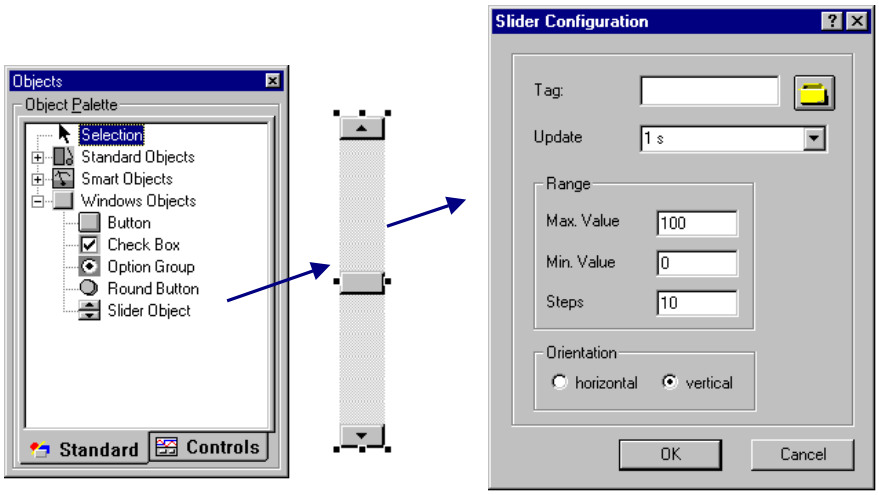
Procedure

The "conveyor" tag is defined as an unsigned 16-bit value in the data manager. Create the static and dynamic screen components according to the screen template in the slide above.

Exercise 9

See the notes at the end of the chapter.

Slider Object for Adjustments



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.78

SITRAIN Training for Automation and Drives

Task

A setpoint is to be entered using the slider object.
The control range lies between the minimum value and the maximum value.

Example

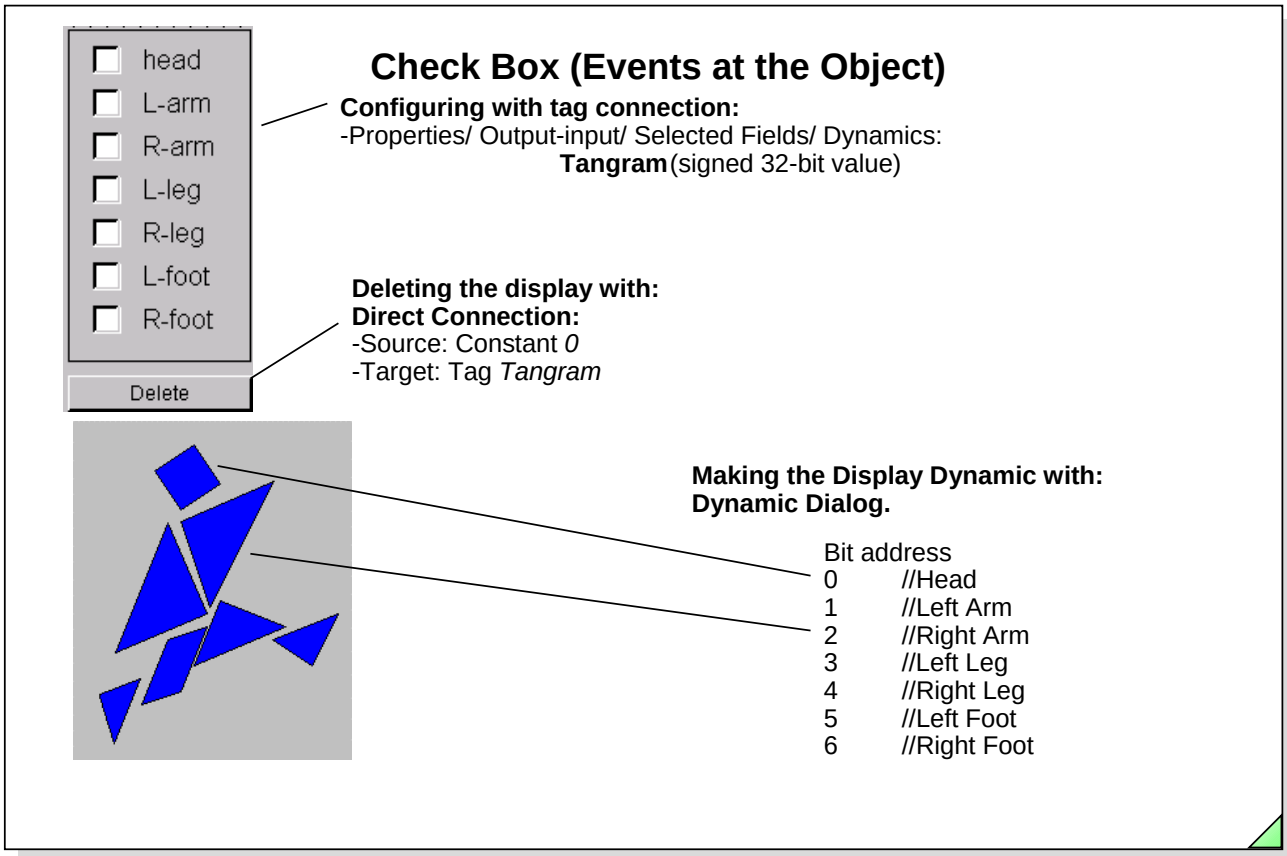
The adjustment value (control range) of the slider object is to lie between 0 and 100.

Procedure

The slider object is generated in the picture using the object palette. You can adjust the size with the resize handle. You yourself label the slider object.

Configuration of Properties

-Properties/ Slider Object/ Miscellaneous/ Maximum Value: **100**
/ Minimum Value: **0**
/ Steps: **10**
/ Process Connection/Dynamic: **value**
/ Password
/ Operator Input message



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.79

SITRAIN Training for Automation and Drives

Task

You can use a check box to make a multiple selection for operations. The switching operation is used for a valve controller, a motor controller, etc., for example.

Example

An object for the tangram is to be displayed for each selected field at the same time as the process value adjustment.

Procedure

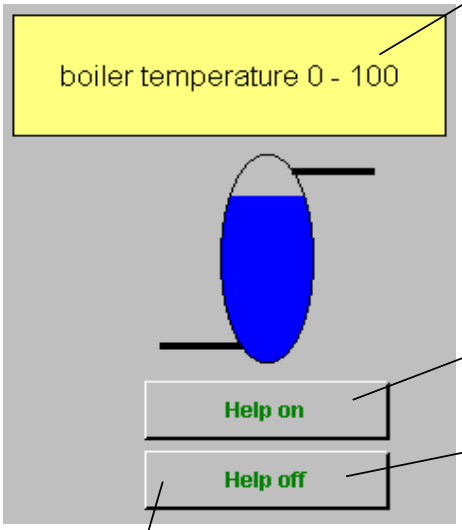
Configure a check box using the Object Palette in the Graphics Designer.
The created check box has three fields that have to be changed in number (max. 32) and labeling, according to the application.
Configuration for the example above:
-Property/ Check Box/ Font/ Index: **1** (1-8)
/Text: **Head** (for Index 1 to 8)

-Property/Geometry/Number of Fields: **8**

Note

The tag used must be a *signed 32-bit value data type*, and the tag occupies a double word. Each individual selection in the check box sets or resets (toggle) a bit in the double word. Several bits are set for a multiple selection.

1. Operator Help per Operation
2. ToolTip Text



Configuring
-Properties/Static Text/ Object Name: **helpText1**
/ Layer: **2**
/ Font/ Text: **boiler temperature 0 to 100**
/ Font/ X Alignment: **centered**
/ Font/ Y Alignment: **centered**
/ Miscellaneous/Display: **no**

Configuration with Direct Connection:
- Source: Constant *1*
- Target: Object in Picture, *helpText1, Display*

Configuring with Direct Connection:
- Source: Constant *0*
- Target: Object in Picture, *helpText1, Display*

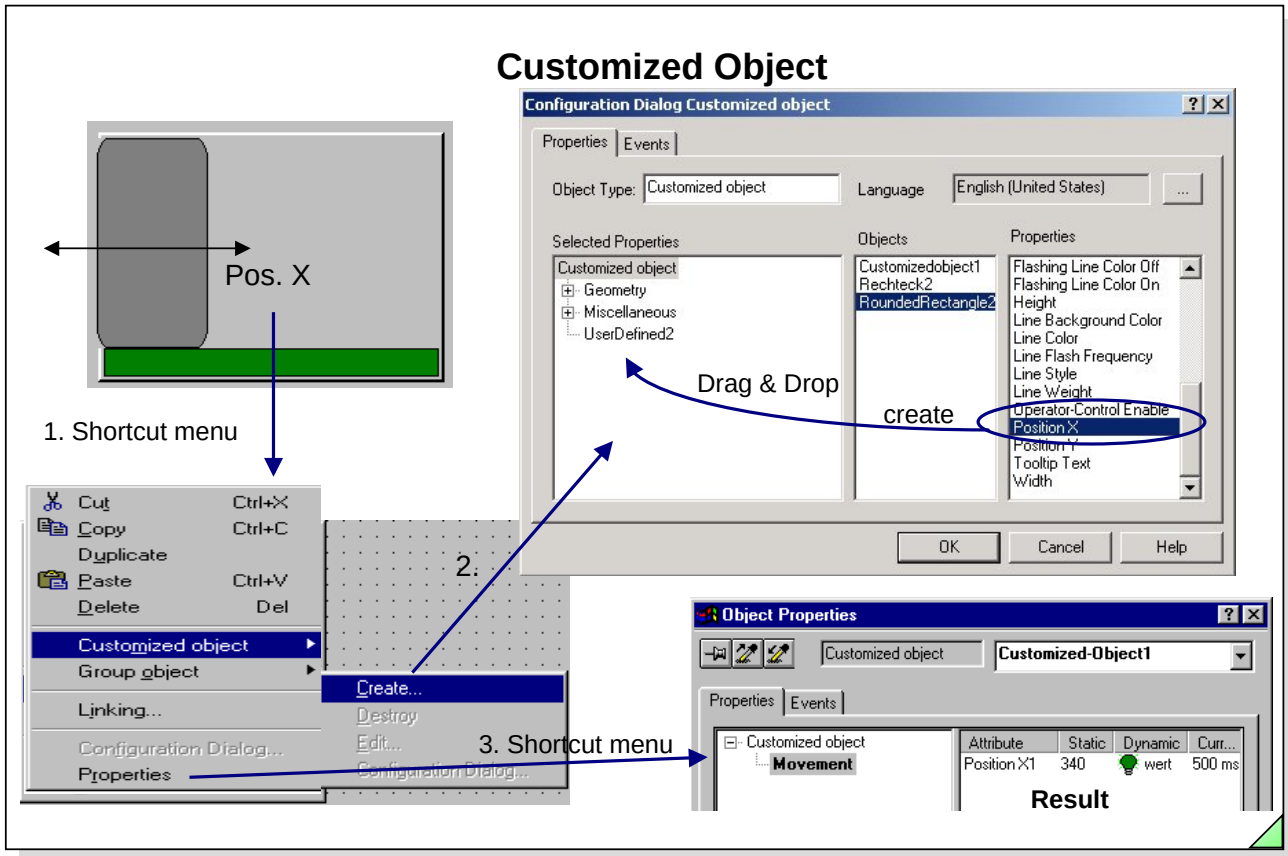
Configuring ToolTip text:
- Properties / Miscs/ ToolTip text: **Boiler** 10

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.80

SITRAIN Training for Automation and Drives

Task	Help texts for the operator are to be displayed and hidden by mouse operation.
Example	The help texts are determined by the technological application and give the operator information about the setting ranges for boiler temperature, for example.
Procedure	Generate a text field and two operator buttons. Label these buttons according to the template in the slide above.
C-Action Help On	-SetVisible(lpszPictureName,"Helptext1",1); -SetFlashForeColor (lpszPictureName, "Button3",FALSE);
C-Action Help Off	-SetVisible(lpszPictureName,"Helptext1",0); -SetFlashForeColor (lpszPictureName, "Button3",FALSE);
Note	You will find additional WinCC functions for changing object attributes under "internal functions/ graphics/ set/ color/, ".
ToolTip	With the ToolTip function you can create a 1 line text with a maximum of 79 characters for display. The text is displayed as soon as the mouse cursor is positioned on the object. - Properties / Miscellaneous / ToolTip text
Exercise 10	See the notes at the end of the chapter.



Date: 27.10.2003
File: SWINCC_04e.81

Customized Object

The customized object allows you to make a hidden configuration of object properties. This makes it possible to display only those properties that are important for the process visualization. Also, the names of the properties can be changed by the designer of the Customized Object, even in multiple languages. The Customized Object can be copied into the library by means of Drag and Drop. Then in the reverse way, the Customized Object can be copied into any WinCC picture.

Note

The Global Library contains a whole series of such Customized Objects (e.g. measuring instruments). These objects can be extended any time with your own objects.

Task

The position of a rectangle is to be dynamicly changed with the help of a Customized Object.
Only this one property is to be shown to the final user.

Procedure

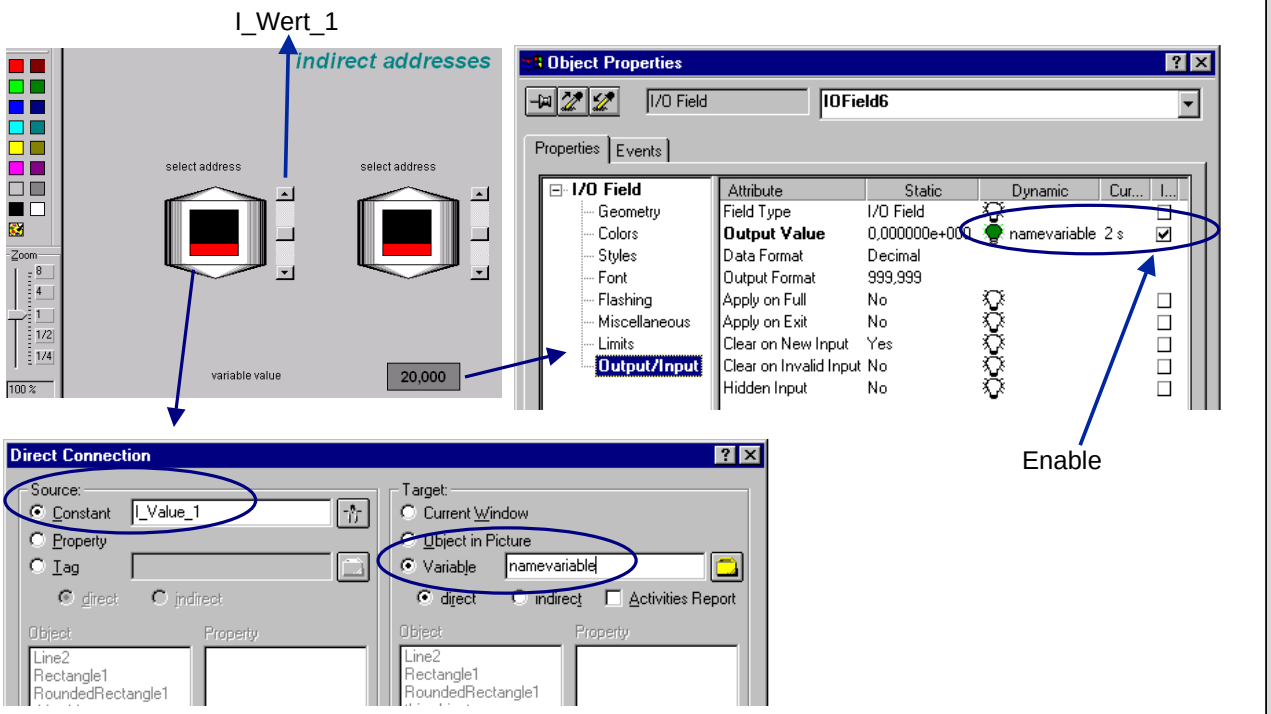
The rounded rectangle (or all objects, belonging to this Customized Object) are positioned in the graphic picture. Next all objects are selected and the

1. Shortcut menu is created with the Customized Object setting
2. Activate "Customized Object" --> "create".
   A Configuration Dialog Customized object will appear where "Selected Properties" (see screen in slide) can be deleted, added and renamed.
3. The process tags can be connected via the properties of the Customized Object.

Exercise 11

See the notes at the end of the chapter.

Indirect Addressing



The tag “namevariable“ must be created as an internal tag of the “Text tag 8-bit character set“ data type.

Date: 27.10.2003
File: SWINCC_04e.82

Objective

With the indirect addressing of attributes, the number of objects and the dynamics (performance profile) associated with it can be reduced.
At the same time, the picture content is easier to read.

Task

The level of several containers is to be represented by an output field.

Procedure

1. Create the internal tag "textvariable" of the "Text tag 8-bit character set" data type.
2. Create the operating and output objects as shown in the graphic slide above.
3. The assignment of the internal tag "textvariable" takes place through the direct connection by mouse click. The source parameter is passed to the target variable.
4. So that the value of the tag "I_Wert_1" is output at the EA_Feld using the attribute "Output Value", the "textvariable" must be enabled through the button in the Indirect column.

Note

If the container name is also to be output using an object "Statischer Text", you have to use the following script by mouse click.

```
char  szText[20]="Behälter 1";
SetTagChar("textvariable","I_Wert_1");
SetPropChar(lpszPictureName,"StatischerText1","Text",szText);
```

Exercise 12

See the notes at the end of the chapter.

Block Technology Overview

-Tag Prefix
(>=V5.0)

-Indirect Addressing

(regardless of the version)

-Picture in Picture Technology with Dynamic Wizard

(Picture blocks: 1. TypeWizard, 2. InstanceWizard, >=V4.02)

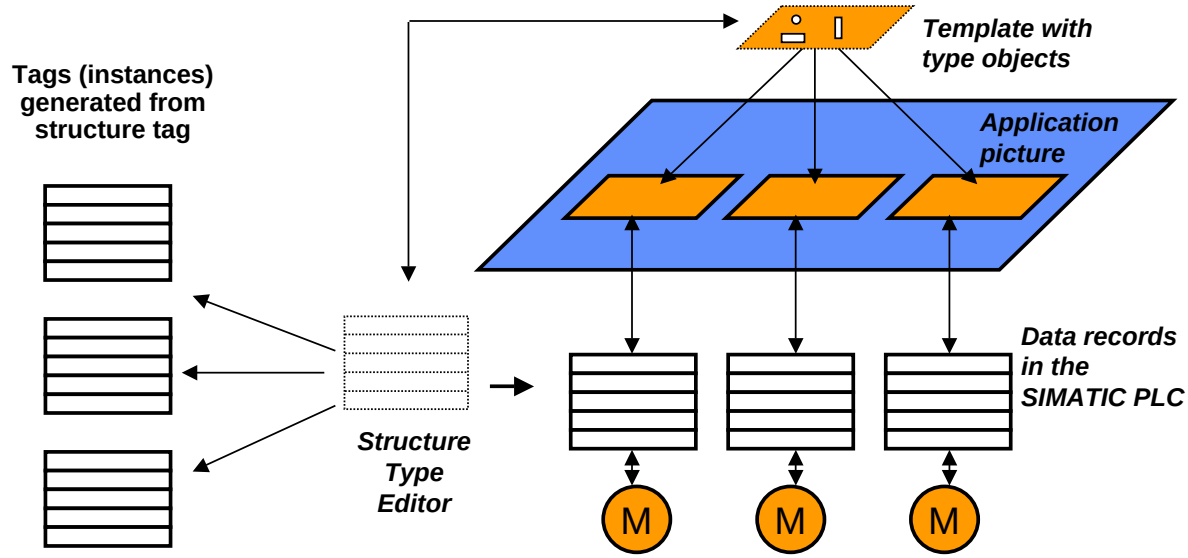
-ActiveX Controls (SIMATIC HMI Option with IndustrialX)

(additional with VB 5 configuration tool)

Date: 27.10.2003
File: SWINCC_04e.83

Block Technology	Picture block technology allows the engineer to create functions (such as motor blocks, drive blocks, open-loop controller blocks etc.) that are used many times, whereby every call of the function works with its own data. The overview shows you four configuration possibilities. The first three can be created with the WinCC tools. ActiveX- Controls can be made with the option package Industrial X and Visual Basic Version 5.
Tag Prefix	Simple creation of picture modules. The template can be changed from a central location. All possibilities for making dynamic can be used: + Configuration Dialog + Dynamic Wizard + Tag Connection + Dynamic Dialog + Direct Connection + Property Actions according to ANSI C Syntax + Event Actions according to ANSI C Syntax
Ind. Addressing	Takes into consideration all steps for creating picture modules. Changes are made centrally. There is only one template that cannot be ‘instantiated‘.
Picture in Picture Technology	Easy to configure picture modules. Changes are made centrally.
ActiveX Controls	To use this you need to have knowledge of an additional editor. User-application blocks with high performance (Templates) and with know-how protection.

Block Technology in General



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.84

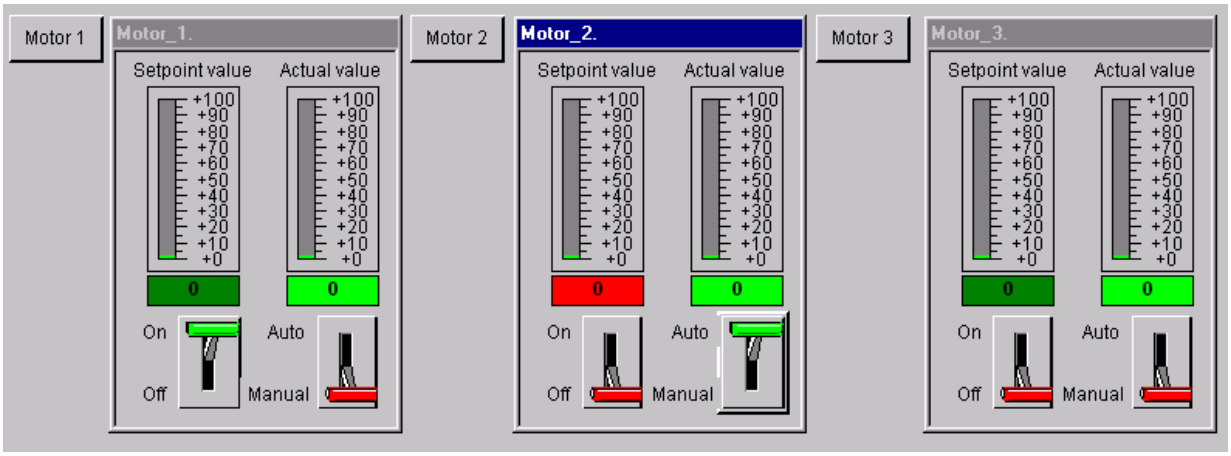
SITRAIN Training for Automation and Drives

General

To reduce the costs of configuration and management of simular process objects, WinCC offers you the opportunity of using picture block technology.

Based on similarly structured data records in the PLC (interface description for function blocks), the **Structure Tag** (template) is created (for example motor blocks, drive blocks, open-loop controller blocks etc.) once with the **Structure Type** Editor in the WinCC **Explorer**.
This structure tag represents the interface used to read and write data to the PLC for one specific object type.
Then, the type objects are created in the template and these are connected to the Struct-Membername in the structure tag. For example, when a type object is called, this object is connected to the relevant addresses from the PLC.

Task: Project in Runtime, One Picture Window for Each Button Call Tag Prefix



SIMATIC HMI

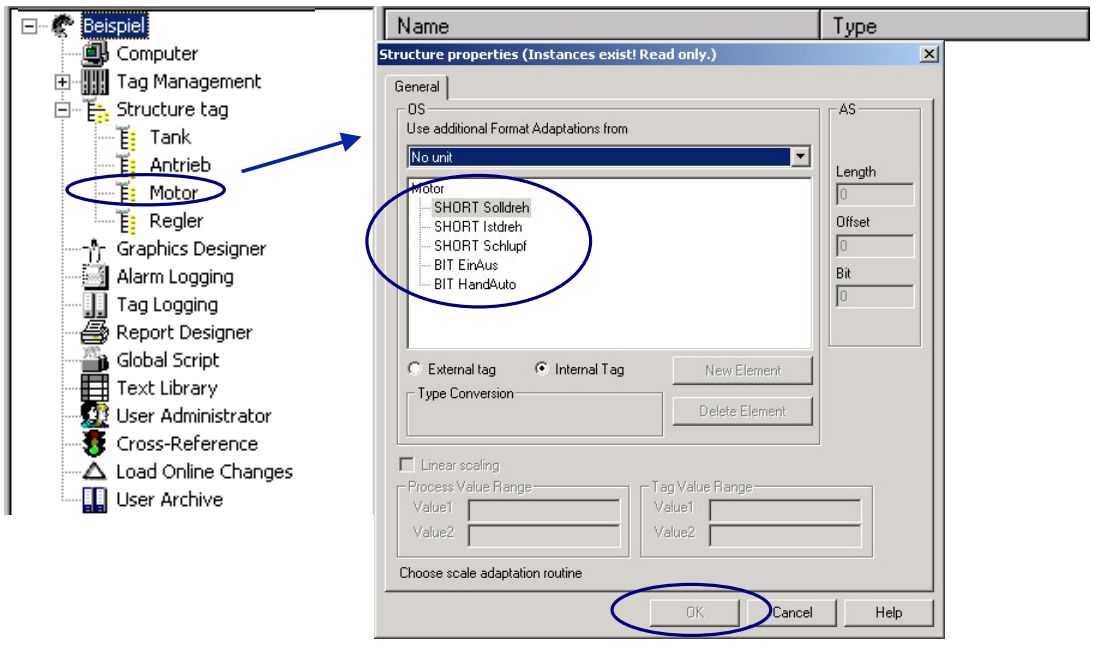
Date: 27.10.2003
File: SWINCC_04e.85

SITRAIN Training for Automation and Drives

Task

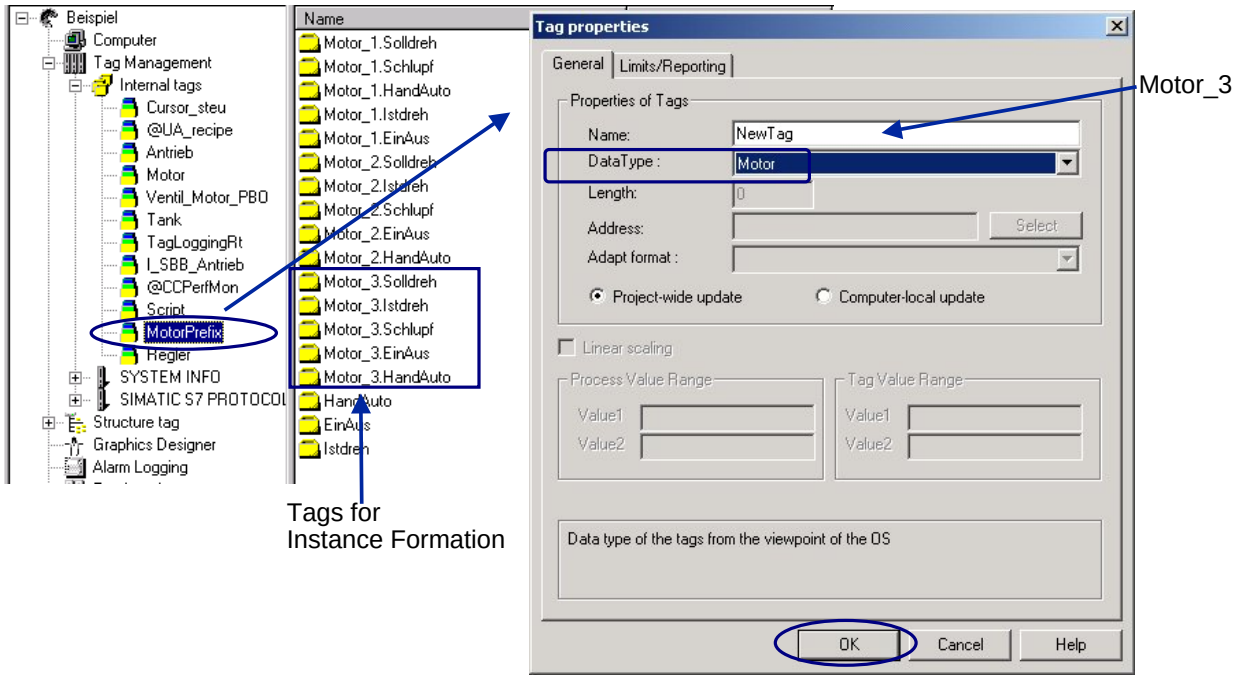
The picture block for a motor shows the typical elements for operating and displaying. You can make a switch with the left switch. With the right switch, can set the manual mode for adjusting the setpoint.

Create Structure Tag in the Project



Date: 27.10.2003
File: SWINCC_04e.86

Create Internal Tags and/or Process Tags fitting to the Structure



Tags for Instances

In the slide you can see the tags for three motors.

Internal Tags

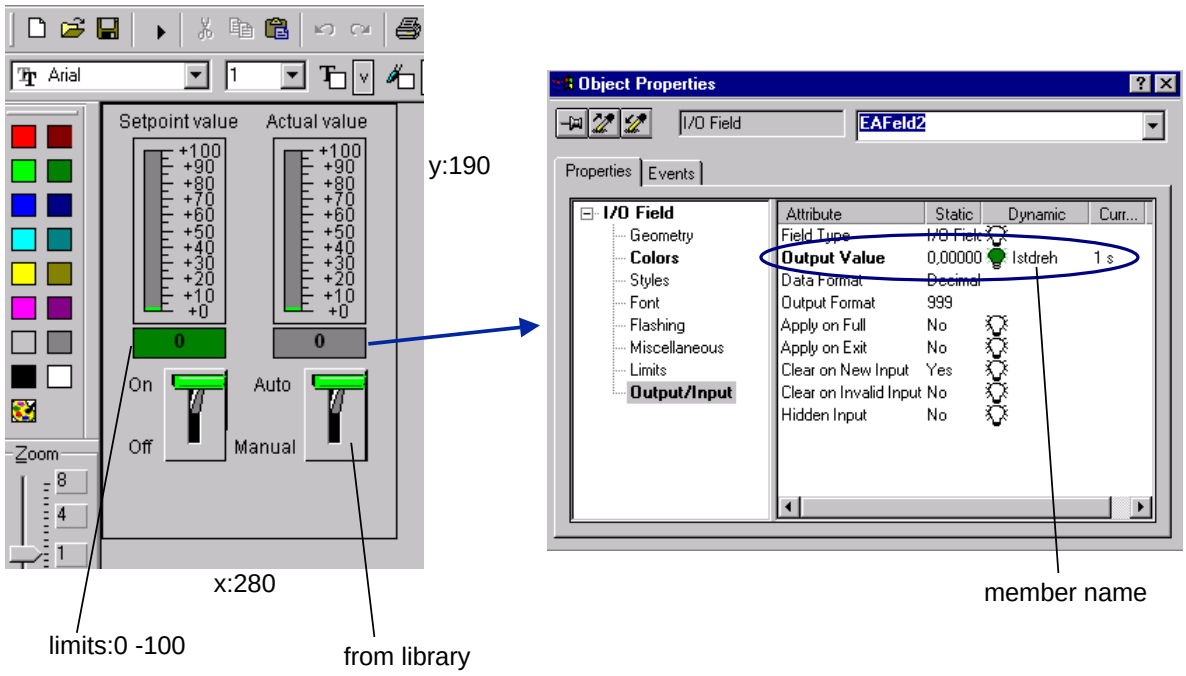
If you use the Dynamic Dialog, you will require internal tags as well, since the Dynamic Dialog checks for the existence of the tag in the Tag Management. If these internal tags don‘t exist, the Dynamic Dialog issues an error message when you assign parameters with the member name.
See the example on Page 84.

Character Length

Length of tag names: [Instancename] .[Membername]
is a maximum of 128 characters.

-a maximum of 31 instances of the same template can be used in one system picture.

Configure Template, Output Value 1/4



SIMATIC HMI

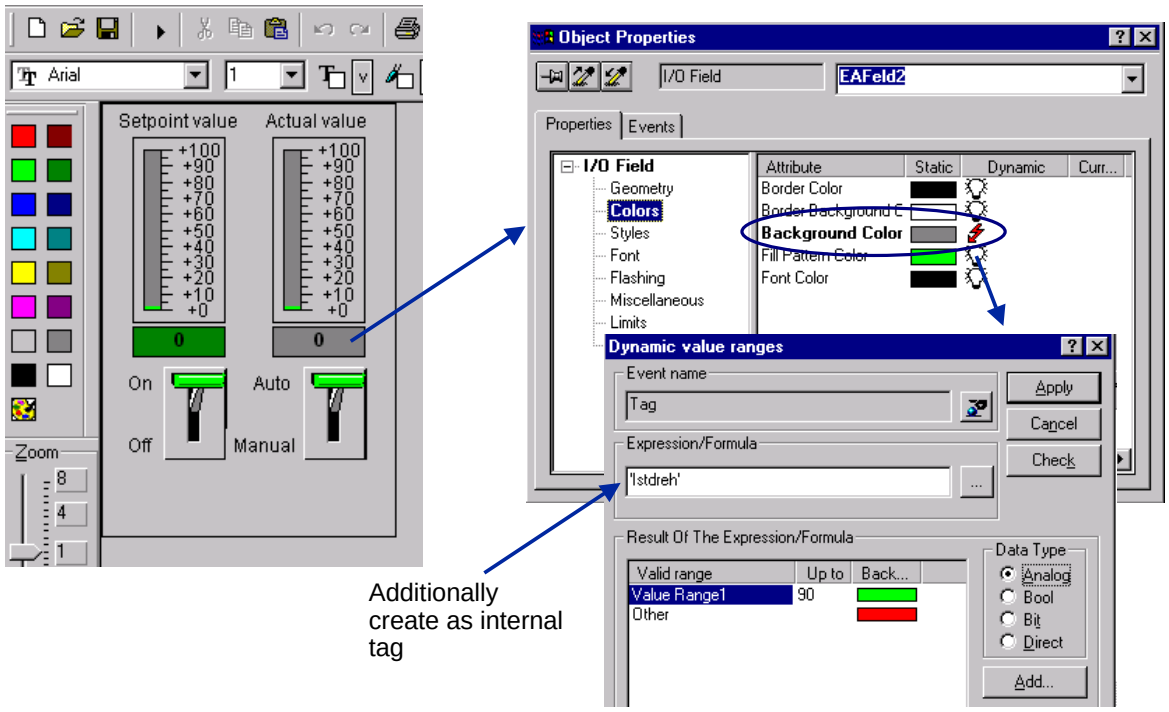
Date: 27.10.2003
File: SWINCC_04e.88

SITRAIN Training for Automation and Drives

Objective

With the template shown in the slide, you are to be able to change the setpoint speed of a motor while in manual mode. At the same time, input limits (0-100) are to be monitored. In automatic mode, you recognize from the color attribute of the I/O field that the input is disabled. The setpoint and actual (measured) speed is also displayed graphically. An additional switch button is used to switch the motor on or off.

Configure Template, Background Color for Limit Values 2/4



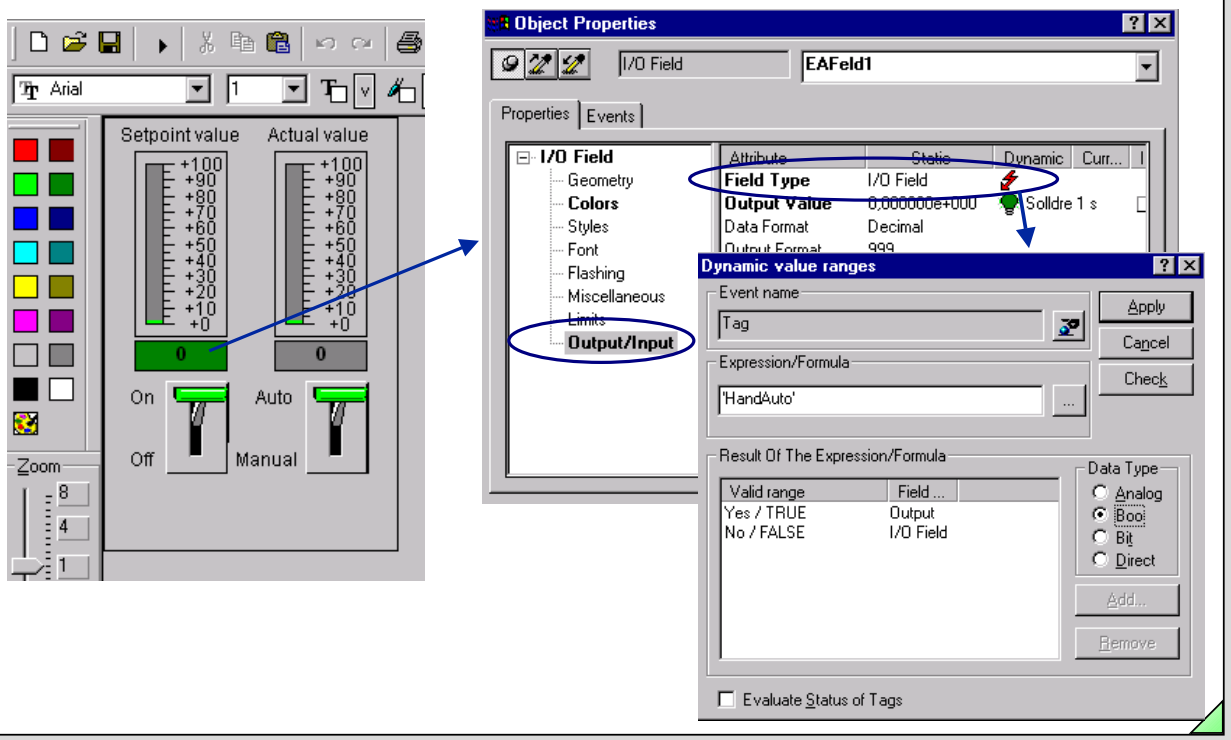
Dynamic Dialog

Internal tags are needed so that the Dynamic Dialog can be closed without an error message appearing (a reference to open tag references appears). The internal tags are converted into structure tags during runtime. Tag prefix of the picture window and tag name of the internal tag--> [Instancename] .[Membername].

Note

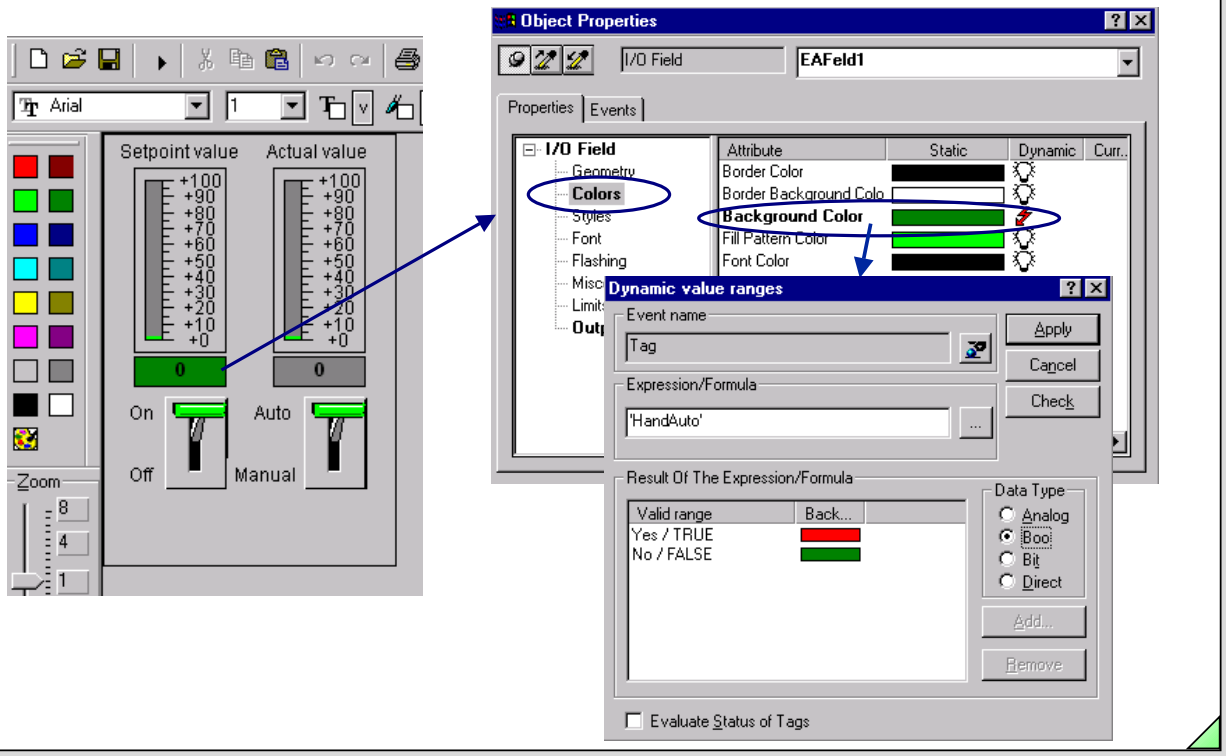
Editing of the internal tag can take place from the Dynamic Dialog call (if it has not taken place in the Tag Management).

Configure Template, Field Type 3/4



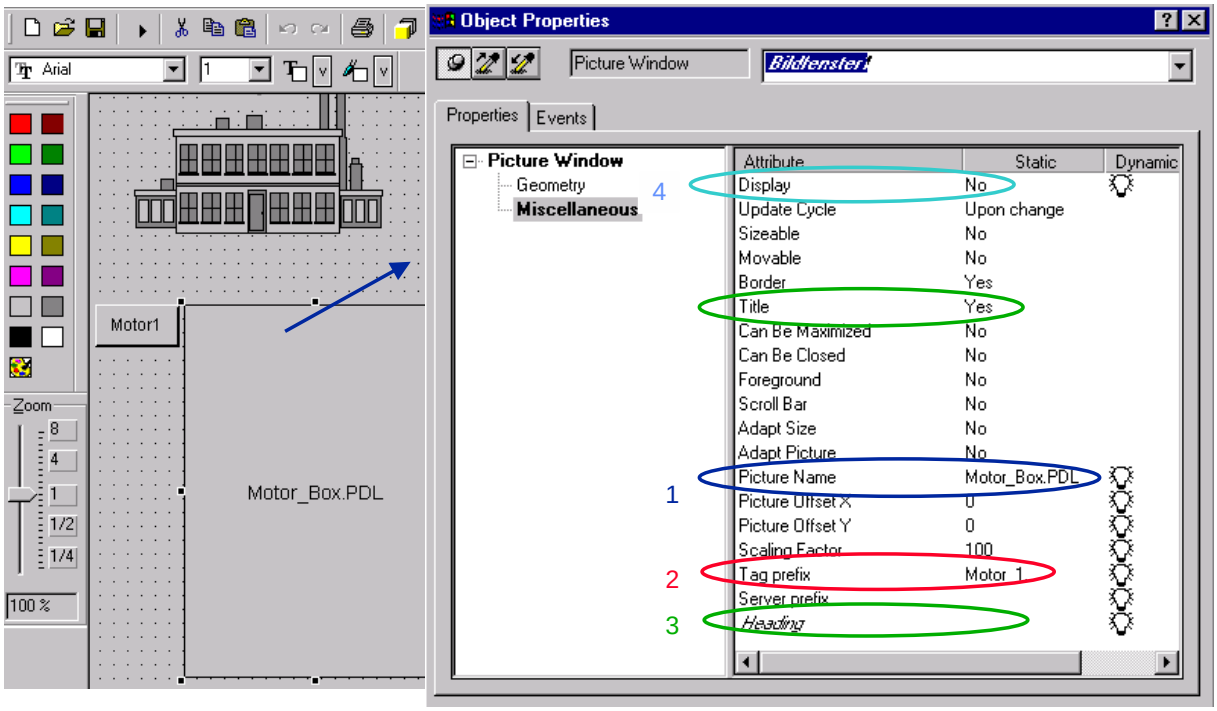
Date: 27.10.2003
File: SWINCC_04e.90

Configure Template, Background Color for Switch Condition 4/4



Date: 27.10.2003
File: SWINCC_04e.91

Configure Motor Picture, Attributes for Picture Window, 1/4



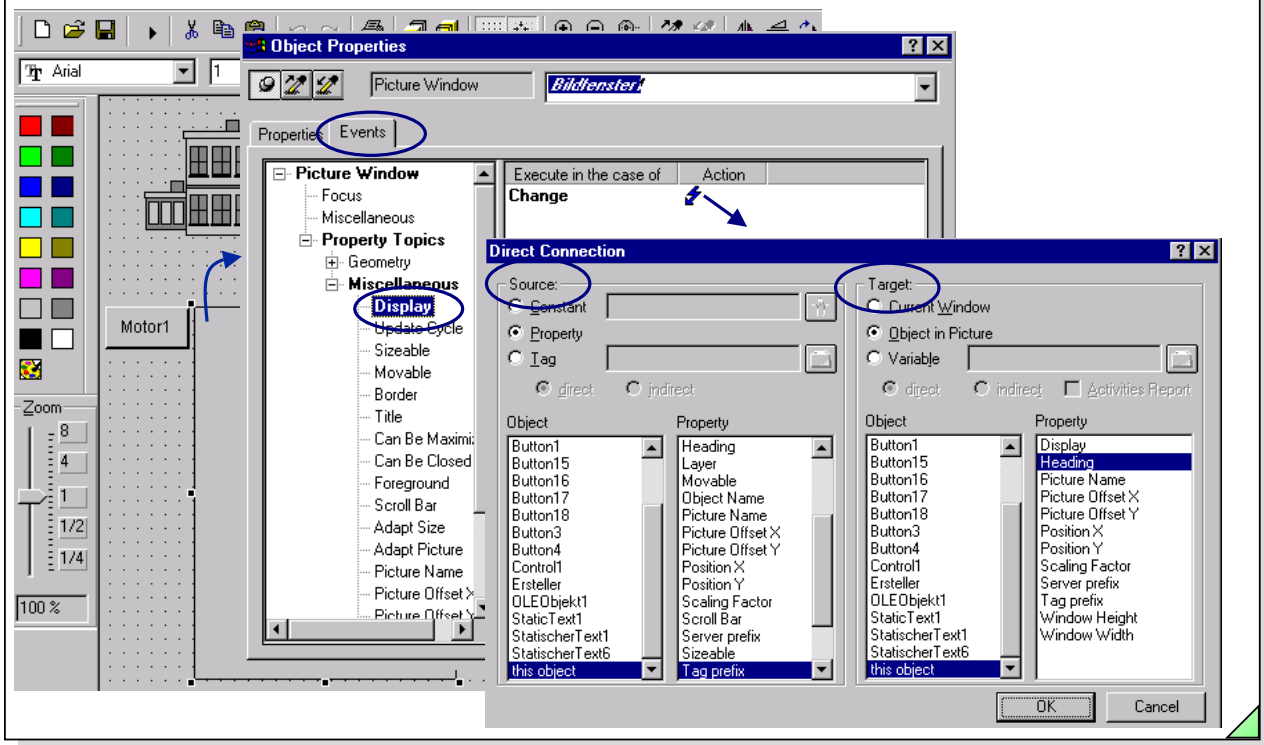
Picture Name (1) The name of the picture that is to be displayed in the picture window is entered in this attribute.

Tag Prefix (2) The tag prefix of the structure that is to be displayed in the picture window is entered in this attribute. A maximum of 31 picture windows with the same structure can be connected in one picture.

Heading (3) The text that is to appear in the title of the picture window is entered in this attribute. (In this example, the text is entered through a direct connection, see next page).

Display (4) The Static is set to "no" in this attribute so that the window does not immediately appear when the system picture is called.

Configure Motor Picture, Title for Picture Window, 2/4



SIMATIC HMI

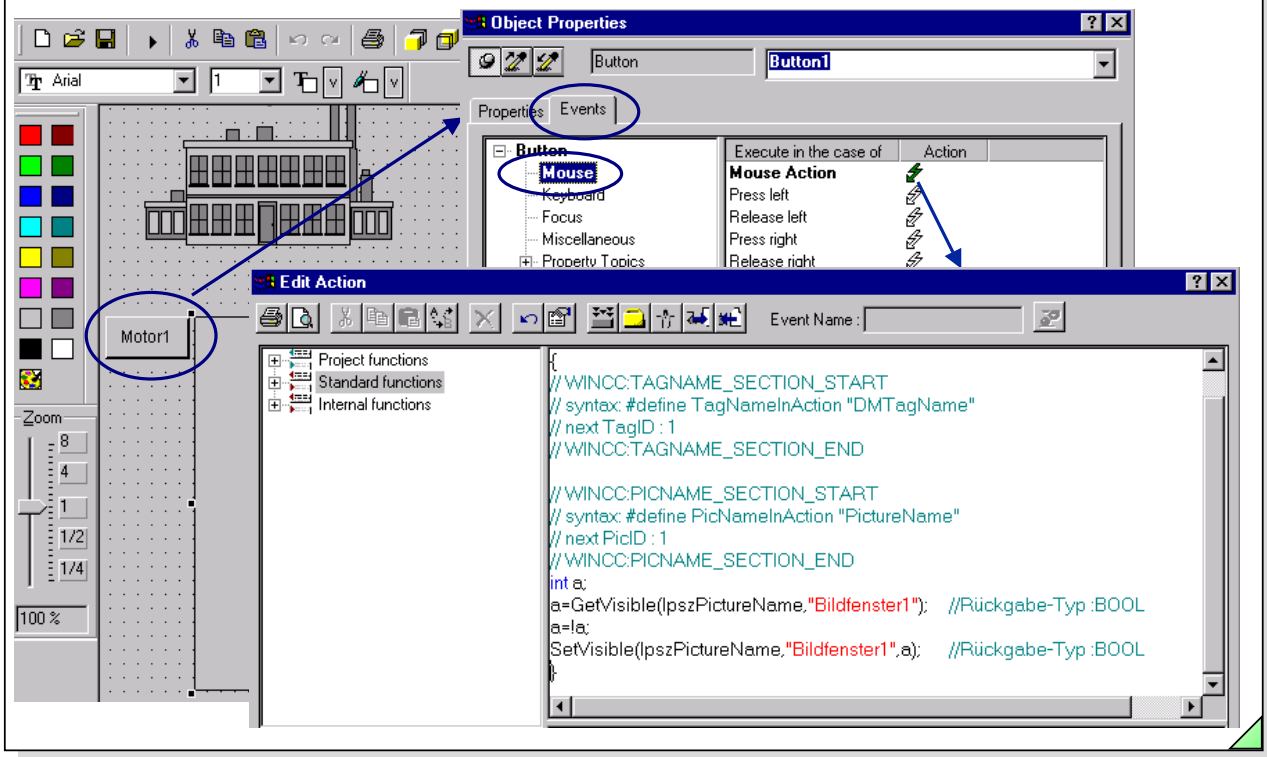
Date: 27.10.2003
File: SWINCC_04e.93

SITRAIN Training for Automation and Drives

Action When the picture window is opened, the tag prefix is read out and copied to the Heading attribute. Since the object "this object" was used in the direct connection, the picture window can be copied as often as you like without having to change the direct connection.

SetPropChar You can create a specific picture window title (heading) with **SetPropChar**(lpszPictureName,“Bildfenster6“,“CaptionText“, “Hydraulikpumpe1“);

Configure Motor Picture, Call Picture Window, 3/4



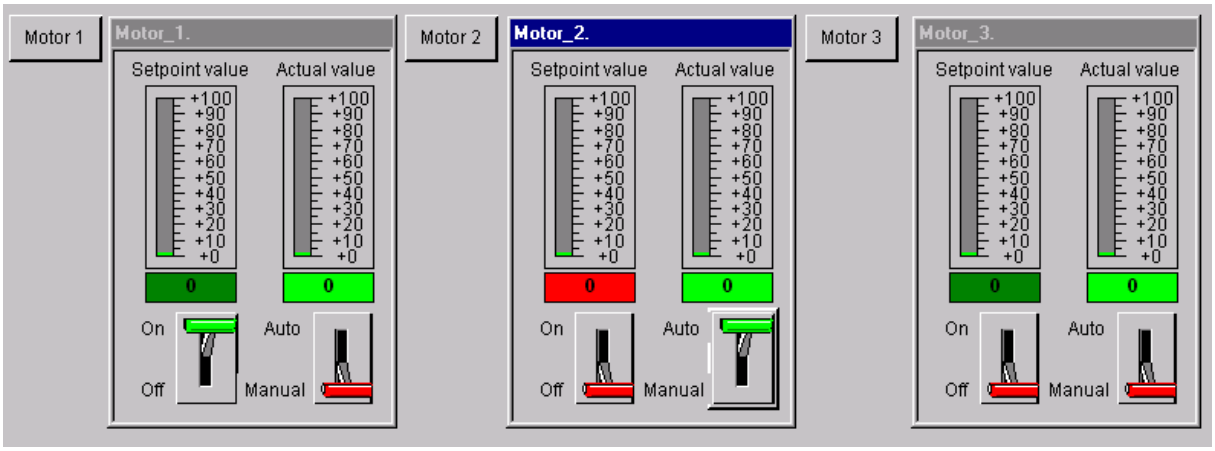
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.94

SITRAIN Training for Automation and Drives

Action The C-Script describes the toggle function Show/Hide for the picture window.

Result: Project in Runtime 4/4



SIMATIC HMI

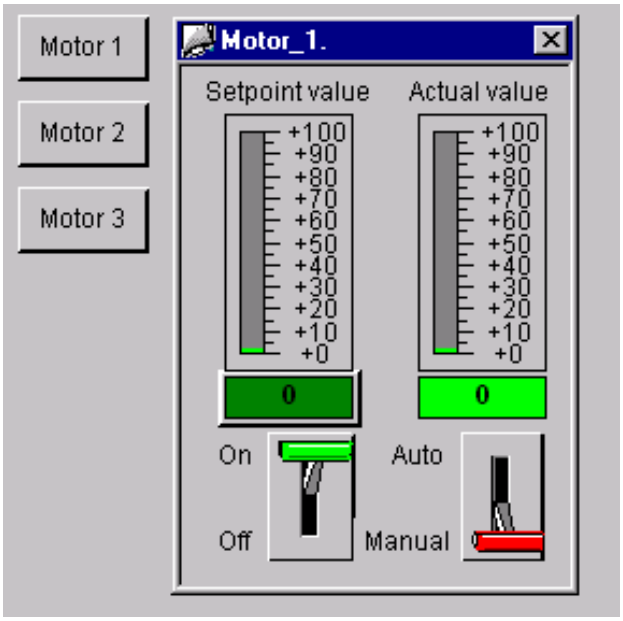
Date: 27.10.2003
File: SWINCC_04e.95

SITRAIN Training for Automation and Drives

Result Each button (Motor 1 to Motor 3) can switch the associated picture window on and off.

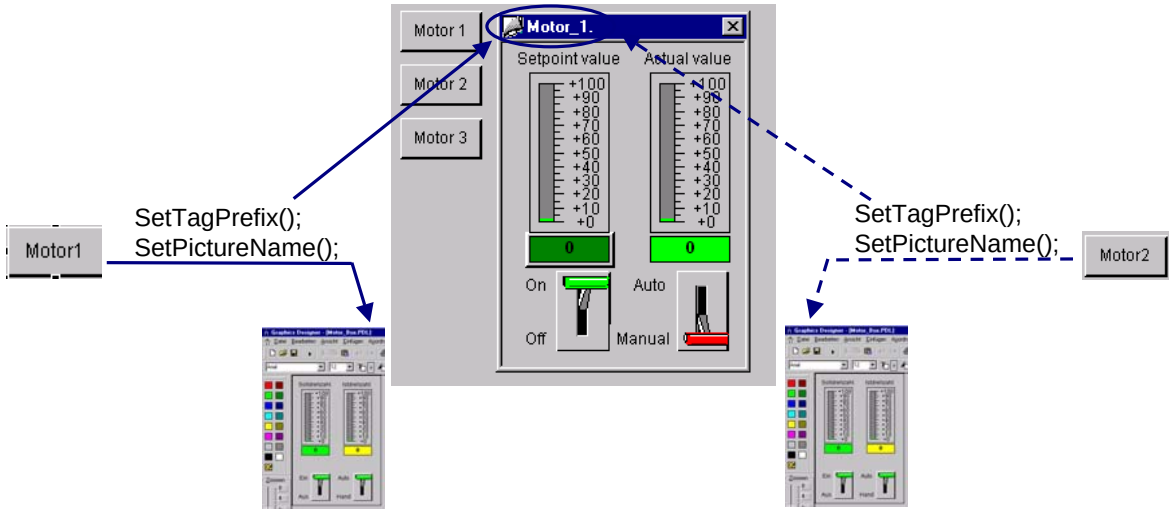
Exercise 13 Configure the steps from **Pages 83-93** for the picture block technology.

Task: Project in Runtime, One Picture Window for Several Motors



Date: 27.10.2003
File: SWINCC_04e.96

Sequence in Instance Formation



Note: SetTagPrefix(); The “Tag Prefix“ attribute can be made dynamic with the name “TagPrefix“.
SetPictureName(); The “Picture Name“ attribute can be made dynamic with the name “PictureName“.

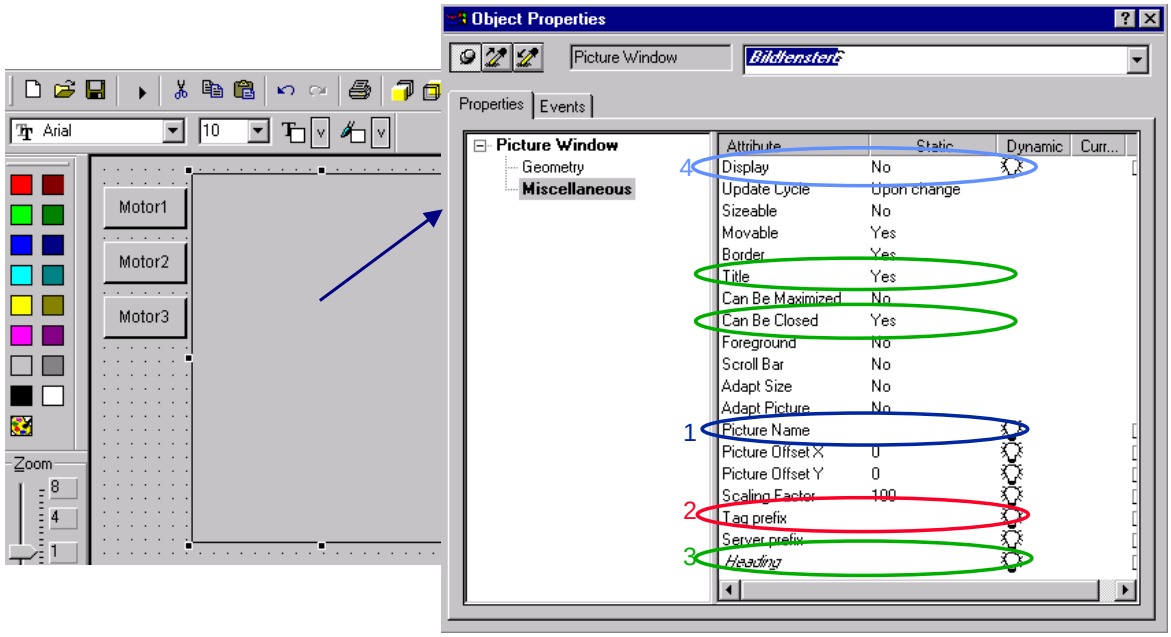
SetTagPrefix

SetTagPrefix(lpszPictureName,"Bildfenster1", "Motor_1.");
This function sets the tag prefix of a picture window.
In one picture window, the tag "Measured" is called in an object. If a tag prefix "Motor_1." is assigned to the picture window, then the tag "Motor_1.Measured" is called. The setting of the tag prefix only becomes effective when the picture name is reassigned.

SetPictureName

SetPictureName(lpszPictureName,"Bildfenster1",“Motor_Box.pdl");
You have to set the prefix before you select the picture and reassign the picture name, if the picture is not to be changed.

Configure Motor Picture, Attributes for Picture Window



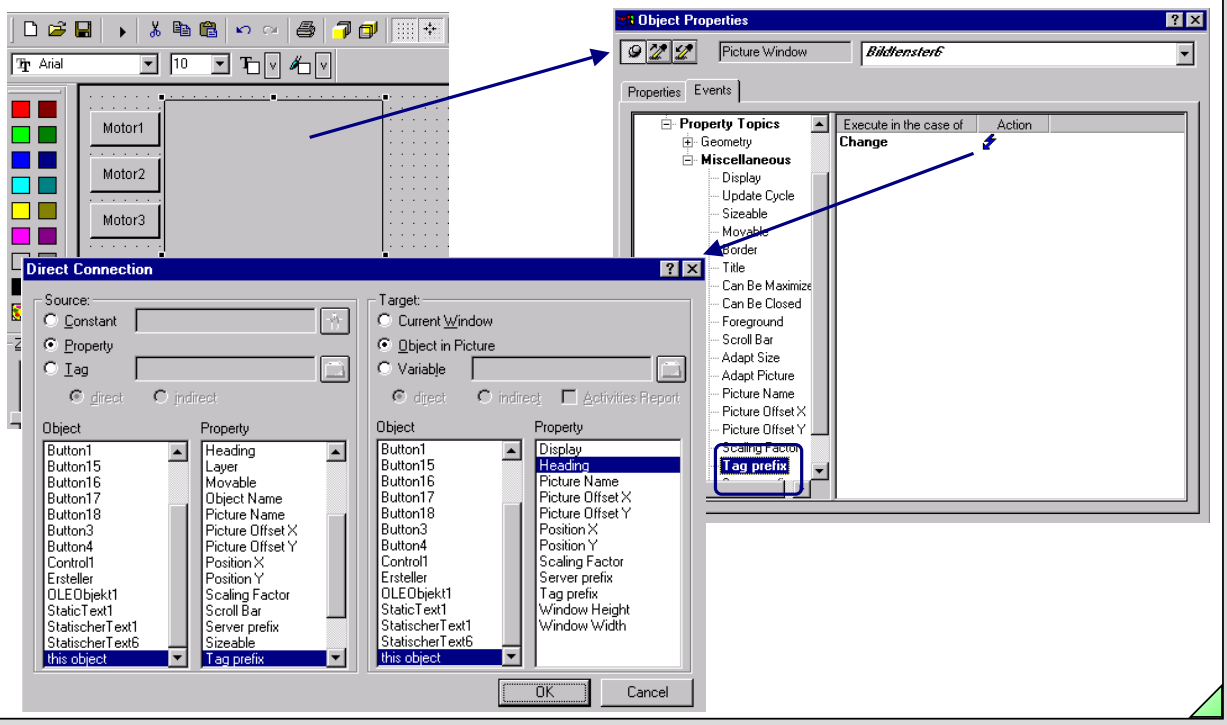
Picture Name (1) No name of the picture is enetered in this attribute. The dynamics occurs through SetPictureName();.

Tag Prefix (2) No tag prefix is entered in this attribute. The tag prefix is assigned using the function: SetTagPrefix(); (or) with a mouse click on the button Motor1.

Heading (3) The assignment of this attribute is made using the Direct Connection. The Direct Connection can be seen on the following pages.

Display (4) The Static is set to "no" in this attribute so that the window does not immediately appear when the system picture is called.

Configure Motor Picture, Title for Picture Window

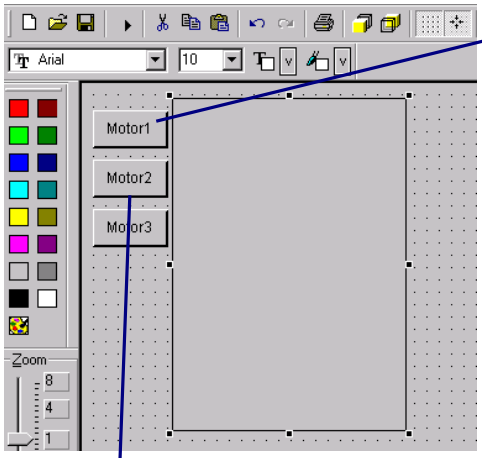


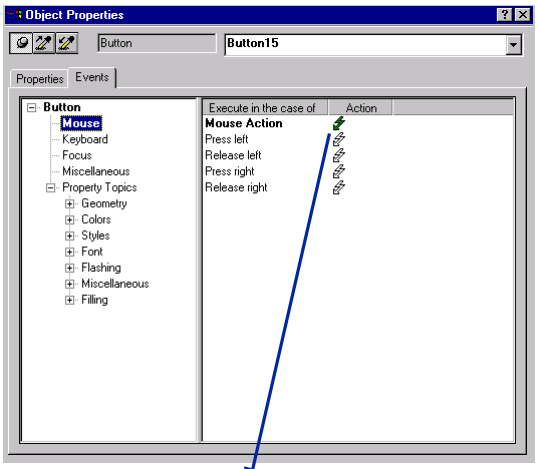
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.99

SITRAIN Training for Automation and Drives

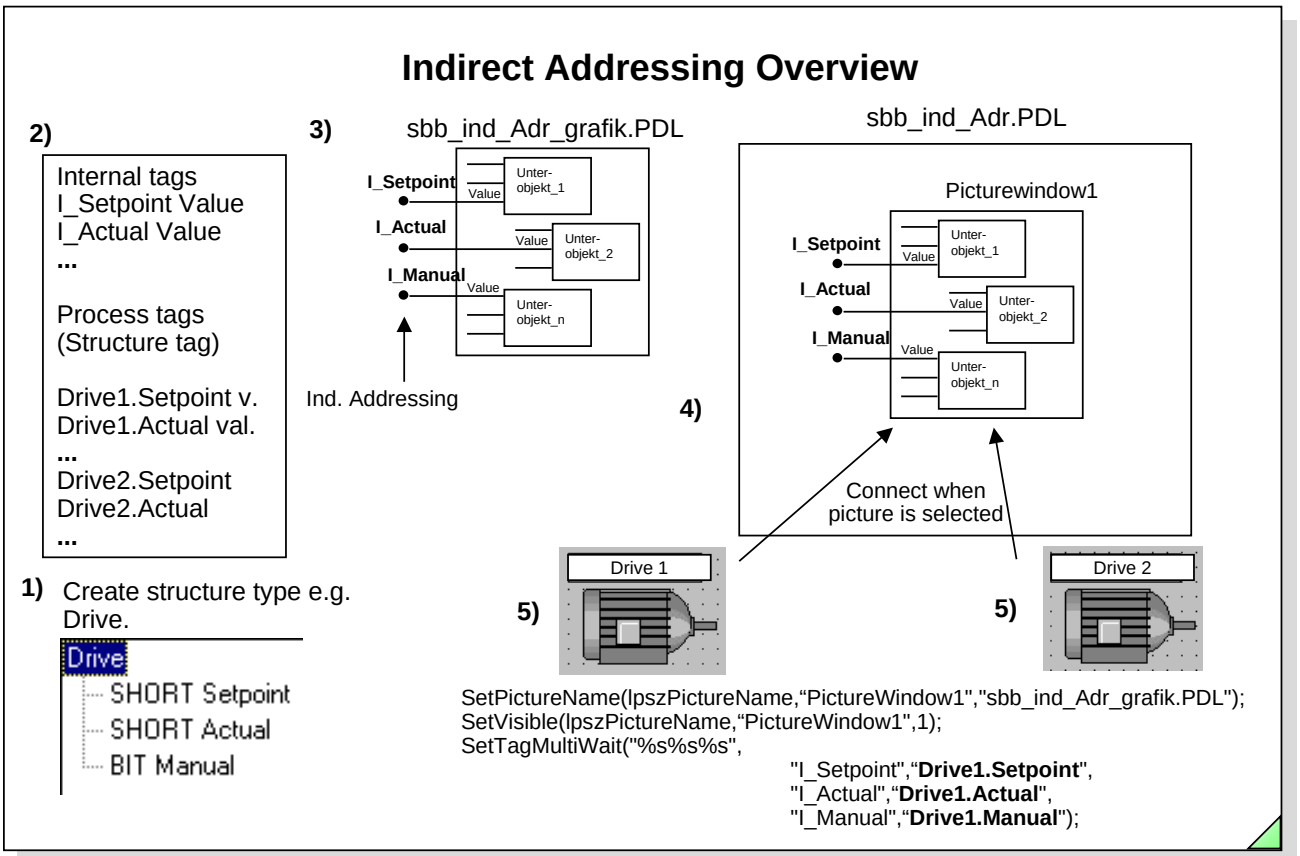
Configure Motor Picture, Button for Connecting the Tag Prefix





SetTagPrefix(lpszPictureName,"Bildfenster6",“Motor_1.");
SetPictureName(lpszPictureName,"Bildfenster6",“Motor_Box.pdl");

SetTagPrefix(lpszPictureName,"Bildfenster6",“Motor_2.");
SetPictureName(lpszPictureName,"Bildfenster6",“Motor_Box.pdl");



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.101

SITRAIN Training for Automation and Drives

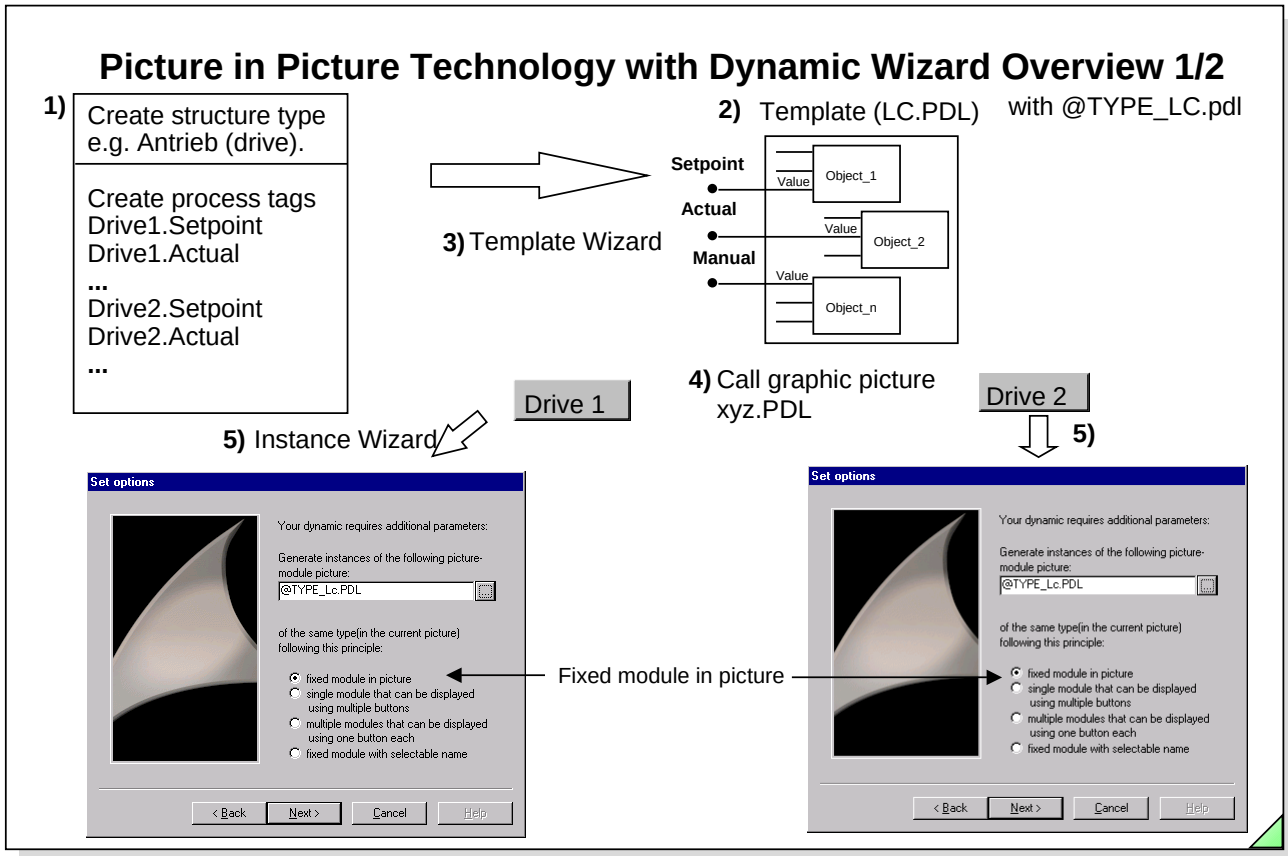
Configuration

1. With the "Structure Type" editor, in the WinCC Explorer, create the necessary data structures for each type.
2. With the "Tag Management" editor, in the WinCC Explorer, create the necessary internal and process tags, simular to the data structure of step 1.
3. With the Graphics Designer configure a template (picture module), that is connected to the internal tags from step 2.

   Important: It is very important that for every tag connection ‘indirect addressing‘ is selected.
4. Now create a process picture with a picture window and the --> Property -->Display:no.
5. The call of the picture module with a connection to the SIMATIC addresses now takes place using an object in the picture (e.g. motorsymbol). For this you have two possibilities:
   - SetTagChar("I_Sollwert","Antrieb1.Sollwert");
   - SetTagChar("I_Istwert","Antrieb1.Istwert");
   - SetTagChar("I_Hand","Antrieb1.Hand");

   Note: The structure tag name has to be identical to the name in the Tag Management (upper/lower case).
   or
   -the SetTagMultiWait Function, see slide (as of WinCC Version V4).
   Call: Internal Functions--> tag--> Set--> wait--> SetTagMultiWait.

For many tags, the use of the SetTagMultiWait function is ideal since it writes all tags once. With the SetTag function each tag is written individually. This becomes obvious when pictures are updated.



Configuration

1. With the "Structure Type" editor, in the WinCC Explorer, create the necessary data structures for each type.

   With the "Tag Management" editor, in the WinCC Explorer, create the necessary process tags, simular to the data structure of step 1.

2.+3.
   With the Template Wizard, started in the Graphics Designer, located in the Dynamic Wizard behind the Picture Modules tab, the connection between the object properties from the objects in the picture and the structure components of a structured data type are made.
   The wizard generates an I/O-field with the object name "InstanceName" in the picture LC.PDL and saves the picture under the name @TYPE_LC.PDL. This is the new template.

4.+5.
   Now open the graphic picture xyz.PDL in the Graphics Designer and start the Instance Wizard "Picture-Module Instances" (in the Dynamic Wizard behind the Picture Modules tab). Select the template @TYPE_LC.PDL and select the instance type (1) with the first selection in the option group (see Window "Set Option" in the slide above).
   One by one select the structure tag that will be connected to a picture window (each picture window will get a position). After the wizard is finished, all picture windows will be generated automatically (two in the picture xyz.PDL, in this example).

   Activate the picture xyz.PDL and test the result in Runtime

Picture in Picture Technology with Dynamic Wizard Overview 2/2

Performance Profile

Instance Type 1: **fixed (picture) module in picture**

Instance Type 2: **one module that can be displayed and that can be called using one button (per tag)**

Instance Type 3: **several modules that can be displayed and that each can be called using one button (tag)**

Instance Type 4: **fixed module with selectable name**

Date: 27.10.2003
File: SWINCC_04e.103

Length of tag names: [Instancename].[Membername] max. 128 characters.
Number of tags: max. 100 variables for each instance (maximum number of structure components).

Number of Instances in a main picture

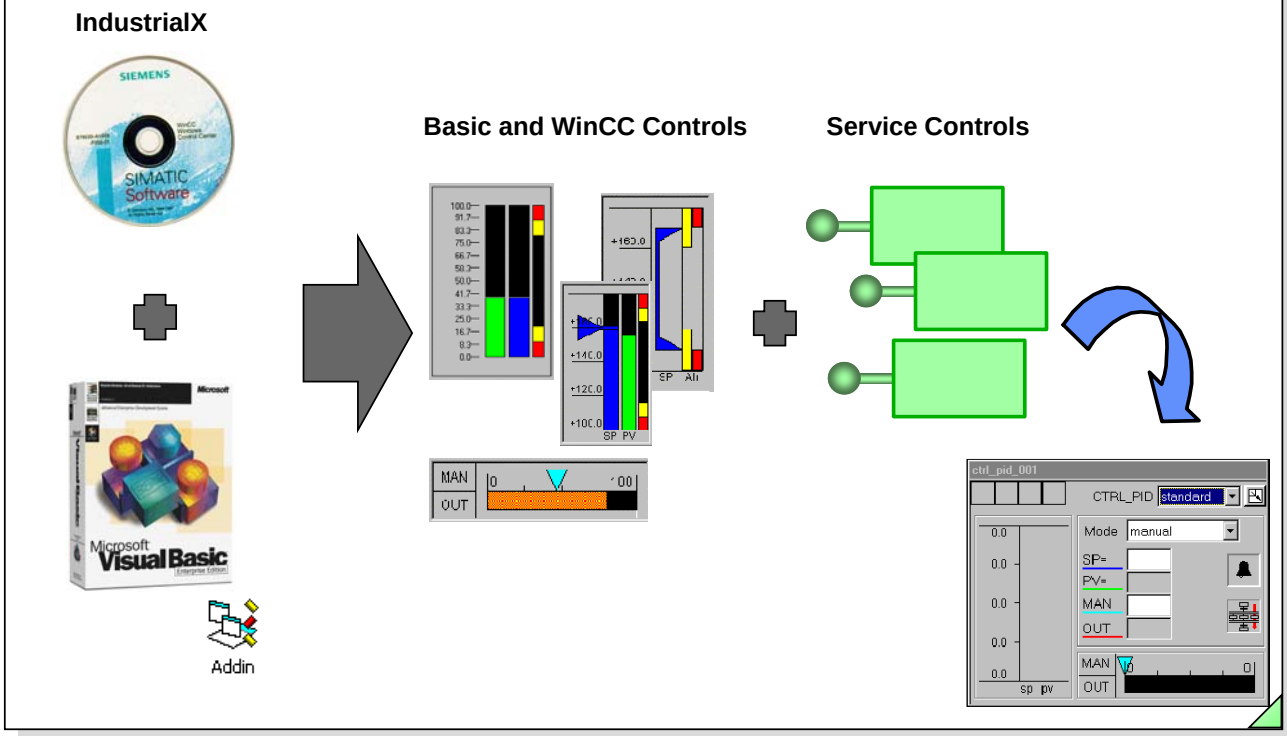
· Instance type (1): maxímum of 19 instances of the same template.

· Instance type (2): any number of buttons in a main picture (only one picture-instance window can be opened at a time).

· Instance type (3): up to 100 buttons in one picture. Of these, you can have up to a maximum of 19 picture window instances of the same template open at one time.

· Instance type (4): maximum of 19 instances of the same template.

Block Technology with ActiveX- Controls (Option with IndustrialX)



Development Toolset to develop picture modules.

The IndustrialX-CD consists of:

- Development Toolset with Basic and Service Controls

for example, **Basic Controls** with operator interface
Bar (Graphs)
I/O fields
List Box
etc. (depending of the version)

for example, **Service Controls** without operator interface
for process connections
for handling messages
for pasword handling
etc. (depending of the version)

-WinCC Controls and Controls of Others (Demo)

Display Picture Name in any Picture

-With the key combination Shift+Ctrl+Alt

(In Runtime, point with the mouse on an object in the picture. After a short time a ToolTip text is displayed).

-Display status line at the bottom

(In the WinCC Explorer, using Computer properties, select the window attribute "Status-Line ". After Deactivate - Activate Runtime the picture name of every picture is displayed in the Status-Line at the bottom).

-Updating a static text per C-Script

(Solution is dislayed in the text below).

Task

The current file name is to be displayed in the picture .

Procedure

To display the current file name in the picture, it is sufficient to configure a static text for which a small C action makes the text output dynamic. The lpszPictureName call parameter of the C action is simply returned in the return value.
Configuring properties:
Static Text / Font / Text / C-Action ...

return lpszPictureName;

Note

The following must be taken into consideration during the configuration:

1. One hour is to be set as the action trigger
2. Assign the text attribute once
3. Properties / miscellaneous / Adapt Border: **Yes**

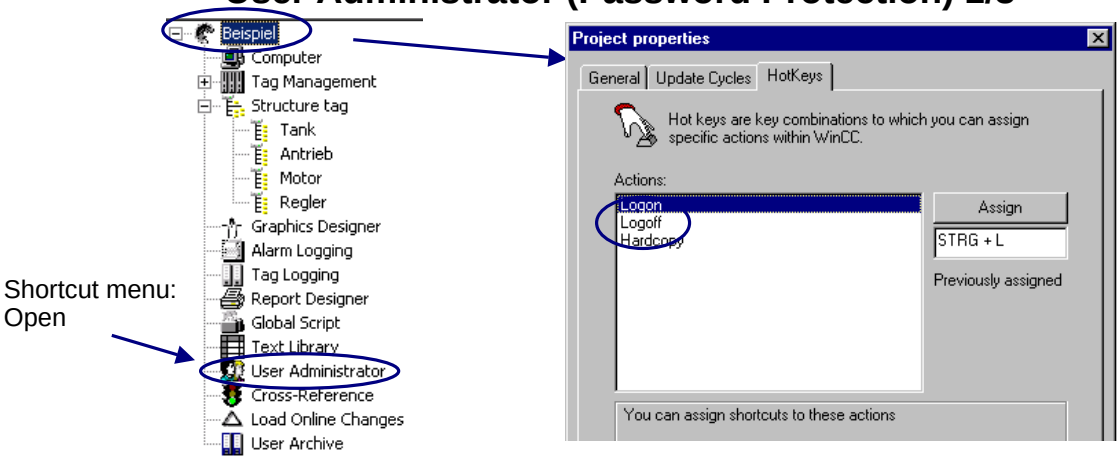
This static text can be copied into each picture and displays the current picture name during Runtime.

When needed, the visibility can be controlled by a central, internal tag, so that the picture name output can be suppressed.

Recommendation

Copy the object after it is created in the project library.

User Administrator (Password Protection) 1/3



Group		Electrician		Locksmith		
User	Login Password	**Otto** *Otto_WB*		**Willi** *Willi_B*		
	No	Function	Enable	No	Function	Enable
	1	Value input	⊙	1	Value input	-
	2	Picture change	⊙	2	Picture change	⊙

General Information

The assignment and management of access rights can be configured with the User Administrator so that there is no unauthorized access. That is, each process operation, archive operation, and WinCC system operation can be locked against unauthorized access.
Examples of operations that can be protected against unauthorized access include: changes in setpoints, recipes, picture selection, or the call of configuration software from the process operation.

There are different access levels that enable the configuration of a hierarchical access protection, such as exclusive operating authorization for individual operators.

Example

A setpoint adjustment and the call of a process picture for entering recipes is to be protected against unauthorized operations. The picture selection and the setpoint adjustment is to be permitted for the process engineer "Otto", using the "Otto" login and the "Otto_MS" password. The operator "Willi" can use the "Willi" login and the "Willi_B" password to view the recipes, but he cannot carry out any setpoint adjustments.

Configuring

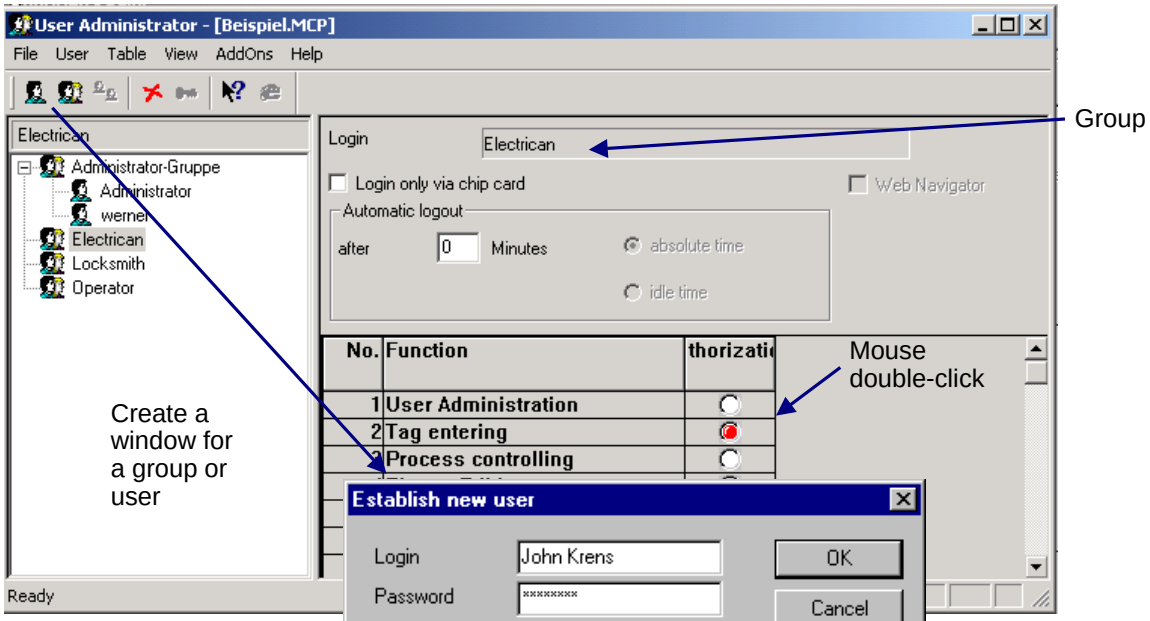
Refer to the example in the slide above for defining the hot keys for logon and logoff.

When you use Ctrl+L, you call a system box in the runtime system. You enter a login name and password in the system box, so that you as the user can carry out password-protected operations. When you use Ctrl+Q, you logoff again, so that no one can come after you and carry out unauthorized operations.
The User Administrator Editor is used to assign the login name and the password.

Note

You may not use Ctrl+P.
Before you begin with configuration, you have to deactivate an active Runtime.

User Administrator (Password Protection) 2/3



Opening the User Administrator

1. Double click on the "Editor" project component in the project navigation window. You then see a list of all installed editors.
2. Use the right mouse button to click on the "User Administrator" editor in the tree structure that opens.
3. Select the "Open" menu item in the shortcut menu that appears. The User Administrator starts.
4. Now you can assign the user authorizations and group authorizations.

Creating a Group

Create...

1. In the menu bar, click on the "User" menu.
2. In the drop-down menu, click on "Group" .

Make the necessary inputs in the displayed window. Refer to the slide above. The login name (user name) must be at least four characters in length.

Creating a User

Create...

1. In the menu bar, click on the "User" menu.
2. In the drop-down menu, click on "User" .

Make the necessary entries in the displayed window. Refer to the slide above. The login name (user name) must be at least four characters in length and the password must be at least six characters in length.

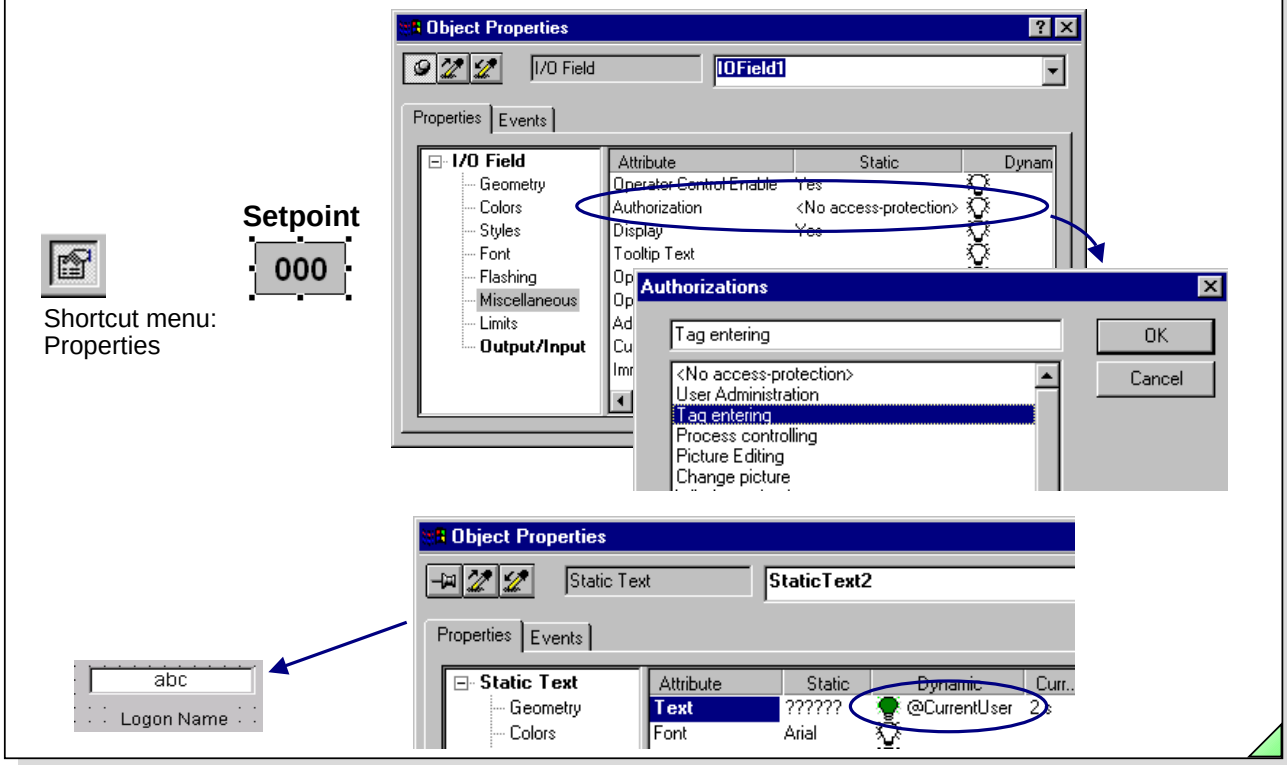
Enabling

You give each user the necessary enables (double click in the "Enable" column) for the authorization levels. You can later change (double click in the "Function" column) the function description that is transferred from the Text Library.

Note

You can only assign a group name and user name once.
The authorizations in the range 1000 to 1099 are system authorizations and cannot be changed by the user.

User Administrator (Password Protection) 3/3



Date: 27.10.2003
File: SWINCC_04e.108

Graphics Designer

In the individual pictures you select the object that you want to protect with a password and then you select the Permission level as described in the User Administrator (example in the slide above).

Runtime

A de-activated runtime must be reactivated after the password configuration.

User name

You display a user name in any picture.
Connect the internal tag **@CurrentUser** to a static text.
As of Version 5, this is generated when you create a project.

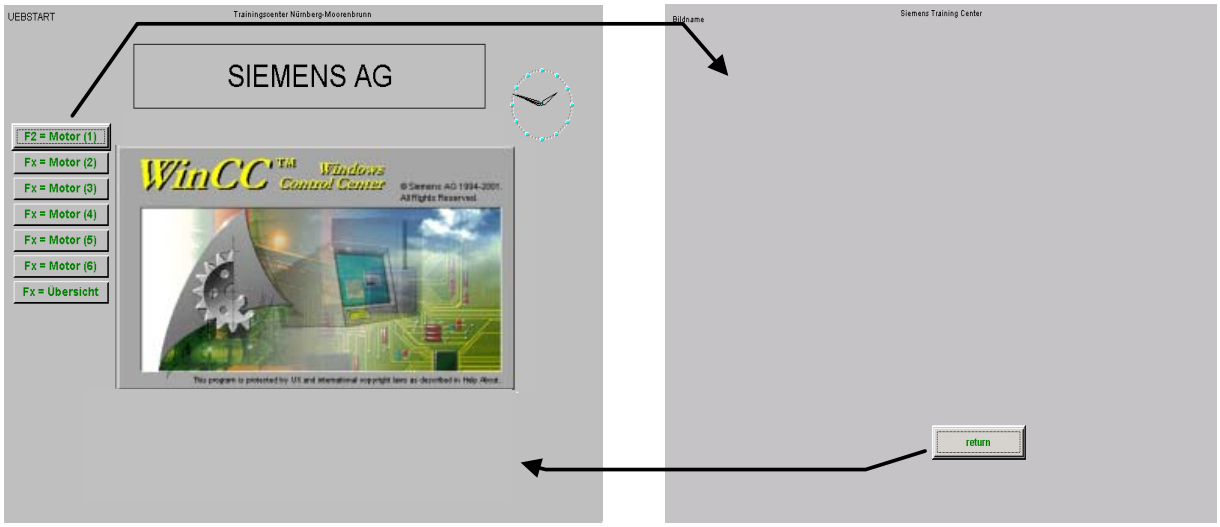
Options

With the **SIMATIC Logon** and **SIMATIC Electronic Signature** option packages you can centrally manage all users of WinCC throughout the whole plant. There are many security mechanisms. Included are: user information is encrypted in a central database, password aging, automatic log-off after a specified time, lock-out after a password has been entered incorrectly several times. Signing the data records provides security against erroneous logging. In this way, the access requirements of 21 CFR Part 11 from the pharmaceutical and food and beverage industry (FDA) are fulfilled.

Exercise 14

Configure the steps from **Pages 88-90** for password protection in the *Start.pdl* picture. The Runtime function is to be exited and the operating of the setpoint speed in the Motor.pdl picture is to be operable depending on which user is loggon on.
As well, the logged-on user name is to be displayed in the Start.pdl picture.

Exercise 1: Creating a Picture



Date: 27.10.2003
File: SWINCC_04e.109

Task

Two pictures are to be created.
You are to go from one picture to the other by means of a simple picture change.
From this picture it must be possible to select the start picture.

Configuring

1. Configure two pictures:

   a. Start picture: Start.PDL
   b. Motor picture: Motor.PDL

2. Configure the start picture and the motor picture with the aid of static text and other static elements.

3. Configure the Smart Object->OLE Object and link it with the file C:\SIEMENS\WinCC\bin\Mcpsplsh.bmp (pay attention to the installation path of your PC).

4. Configure a button for the picture selection function.

5. Configure both picture selection functions (switch to Motor.PDL and back; see slide above).

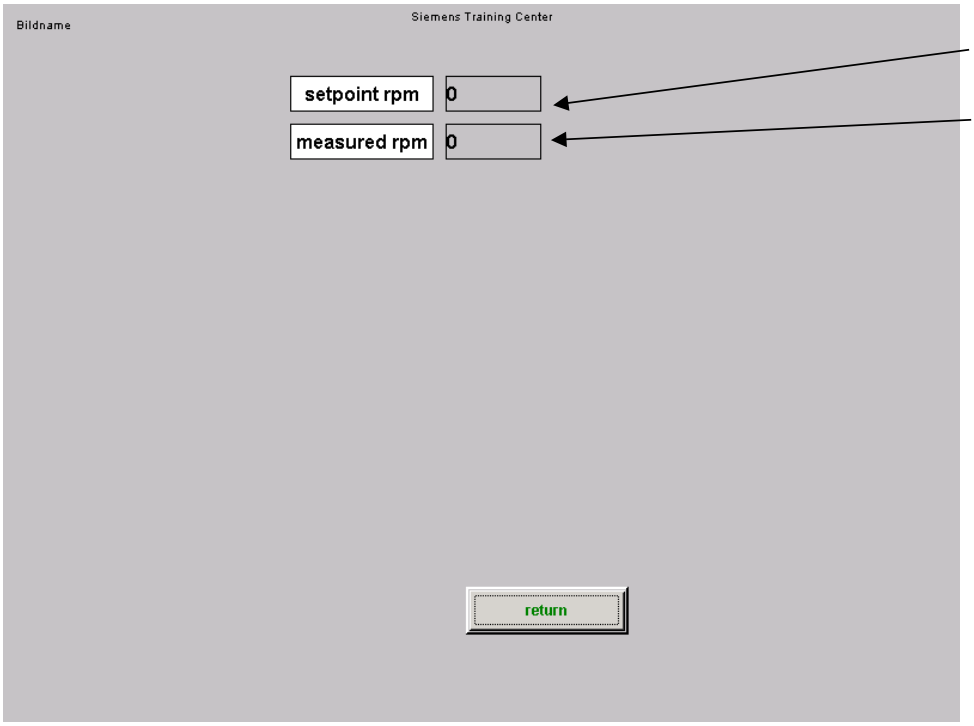
6. Save the picture with the above-mentioned name.

7. Repeat steps 2,4-6 for the motor picture.

8. Select the start.PDL start picture for the Runtime mode.

9. Save the pictures and test the functions.

Exercise 2: I/O Fields for Setpoints and Actual Values



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.110

SITRAIN Training for Automation and Drives

Task

Two I/O fields in the motor picture are to be created.
One of the I/O fields is needed as the input/output field for the setpoint speed. The second I/O field functions only as the output field for the actual speed.

Configuring

1. Configure two I/O fields:

   a. Setpoint speed
   b. Actual speed

2. Configure the I/O field for the setpoint speed as an input field and output field.

3. Configure the I/O field for the actual speed as an output field.

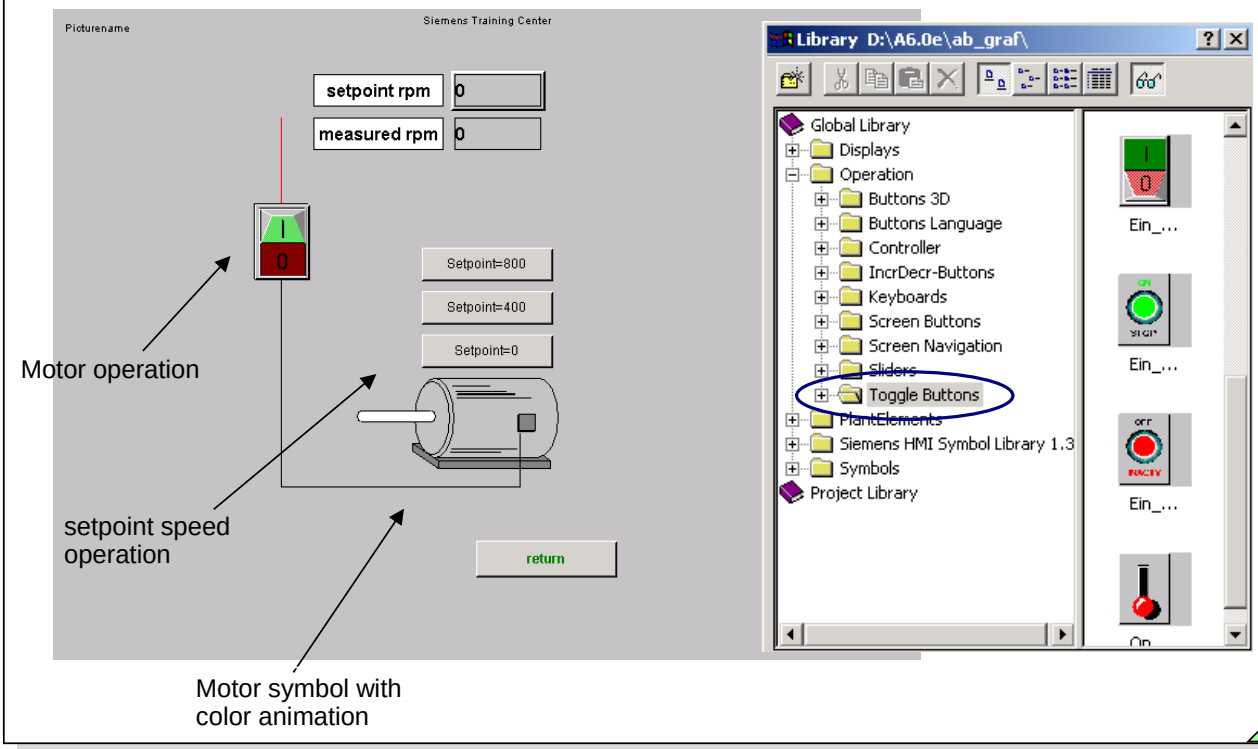
4. Connect the I/O fields with the setpoint speed and actual speed tags.

5. You can use "Properties" to set the attributes (such as font size, number of digits, color, etc.) for the I/O fields.

6. In the start picture, configure the system function Exit Runtime and Windows with the Dynamic Wizard.

7. Save the picture and test the functions.

Exercise 3: Motor Operation and Motor Color Change



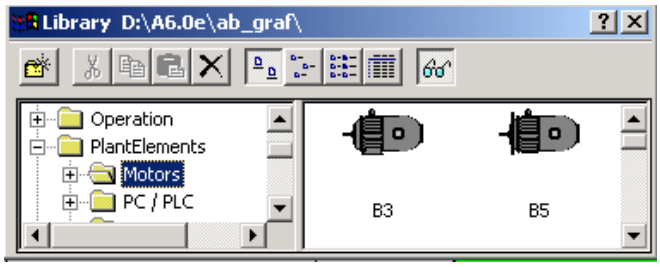
Date: 27.10.2003
File: SWINCC_04e.111

Task

The color of the motor is to be changed to display the motor status (on/off). (You can use the motor display from the Library).
You can use either the Dynamic Dialog or even a C-action for the color change.
You are to configure an operation with which a motor can be switched on and off (You can use the toggle button from the symbol library).
This toggle button also indicates the current switching state.
You can operate the setpoint speed using the setpoint speed button.

Configuring

1. Select a motor display from the Library and place the selected symbol in your picture by using the "drag and drop" feature.



2. Make the color of the motor display (group) dynamic with the aid of the Dynamic Dialog using the "Motor1" tag as input.

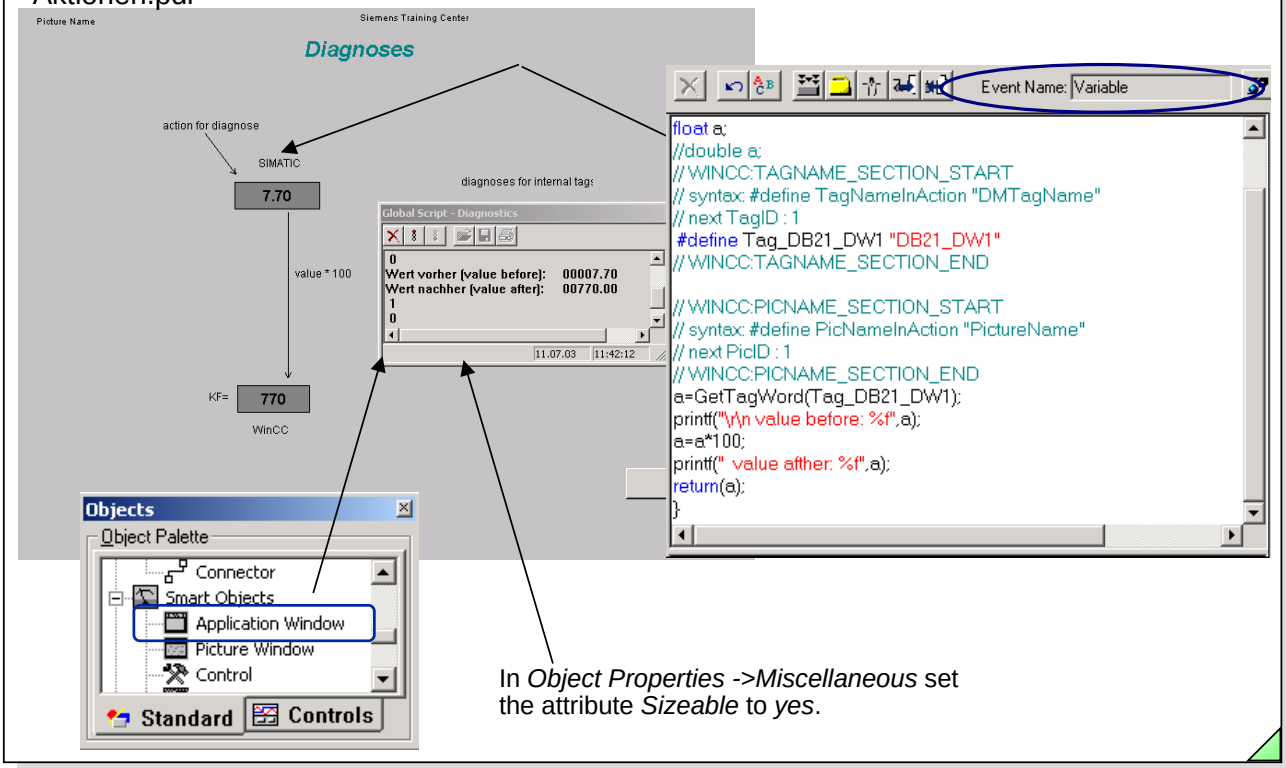
3. Select a toggle button from the symbol library, see slide above, and place the selected symbol in your picture using "drag and drop".
4. Make the toggle button dynamic with the aid of the Process driver connection property using the Switch1 tag.

5. Configure the three setpoint buttons for the setpoint speed operation of 0, 400 and 800 rpm. Use the Direct connection with the Setpoint speed (setpoint speed) tag.

6. Save the picture and test the functions.

Aktionen.pdl

Exercise 4: Calculation of an Output Value with C



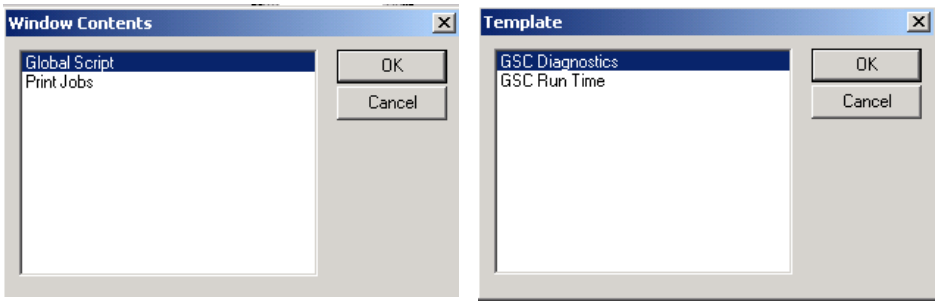
Date: 27.10.2003
File: SWINCC_04e.112

Task

The length measurement in meters from the SIMATIC is to be displayed in centimeters through a calculation in WinCC.

Configuring

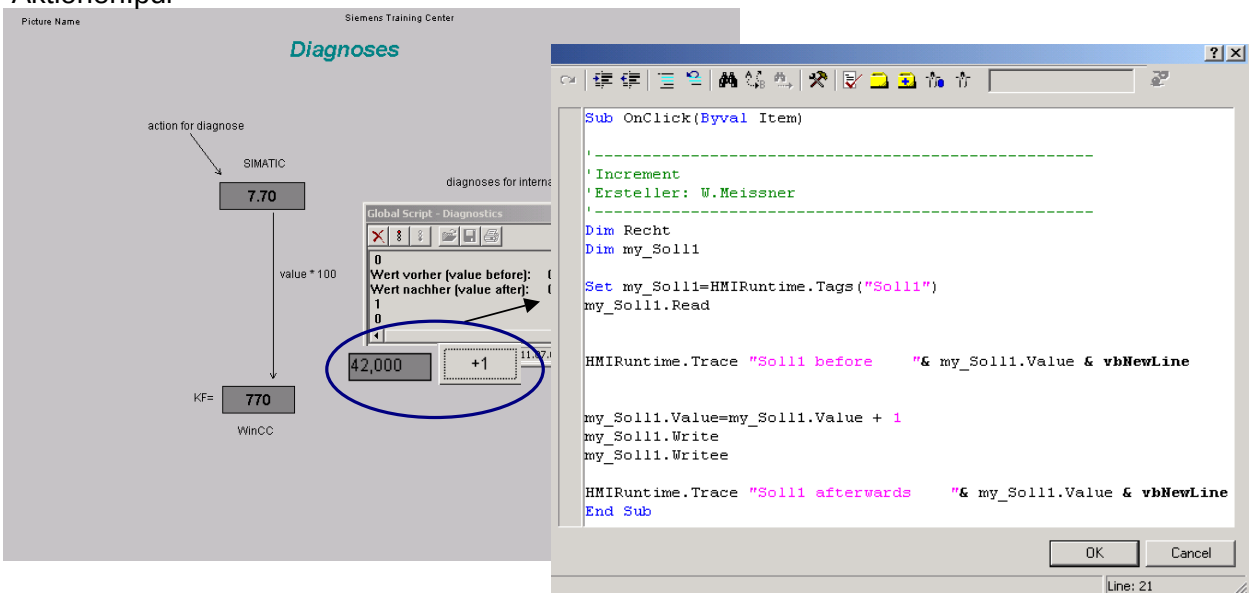
1. Create a new picture with the name Aktionen.pdl and call it from the start picture .
2. Create the internal tag DB21_DW1 of the floating point number data type 32Bit IEEE754.
3. Create two I/O fields and make the necessary attribute settings.
4. Edit the Application window and set the attribute Sizeable to yes.



5. Accept the C Script shown in the slide above in the attribute output value of the I/O field.
6. Save the picture and test the function.

Exercise 5: Calculation of an Output Value with VBS and Troubleshooting

Aktionen.pdl



Date: 27.10.2003
File: SWINCC_04e.113

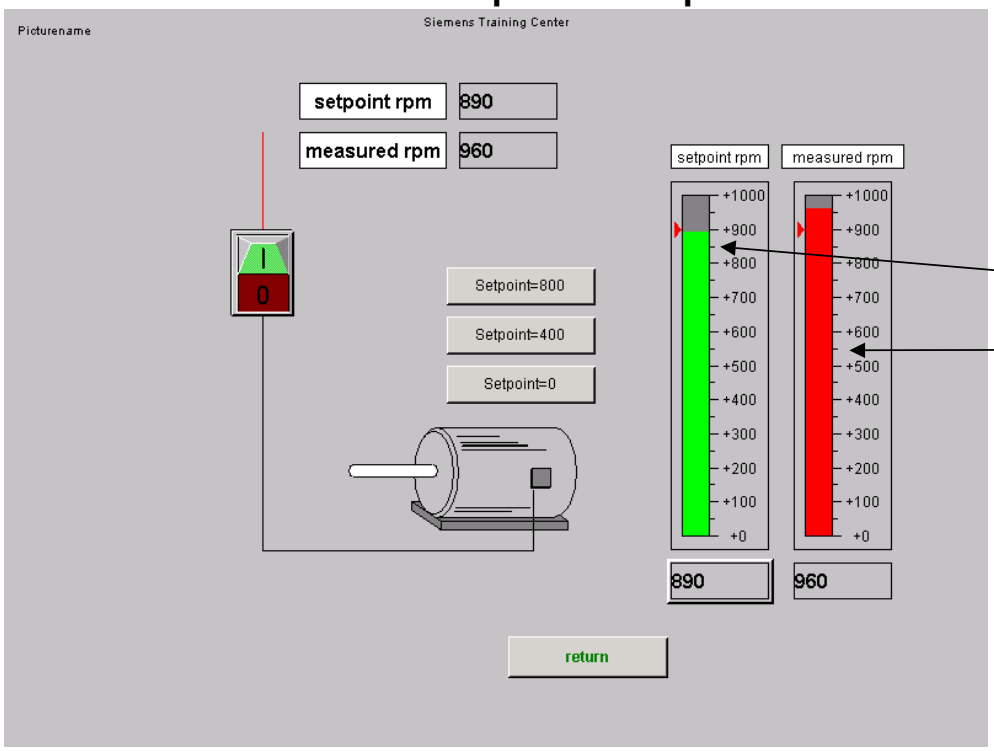
Task

The value of a tag is to be incremented per mouse click using a Window button. The operating value is also to be displayed in an output field.

Configuring

1. Call the picture with the name Aktionen.pdl.
2. Create the internal tag Setpoint1 of the signed 16 bit value data type.
3. Create an output field and make the necessary attribute settings.
5. Accept the VBS Script shown in the slide above in the Event Mouse Click.
6. Save the picture and test the function.

Exercise 6: Bar Graphs for Setpoints and Actual Values



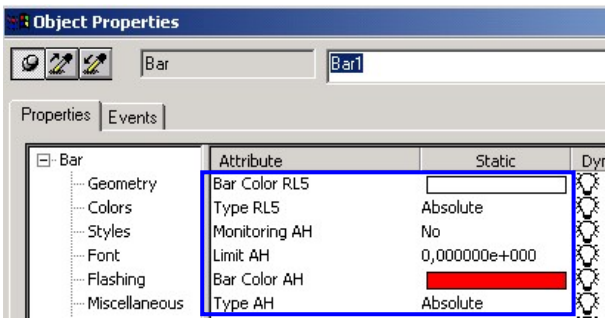
Date: 27.10.2003
File: SWINCC_04e.114

Task

Two bar graphs are to be created. One bar graph is to display the setpoint speed, and the other bar graph is to display the actual speed.

Configuring

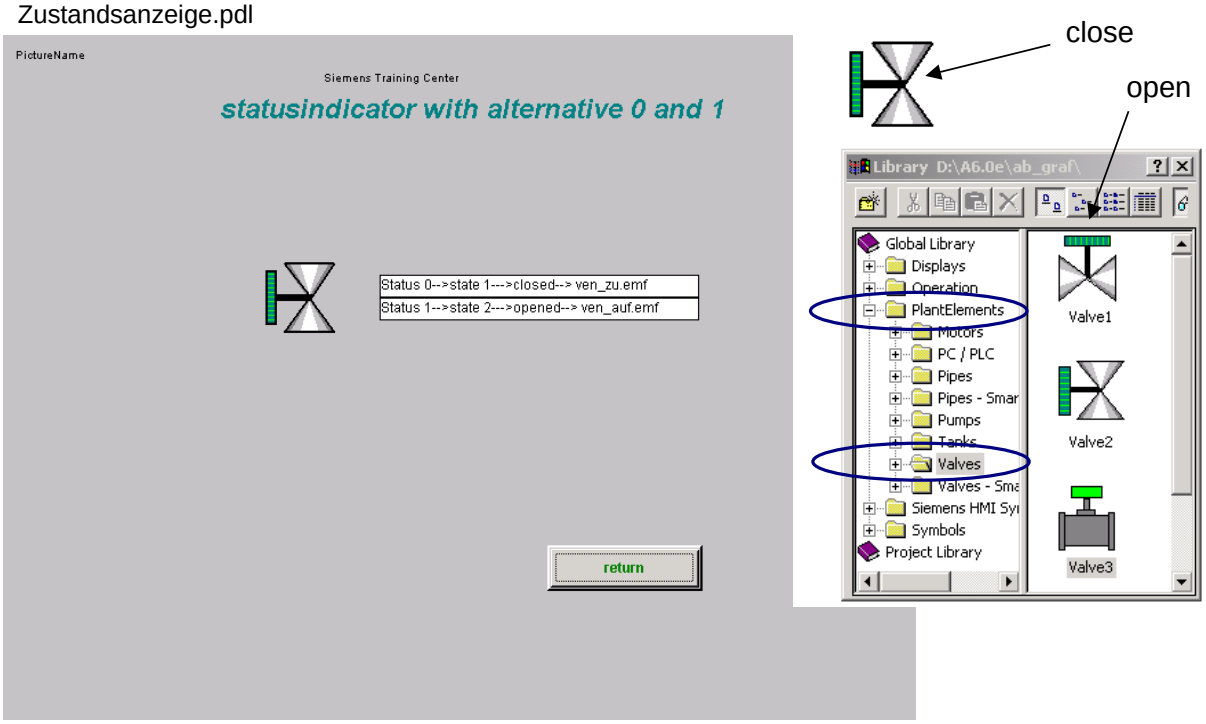
1. Configure two bar graphs:
   a. Setpoint speed
   b. Actual speed
2. Connect the bar graphs with the "setpoint speed" and "actual speed" tags.
3. Configure the lower display limit with 0 and the upper display limit with 1000.
4. You can use "Properties" to set the attributes (such as font size, number of digits, color, etc.) for the bar graphs.
5. In the Properties->Limits->Limit Marker select Display.
6. When the speed exceeds >900, the bar graph color is to switch to red.



7. Save the picture and test the functions.

Exercise 7: Status Display

Zustandsanzeige.pdl



Date: 27.10.2003
File: SWINCC_04e.115

Task

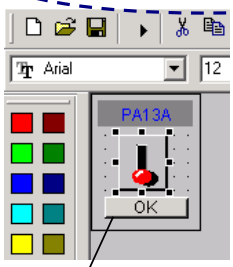
You are to configure a valve indicator to display the valve operation open and closed. The configuration is made with the status indicator that can also be used for other object changes.

Configuring

1. Create a new picture with the name Zustansanzeige.pdl and call it from the start picture.
2. Select a valve indicator from the symbol library (see slide above right) and place the selected symbol in the picture using "drag and drop".
3. Repeat Step 2 for the second valve.
4. Click on File in the menu and then click Export... in the drop down menu to export the objects individually into the .emf format.
5. Configure the Status display from the Smart Objects.
6. In the configuration dialog that appears, select a valve indicator from the picture selection window and place it in the Status Table, column Basic Picture (for the assignment of the states see slide above) using "drag and drop".
7. Create the process tag color valve of the Binary Tag data type with the address I 0.0 and connect the process tag to the status display.
8. Save the picture and test the function.

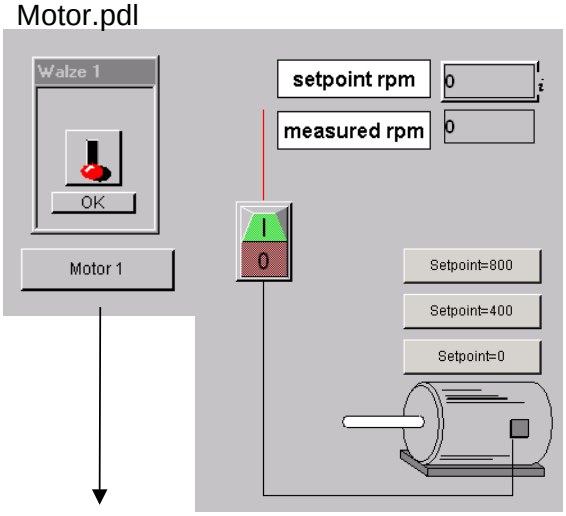
Exercise 8: Two-Step Operation

Operating picture: Mot1.pdl



Configuration with Direct Connection:
-Source: Constant *0*
-Target: Current Window-> *Display*

Motor.pdl



Configuration with Dynamic Wizard:
-Picture Function
-Display WinCC Dialog Box with *Mot1.pdl*

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.116

SITRAIN Training for Automation and Drives

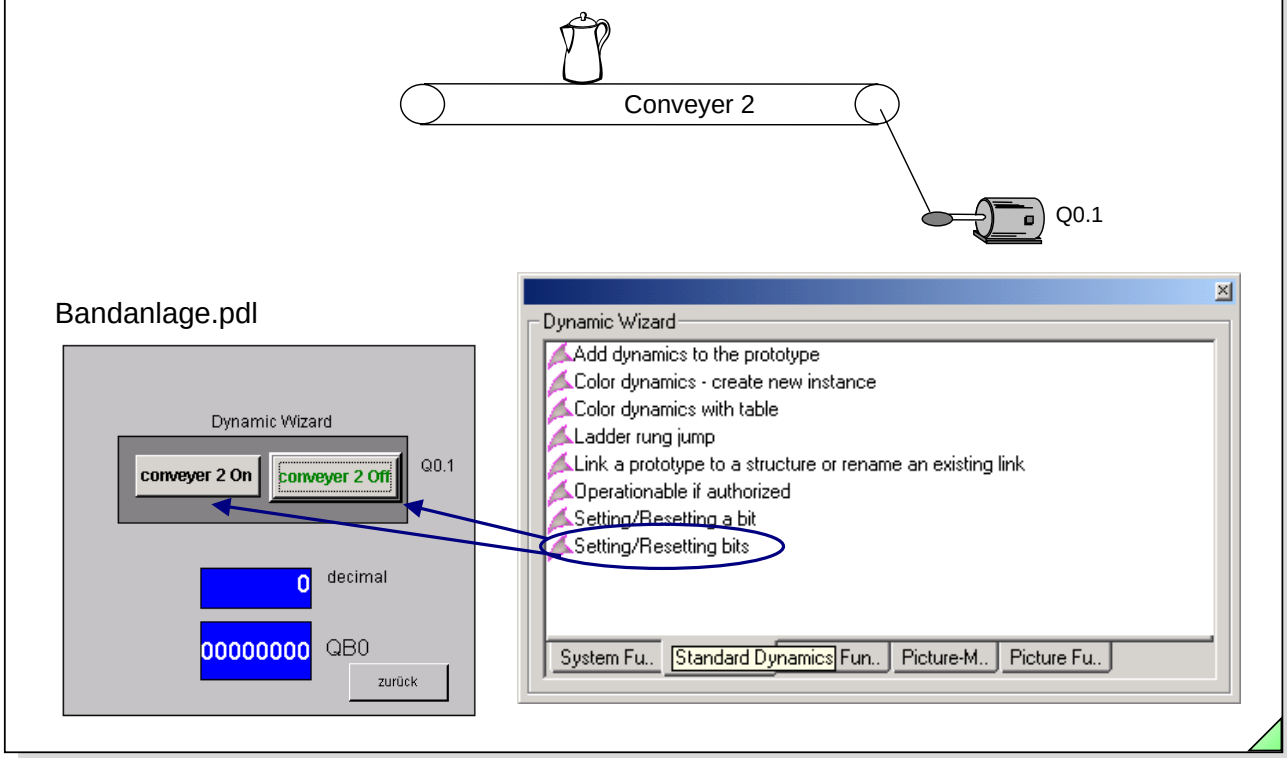
Task

So that you have a higher switching security, the switching on and switching off of a motor is to take place using a toggle button in a two-step operation.

Configuring

1. Create the operating picture with the name Mot1.pdl
2. Edit a button with the label Motor1 in the motor picture.
3. Start the Dynamic Wizard and generate the function Display WCCDialog box
4. Save the picture and test the functions.

Exercise 9: Setting a Bit in a Byte



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_04e.117

SITRAIN Training for Automation and Drives

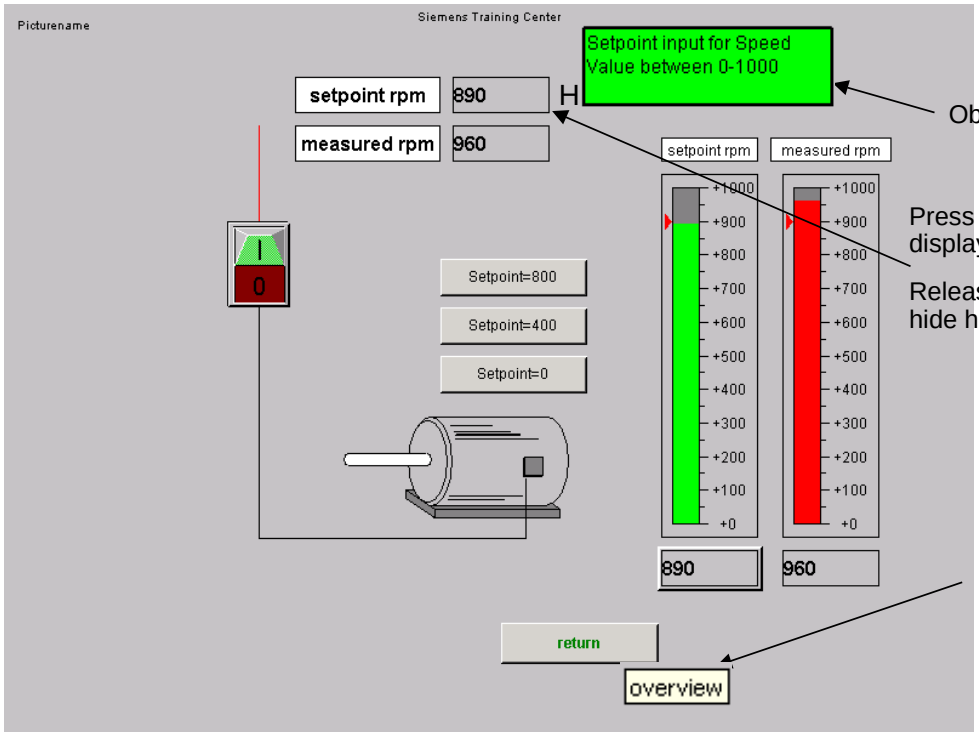
Task

A conveyor belt is to be switched on and off using two buttons.
The switching action results in a bit being set and reset in a byte.
The unused bit addresses can be used in the control program for other operations.

Configuring

1. Create the operating picture with the name Bandanlage.pdl
2. Create the tag Conveyor as signed 8 bit value with the address QB0.
3. Start the Dynamic Wizard for every Window button and generate the function Setting/Resetting Bits.
4. Save the picture and test the functions.

Exercise 10: Help Function



Date: 27.10.2003
File: SWINCC_04e.118

Task

A help text is to be configured for the I/O field of the setpoint speed. This help text is to display an input note regarding the input limits when the right mouse button is pressed.
A tooltip text is to be displayed on the button ESC=BACK, that appears when the cursor points to the object.

Configuring

1. Call the motor picture and configure a static text with the two-line help text: Input values between 0-1000. Then carry out the necessary attribute settings.

2. Call the direct connection using the I/O field of the setpoint speed, by pressing the right mouse button.

   -Source: Constant 1

   -Target: Object in picture, Help text, Display

3. Call the direct connection using the I/O field of the setpoint speed, by letting go of the right mouse button.

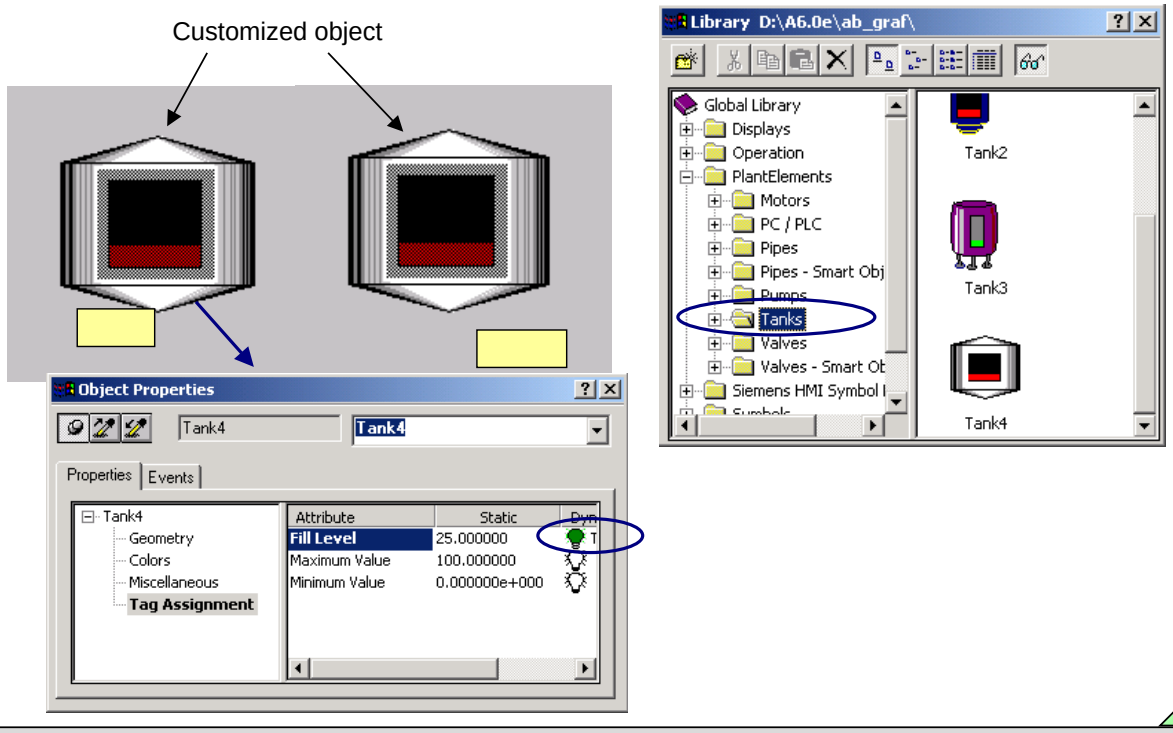
   -Source: Constant 0

   -Target: Object in picture, Help text, Display

4. Configure the Tooltip text for the BACK Window button.

5. Save the picture and test the functions.

Exercise 11: Customized Object



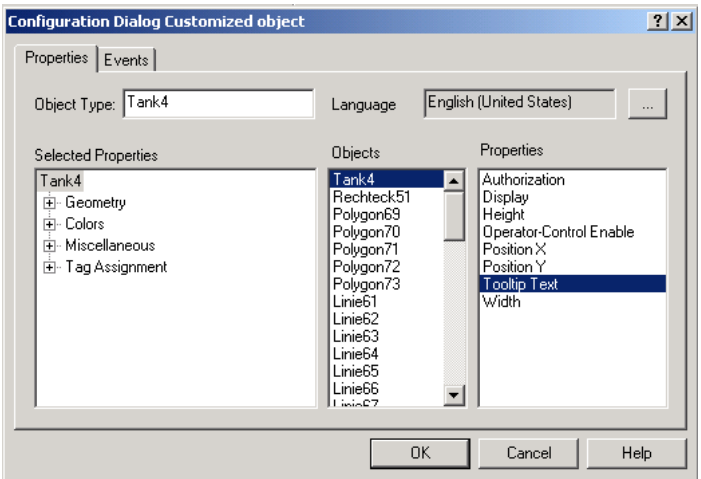
Date: 27.10.2003
File: SWINCC_04e.119

Task

A help text is to be displayed on Tank 1 and Tank 2. The help text is to be configured as a tooltip text.
Through the object properties, you can see that the Tank was stored as a customized object in the symbol library. This customized object is now to be expanded with the attribute Tooltip text using the customized object configuration dialog.

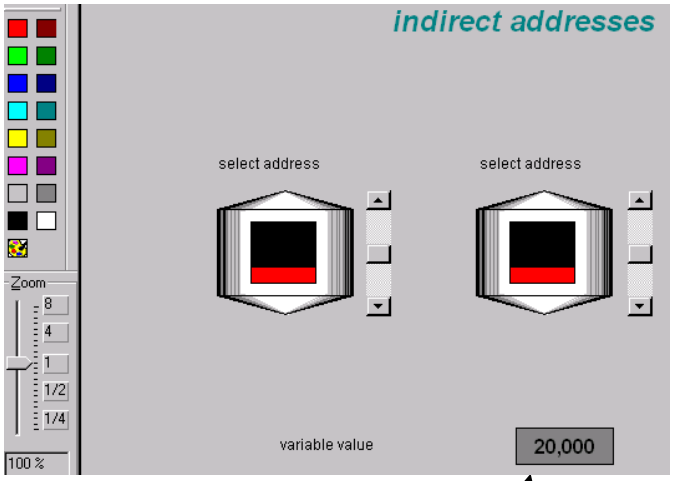
Configuring

1. Create the system picture with the name Indirekte_adresse2.pdl as well as the objects as shown above in the graphic picture.
2. Create the internal tags I_Value_1 , I_Value_2 of the signed 16 bit data type for Tank 1 and Tank 2.
3. Expand the attribute for the Tank in the called picture by adding the attribute Tooltip text.



4. Configure the tooltip text Acid and Water.
5. Save the picture and test the function.

Exercise 12: Indirect Addressing



textvariable with indirect addressing

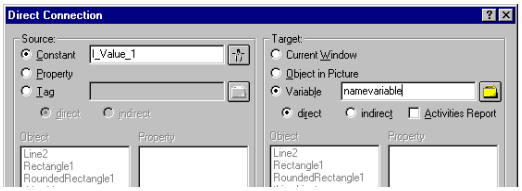
Task

The levels of several tanks are to be displayed using one output field.

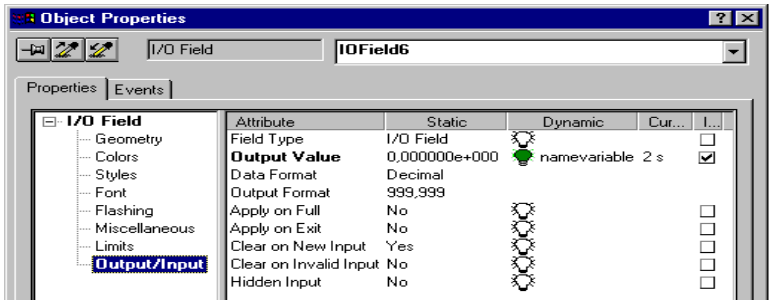
Configuring

1. Create the internal tag textvariable of the Textvariable 8 bit character string data type.

2. The assignment of the internal variable textvariable takes place using the direct connection by mouse click on the tank symbol. The source parameter of the target variable is then passed.



3. So that the value of the tag I_Value_1 can be displayed in the I/O field using the attribute Output value, you must enable the variable textvariable in the column Indir. using the button.



4. Save the picture and test the functions.

Message Display and Message Archiving 1/2

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.1

SITRAIN Training for Automation and Drives

Contents Page

Message Display and Message Archiving 2/2

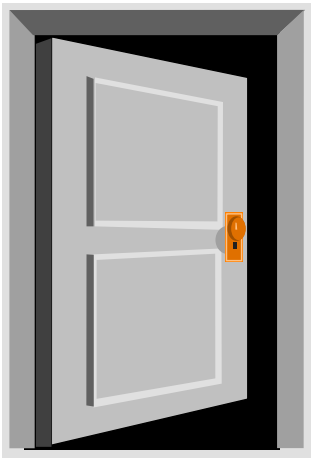
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.2

SITRAIN Training for Automation and Drives

Contents Seite

Objectives



- Edit Message Line
- Edit Messages
- WinCC Alarm Control
- Message Classes
- Message Archives
- Group Messages

SIMATIC HMI

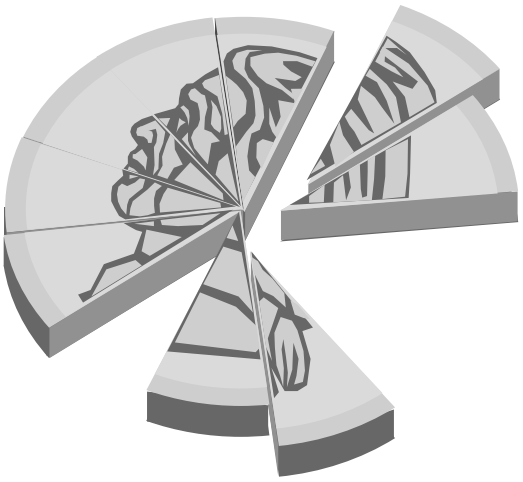
Date: 27.10.2003
File: SWINCC_05E.3

SITRAIN Training for Automation and Drives

Upon completion of this chapter, the participant should be able to:

- know the advantages of the message system
- start the message system
- adapt the message system to the plant requirements
- configure messages
- create message lines
- define message classes
- define message types
- set up a message archive
- identify group messages
- identify the export and import functions
- identify the loop in alarm
- identify message selections
- display messages in a screen
- archive messages

Configuration Preparation



- Message Archive Structure
- Acknowledgement Concept
- Evaluation
- Links

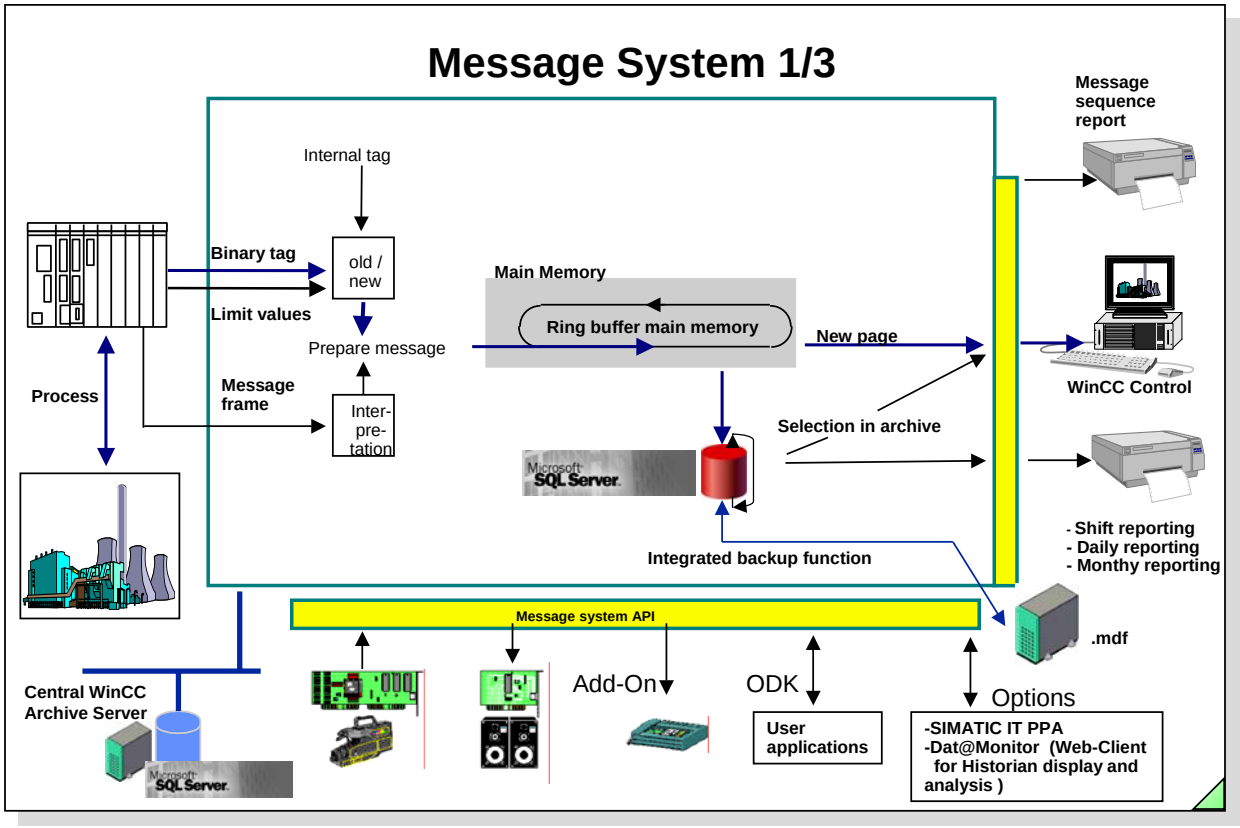
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.4

SITRAIN Training for Automation and Drives

⇨ Configuration Preparation

- ❒ Process connection
  - • defining tags with the bit address assignment (bit message procedure)
- ❒ Message archiving
  - • grouping all messages with number assignment
  - • specifying message blocks
  - • defining contents and display for the message format
  - • specifying the message picture
  - • specifying the acknowledgement concept
  - • specifying the archiving time period
  - • specifying analysis and evaluation routines
  - • specifying reports
  - • specifying process values
  - • assigning message classes
  - • specifying messages from logic operations



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.5

SITRAIN Training for Automation and Drives

General Information

By means of the message system, events from the PLC or from the monitoring function in WinCC (operating states, errors, etc.) are displayed in the form of messages. The events are archived, if applicable, reported and acknowledged, and received by the operator.

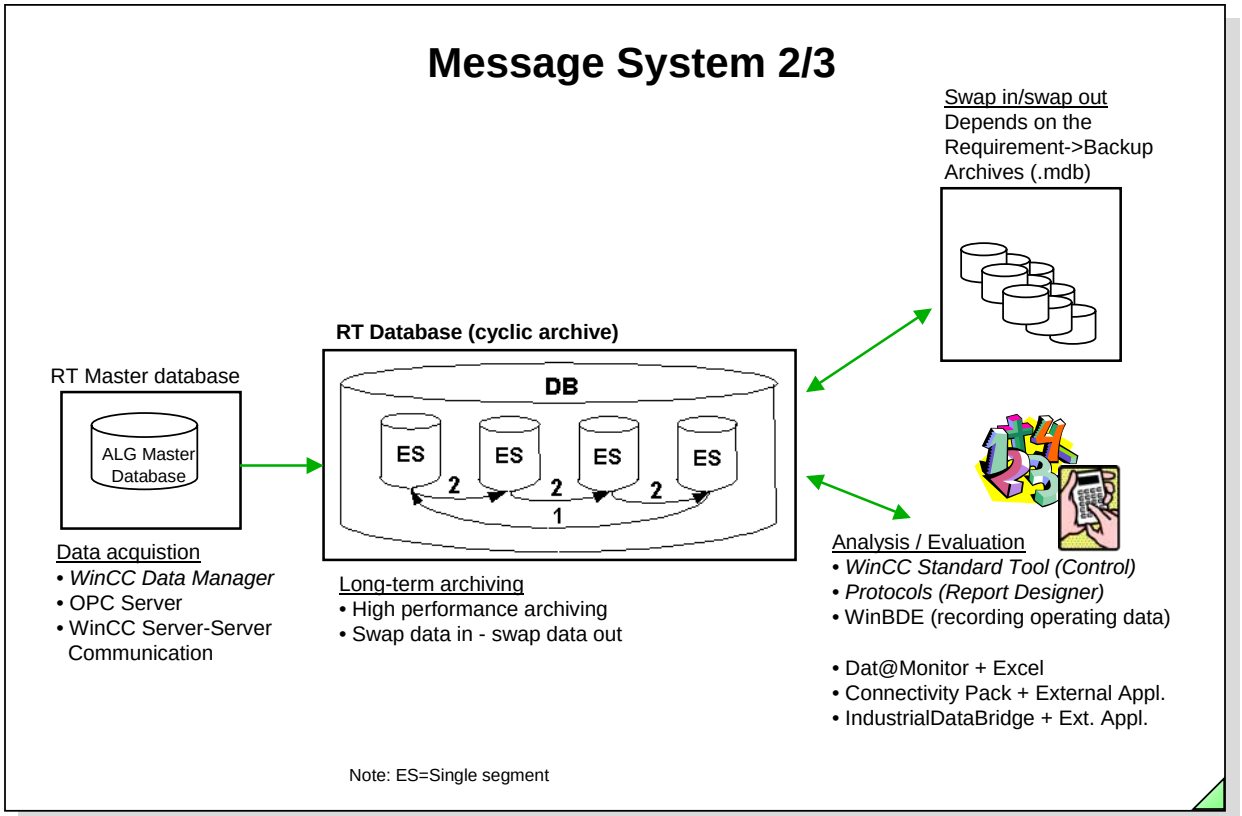
For this purpose, messages must be configured that are divided into message classes.

To also be able to view the history, the messages are saved on the hard drive in a long-term archive. The WinCC message system is based on the information in DIN 19235.

Message System

Bit message procedure and chronologically correct reporting

- Message acknowledgements, even to programmable logic controllers
- Single messages and group messages
- 16 message classes with 16 message types each
- Display using line-oriented messages and graphics
- Acoustical signaling using a soundcard, City Call and SMS
- Views of the message list and archive you can define
- Loop in Alarm
- Disabling/Enabling and selection of messages
- Message comments, Message information
- Opening for analysis and evaluation programs
- Trigger actions
- Configurable: 50,000 messages
- Permanent message load without loss (single-user/server) 10/sec
- Long-term archive, in RT no system limits (depends on the hard drive)



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.6

SITRAIN Training for Automation and Drives

Archiving

For archiving messages, WinCC uses cyclic archives (ring buffer) of a configurable size. You can configure them with or without backup.
The archive files are always stored on the local computer in the associated project.
The WinCC message archive consists of multiple single segments. You can configure both the size/time of the message archive and the size/time of the single segments in WinCC.
Example: The message archive archives all messages that occur within one week (1), each single segment archives the messages that occur within one day (2).
You can always configure both conditions at the same time. If either of the two criteria (time or size) is exceeded, the following happens:
Criteria for message archive (DB) is exceeded (1)-> the oldest messages (that is, the oldest single segment) are deleted.
Criteria for single segment is exceeded (2)-> a new single segment (ES) is created.
A new single segment is also created when you configure message data online (for load Delta online).

Dat@Monitor

For visualization and evaluation, WinCC/Dat@Monitor provides a series of Internet-capable tools for viewing (view only) and online analysis. These support all the usual security mechanisms such as login/password, firewalls, encrypting etc..

Connectivity Pack

Access to WinCC using OPC and OLE-DB. Using OLE-DB, you can directly access the archive data stored by WinCC in the MS SQL Server database. The OPC HDA 1.0 (Historical Data Access) and OPC A&E 1.0 (Alarm & Events) servers enable you to access historical WinCC archive system data and to pass on / acknowledge messages.

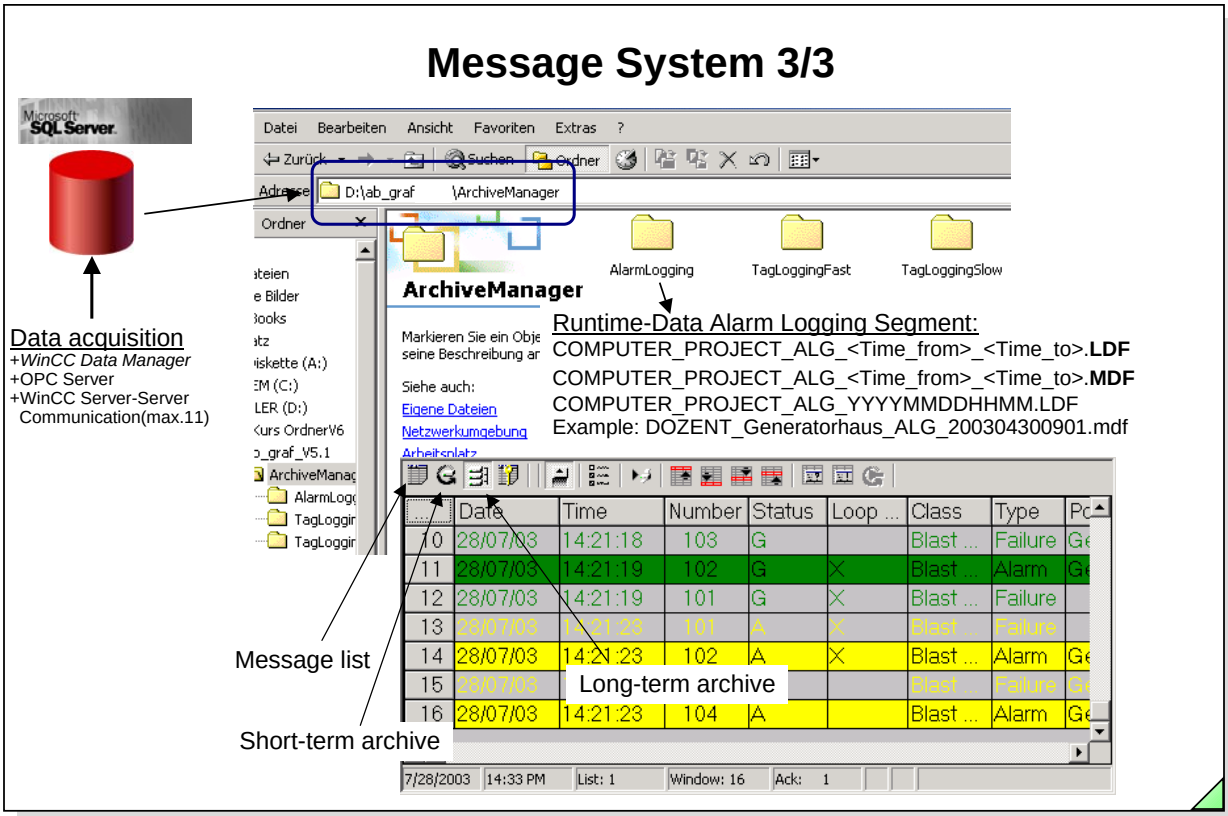
IndustrialDataBridge

Configurable connection to databases and IT systems.

Note

A message requires about 120 bytes of memory.

Message System 3/3



Name RT-DB

The runtime data are distributed amongst the master database and several runtime databases.

The master database manages the runtime databases and refers to the individual runtime databases. The master database is created in the project directory. The name of the master database consists of the project name with a trailing "RT".

The runtime databases each contain the archived data for a specific time period and are stored in the project directory in a subdirectory "ArchiveManager/Alarm Logging".

The name of the runtime database consists of:
+ computer name
+ project name
+ ID for the type of data (ALG = Alarm Logging, TLG_F = Tag Logging Fast, TLG_S = Tag Logging Slow)
+ date/time (year, month, day, hours, minutes).

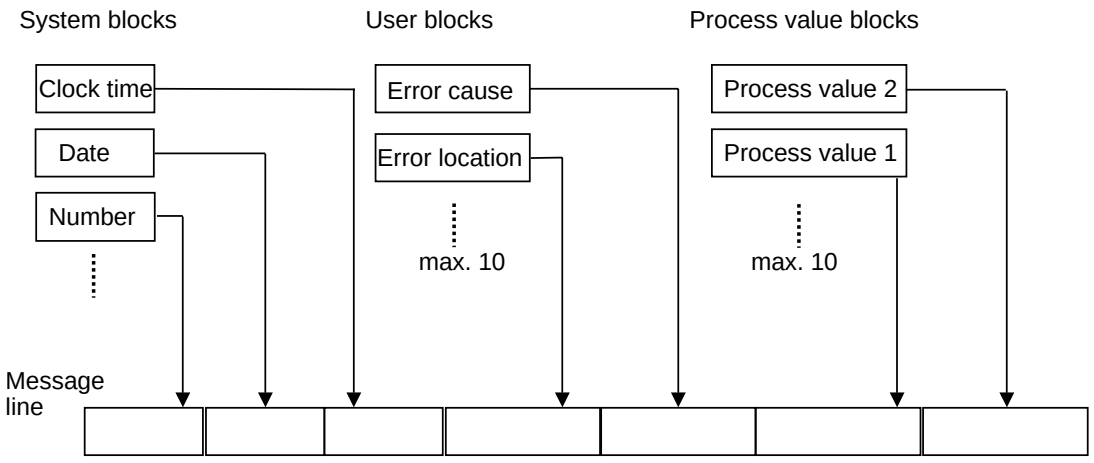
If the time of the segment change is less than the RT Start time, an additional file is created.

Such as, DOZENT_Generatorhaus_ALG_200304291201_200304300701.mdf. This file structure also indicates that messages were recorded in this time period.
If the file only contains the time of the segment change, then no messages were recorded in this time period.

Note

The "Short-term archive" and "Long-term archive" buttons display the same archive data in runtime. In the long-term archive, you can also store the online comments. In the short-term archive window, you can display up to 1000 messages (archive and current messages).
During the migration of data, only the long-term archive is migrated. The data in the short-term archive are lost.

Message Blocks and Message Line



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.8

SITRAIN Training for Automation and Drives

General Information

Messages are displayed as tables within an ActiveX (message window). From the system blocks, process value blocks, and user text blocks, you specify which columns these tables can have using the “Message Line” tab.

System Blocks

The system blocks provide system information, such as the current time of day (clock time).

User Text Blocks

User text blocks (max. 10) output the message text the user edited.

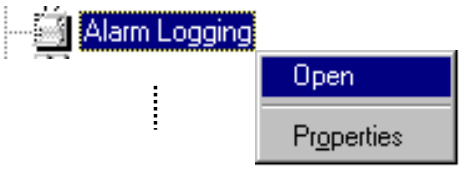
Process Value Blocks

Process value blocks (max. 10) display the value of a tag.

Note

You can design the message line format resulting from the message blocks as you want and this is created in the WinCC Alarm Control window.

Message System Call-Up



1. **Call the shortcut menu.**
2. **Click on the mouse to select “Open.”**

   Result: Alarm Logging starts.
3. **Maximize the window and set up the interface as you want it.**

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.9

SITRAIN Training for Automation and Drives

Task Definition

Configuring the message system with message blocks, messages, message classes, message archive and the WinCC Alarm Control.

The configuration must then undergo a function test.

The explanation of the configuration shows the settings necessary for the exercise.

Prerequisite

If no message events have been configured up until now, then they should be configured now before additional message configurations are carried out. (See Chapter 3 for configuration instructions.)

Message System Call

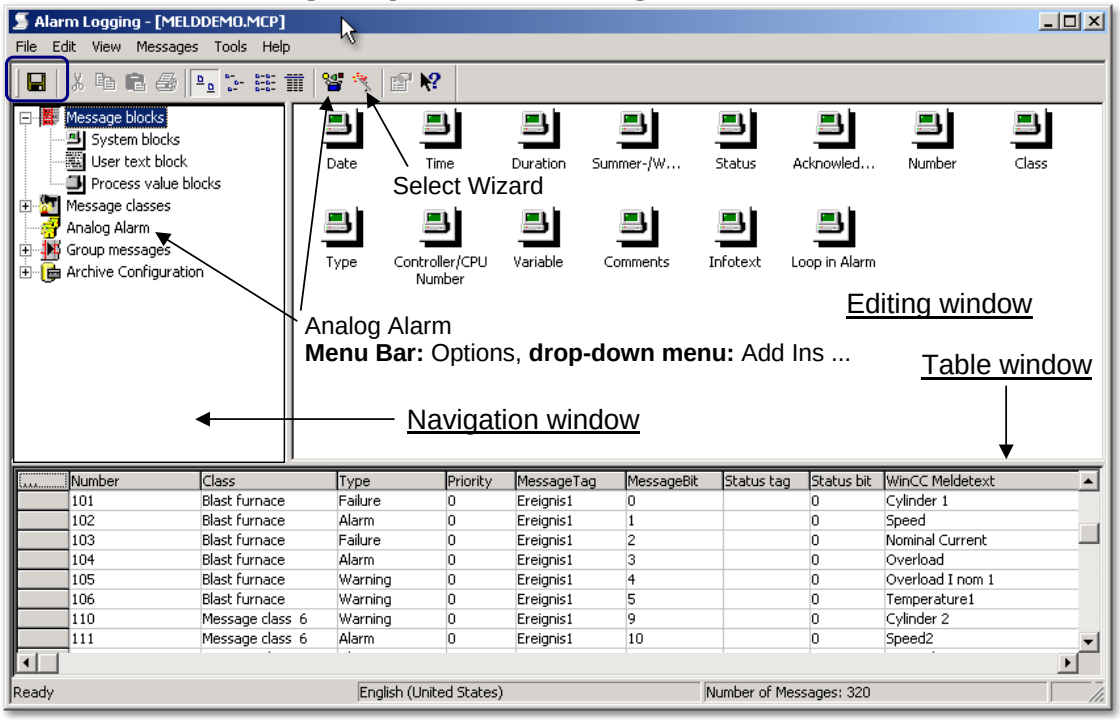
Refer to the slide above.

Exiting the Message System

1. In the **menu bar**, click on **“File.”**
2. In the **drop-down menu**, click on **“Exit.”**

If a message system was open and you carried out changes and you have not saved yet, a dialog box appears after the “Exit” command. The dialog box provides you with the opportunity to save your changes.

Message System Configuration Interface



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.10

SITRAIN Training for Automation and Drives

General Information

After the message system is called, the division displayed in the slide above is displayed on the monitor.

- The left window displays the editors in a tree structure.
- The right window displays the result of configuring the respectively selected editor.
- The table window in the lower part of the screen represents the editing possibilities for texts, tags, and attributes for the message number.

System Wizard

In the menu bar, click on File, in the drop-down menu, Select Wizard, System Wizard

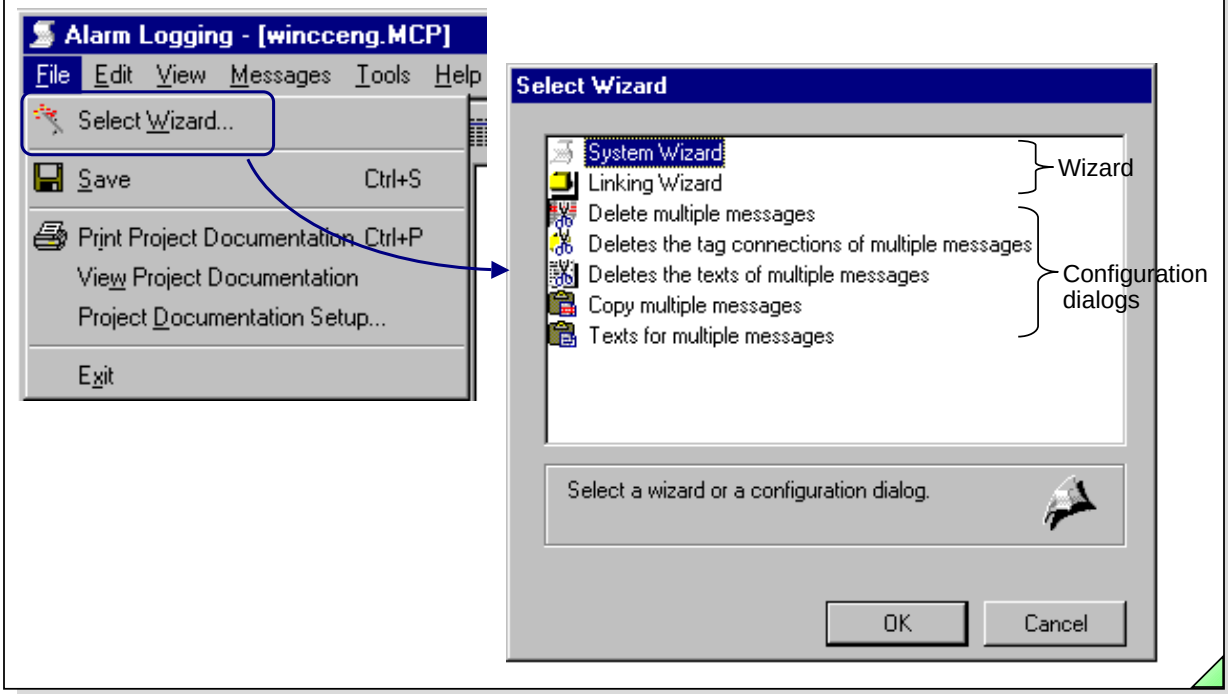
To make it easier for you to begin configuring, the message system has a configuration assistant (System Wizard). This configuration assistant takes all significant settings into consideration in order to set up an executable message system quickly. The System Wizard orients itself on the tree structure displayed in the top left portion of the slide above. You can use the respective editor to change the defaults at any time.

Analog Alarm

In the menu bar, click on Options, in the drop-down menu, Add Ins....

You can enter the number of a message, that will be created in Runtime as soon as an analog alarm appears.

Wizard Overview



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.11

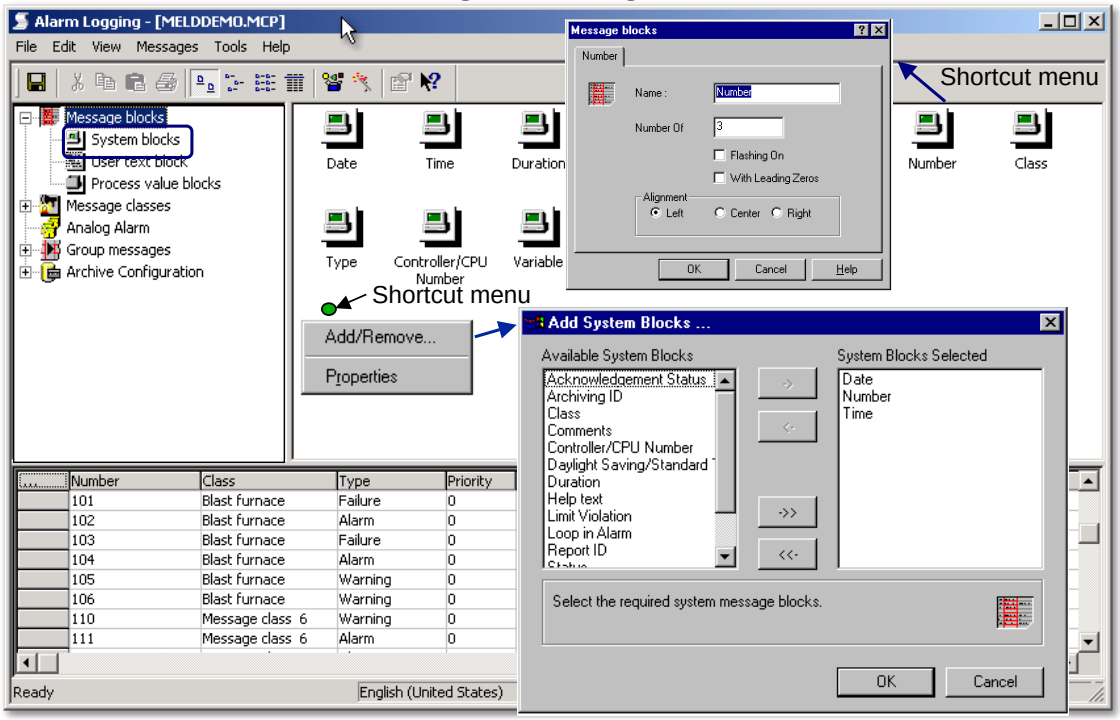
SITRAIN Training for Automation and Drives

System Wizard The System Wizard supports you in creating the default settings for the entire Alarm Logging System.

Linking Wizard The Linking Wizard simplifies the linking of (process) tags to messages.

Configuration Dialogs. Several Configuration Dialogs can be used to edit several messages at the same time.

Editing Message Blocks



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.12

SITRAIN Training for Automation and Drives

System Blocks

From the system blocks available in the system, you can use a mouse to choose a selection dialog (see slide above) to select the system blocks, for example, the date. It is possible to remove system blocks at a later time.

Editing System Blocks

Use the mouse to click on the + character for message blocks in order to open the object structure for the message blocks.

Use the System Blocks “Add/Remove” shortcut menu to call up the selection screen.

System Block Shortcut Menu

Use the shortcut menu for the “Number” system block to call up the “Properties” instruction. You can now change the length of the number (number of positions), flashing, and the name of the column title in the called up window.

This procedure is the same for all system blocks.

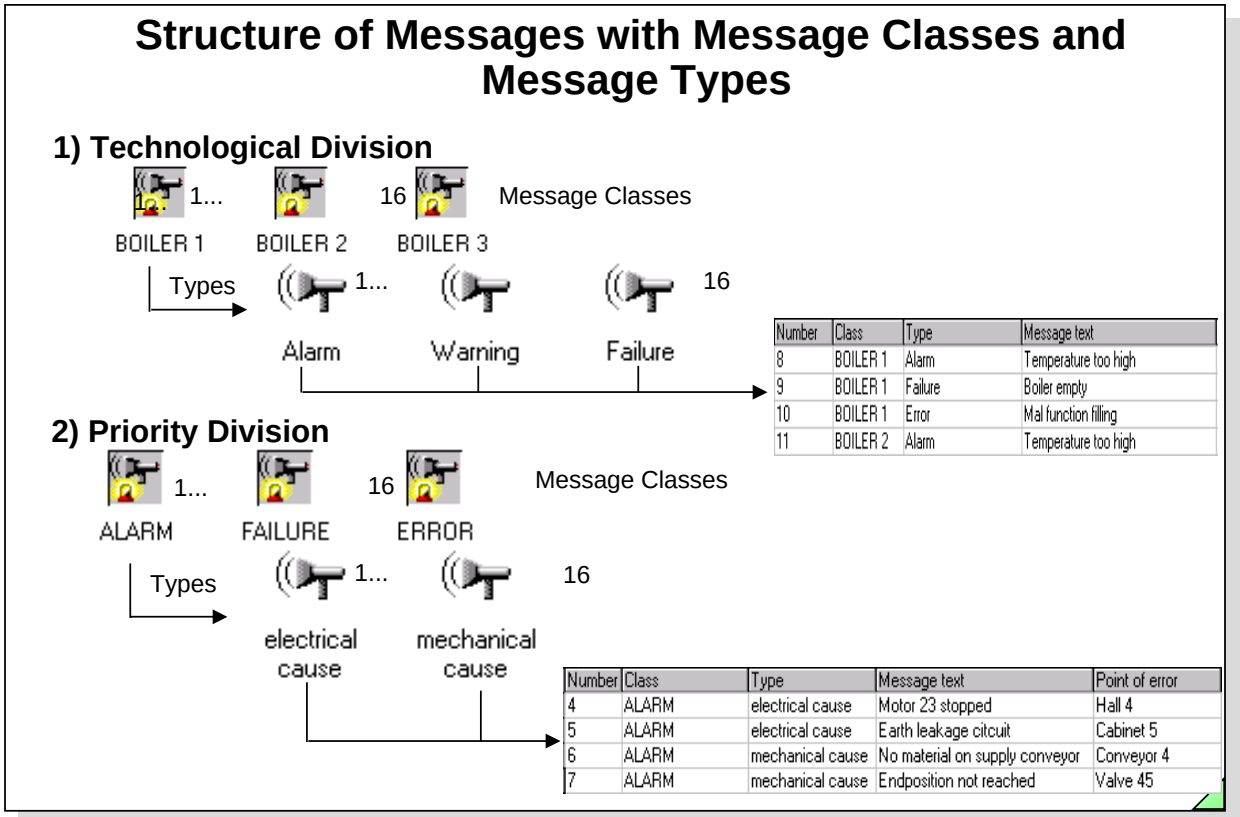
Editing User Text Blocks

Editing user text blocks and changing the properties takes place according to the same rules as for system blocks. You have ten user text blocks available under WinCC. You can change the user text block labels and lengths (for example, block 3: -> error location).

Editing Process Value Blocks

Editing process value blocks and changing the properties takes place according to the same rules as for system blocks. You have ten process value blocks available under WinCC. You can change the process value block labels and lengths (for example, process value 1: -> pressure).

Structure of Messages with Message Classes and Message Types



SIMATIC HMI

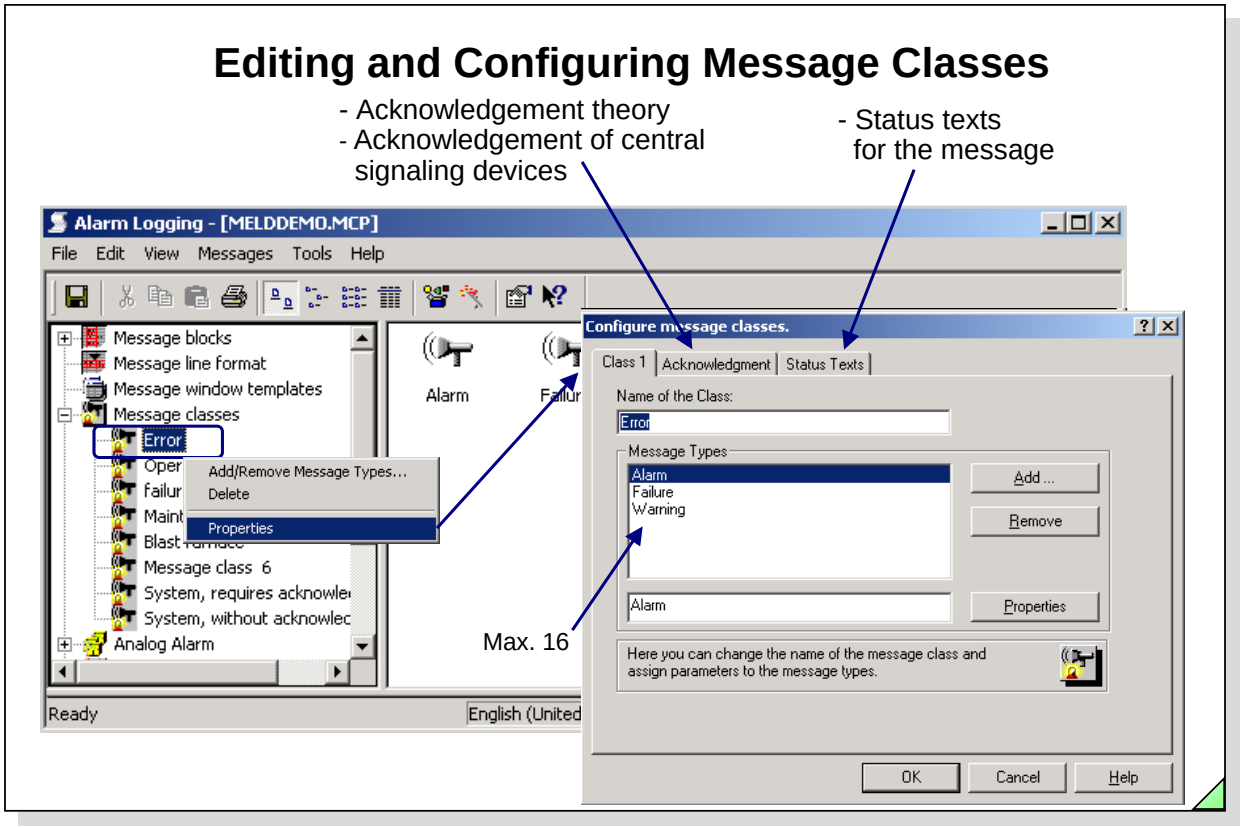
Date: 27.10.2003
File: SWINCC_05E.13

SITRAIN Training for Automation and Drives

General Information A message system is used to signal events that occur sporadically in the process. These events are transmitted chronologically in the form of messages to a central location where they can also be archived. The cause of a message can be either an event or a message (telegram).

Message Class Message classes differ with regard to the acknowledgement philosophy. Messages with the same acknowledgement philosophy can be grouped together in one message class. In Alarm Logging, the message classes "Malfunction", "System, requires acknowledgement" and "System, without acknowledgement requirement" are preconfigured. You can define up to 16 message classes in WinCC.

Message Type Message types are subgroups of message classes and can differ with regard to the type of color of the message states. You can create up to 16 message types for every message class in WinCC.



Each message of a project is located in an assigned storage consisting of 16 message classes with 16 message types each. You can define these message classes and message types.

Editing Message Classes

1. Through the "Message Classes" object, call the shortcut menu.
2. Select the "Add/Remove..." command.
3. Move the message classes from the left part of the window to the right part of the window.
4. Use the "OK" button to save the setting.

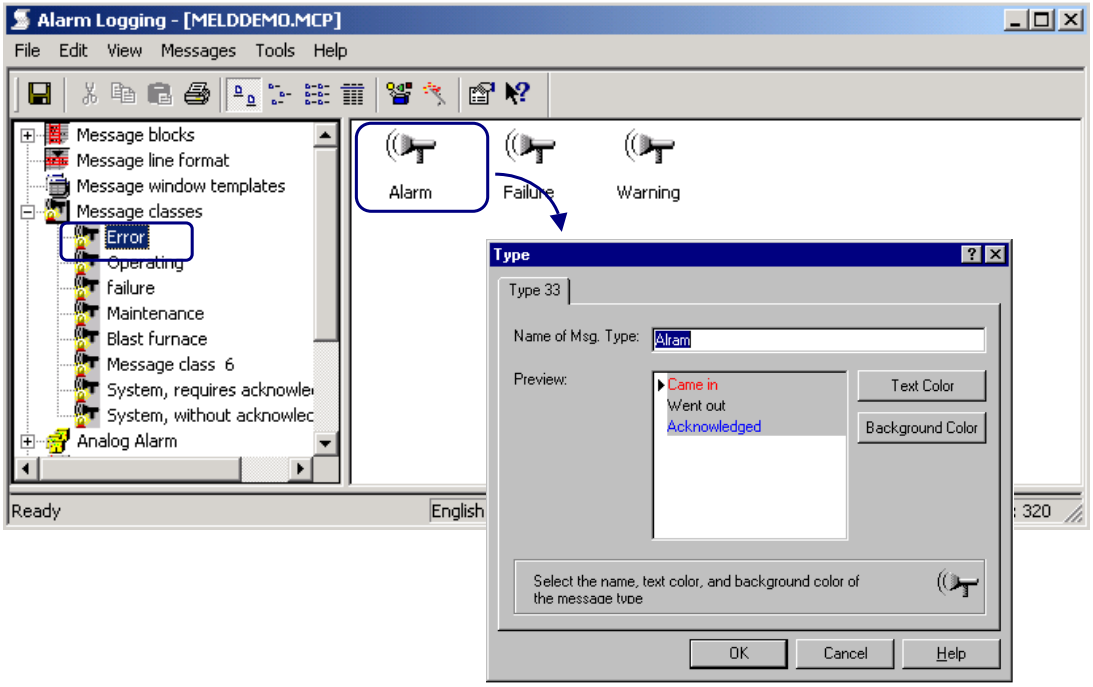
Changing Message Class Properties

1. Call the shortcut menu from the message class object you just created.
2. Select the "Properties" command in the shortcut menu.
   You can change the name (for example, message class 2 -> blast furnace) and insert the message types in the called up window. (Refer to the screen in the slide above.)
3. Use the "OK" button to save the settings.

System Messages

The "System, requires acknowledgement" and "System, without acknowledgement requirement" message classes always exist and you can only change them using the Properties function.
The system messages are generated by the different modules in WinCC.
In the menu bar, click on "Tools", in the drop-down menu, "WinCC - System Messages" to integrate these messages in your message system.

Editing the Message Types



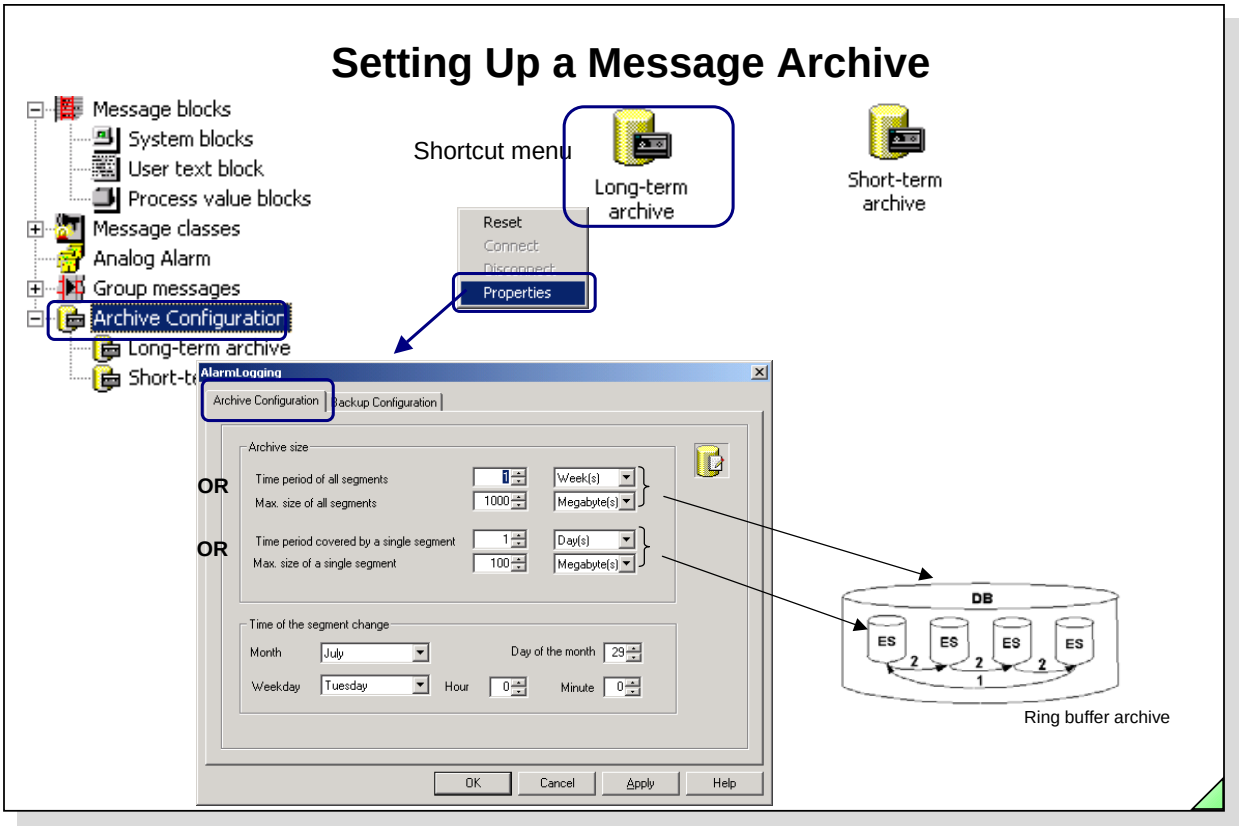
Each message of a project is located in an assigned storage consisting of 16 message classes with 16 message types each. The user can define these message classes and message types.

Editing the Message Types

1. Call the shortcut menu from the message class object you just created.
2. Select the "Add/Remove..." command in the shortcut menu.
3. Move the message types from the left part of the window to the right part of the window.
4. Use the "OK" button to save the setting.

Changing the Message Type Properties

1. Call the shortcut menu from the message type object you just created.
2. Select the "Properties" command in the shortcut menu.
   You can change the name (for example, TYPE 1 -> warning) and the attributes for the status texts in the called up window. (See the screen in the slide above).
3. Use the "OK" button to save the settings.



WinCC >= V6 no longer differentiates between short-term archives and long-term archives. In Alarm Logging, you define for which time period the data are to be archived and as of when the data are to be overwritten.

Note

By default, the backup begins one-quarter hour after the first time-dependent segment change. With every configuration change, a new segment begins in RT.

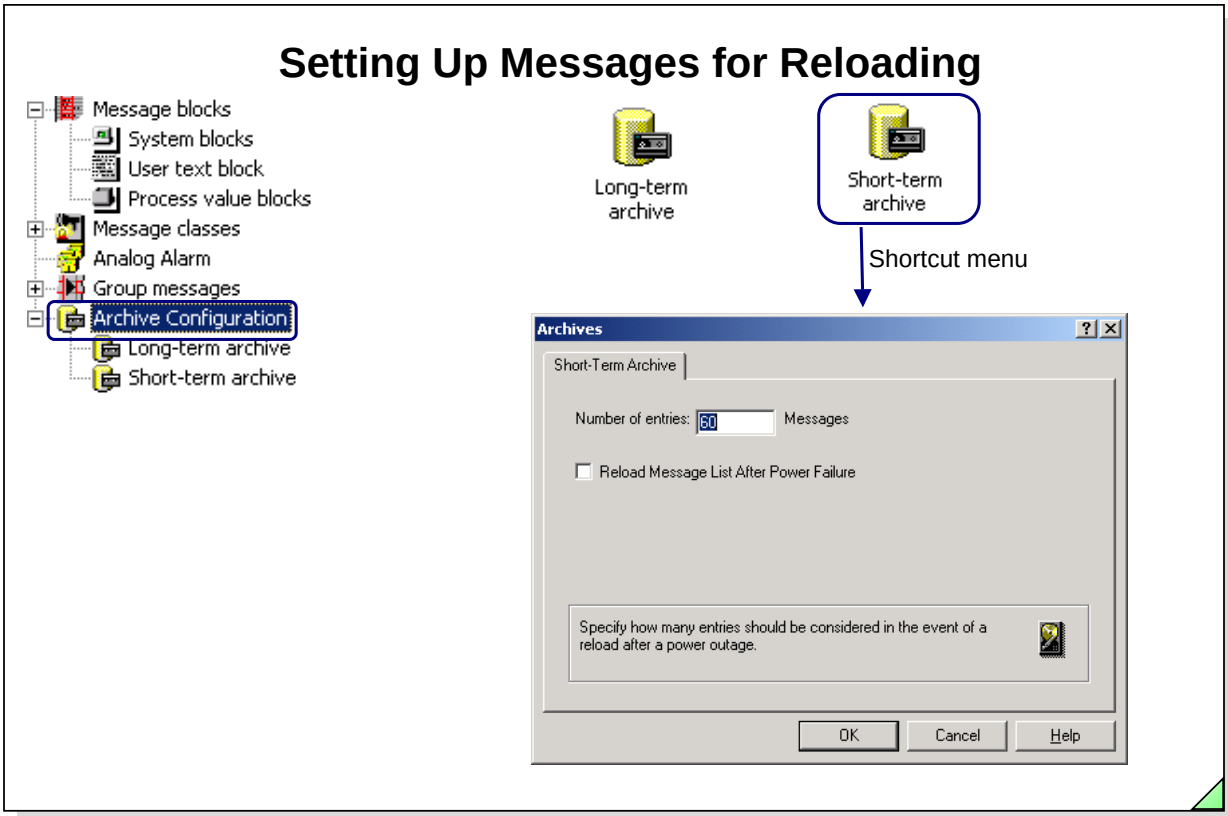
Archive Configuration

The "Time period of all segments" or "Max. size of all segments" refers to the entire archive. If the one or the other limit is exceeded, old segments are deleted.

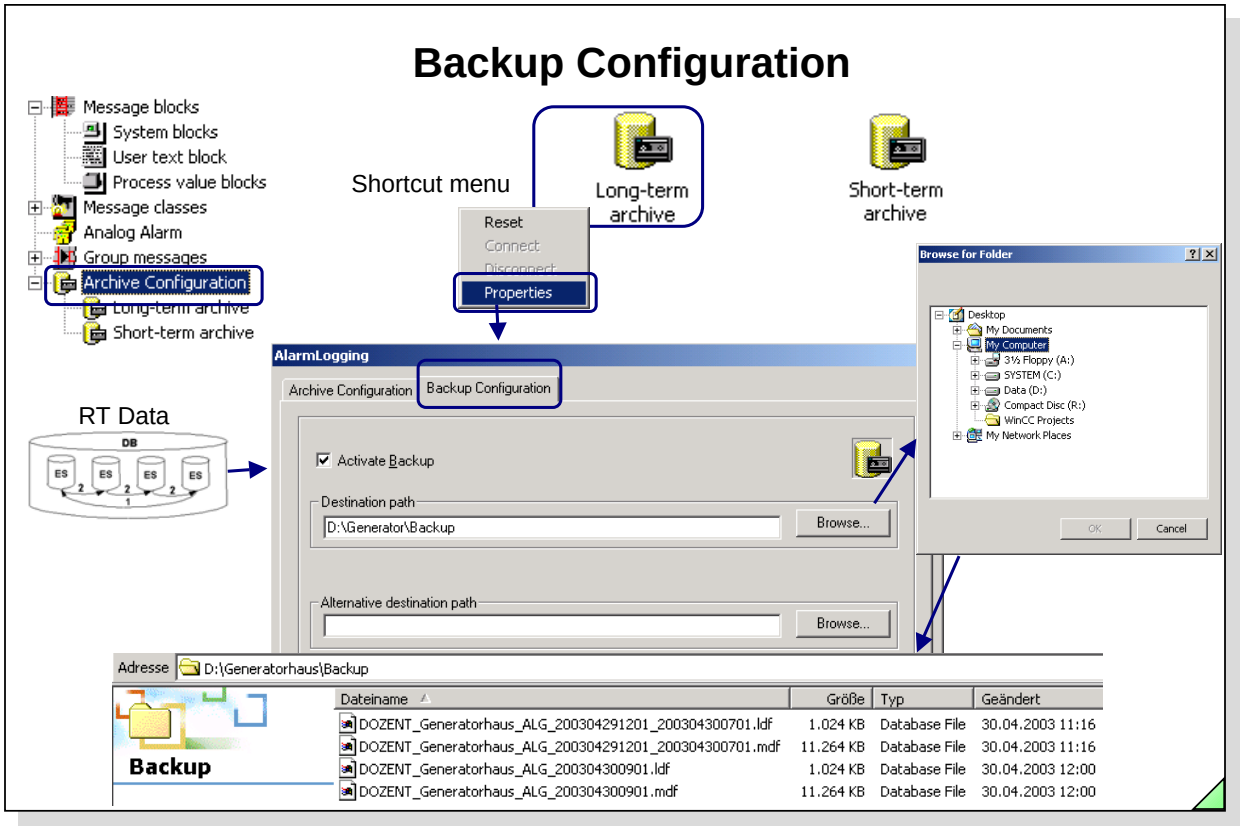
The "Time period for single segments" or "Max. size of a single segment" refers to archive slices. That way, the archive can be divided into several database files.

The "Time of the segment change" : Here you can enter a defined start time, for example for a daily segmentation, every segment is to begin at 0:00 o'clock. If the project is started for the first time at 16:00 o'clock, the first segment goes from 16:00 to 24:00 o'clock. After that, every segment covers the period from 0:00 to 24:00 o'clock.

+smallest unit for the time period of all segments: day
+smallest unit for the size of all segments: 1 megabyte



The short-term archive parameter assignment is defined for reloading messages after a power failure ("Number of entries", like WinCC V5 cannot be defined).



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.18

SITRAIN Training for Automation and Drives

Swap Triggering

+Archive segment is completed (15 minutes after completion) or
+Maximum archive segment size is reached.

Example: RT Data

COMPUTER_PROJECT_ALG_<Time_from>_<Time_to>.LDF
COMPUTER_PROJECT_ALG_<Time_from>_<Time_to>.MDF
DOZENT_Generatorhaus_ALG_200304300901.ldf
DOZENT_Generatorhaus_ALG_200304300901.mdf

Alternative Destination Path

The "Alternative destination path" is used if, for example, the network path for the backup is not available, for example, because of a network failure.

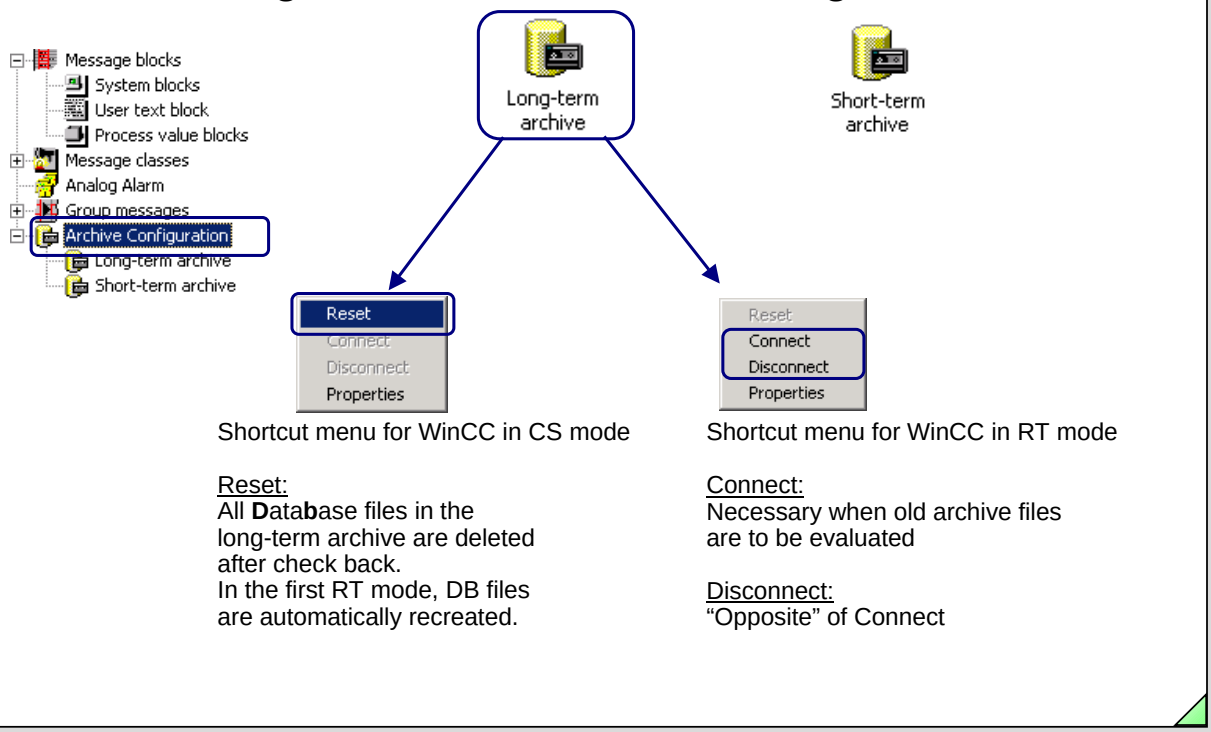
Evaluation

Access to the swapped data:
+such as, with optional WinCC tools
+copying the archive segments back into the associated subdirectory.
Such as: DOZENT_Generatorhaus_ALG_200304300901.mdf and with WinCC "Connect" evaluate old archive.

Delete Backup

To delete an archive backup, delete the associated LDF file and MDF file in the Windows Explorer.

Resetting the Archive and Evaluating Old Archives



Shortcut menu for WinCC in CS mode

Reset:
All **D**ata**b**ase files in the long-term archive are deleted after check back.
In the first RT mode, DB files are automatically recreated.

Shortcut menu for WinCC in RT mode

Connect:
Necessary when old archive files are to be evaluated

Disconnect:
"Opposite" of Connect

Sequence for Connecting

1. Copy backup files to a local drive or insert memory medium (such as, CD).

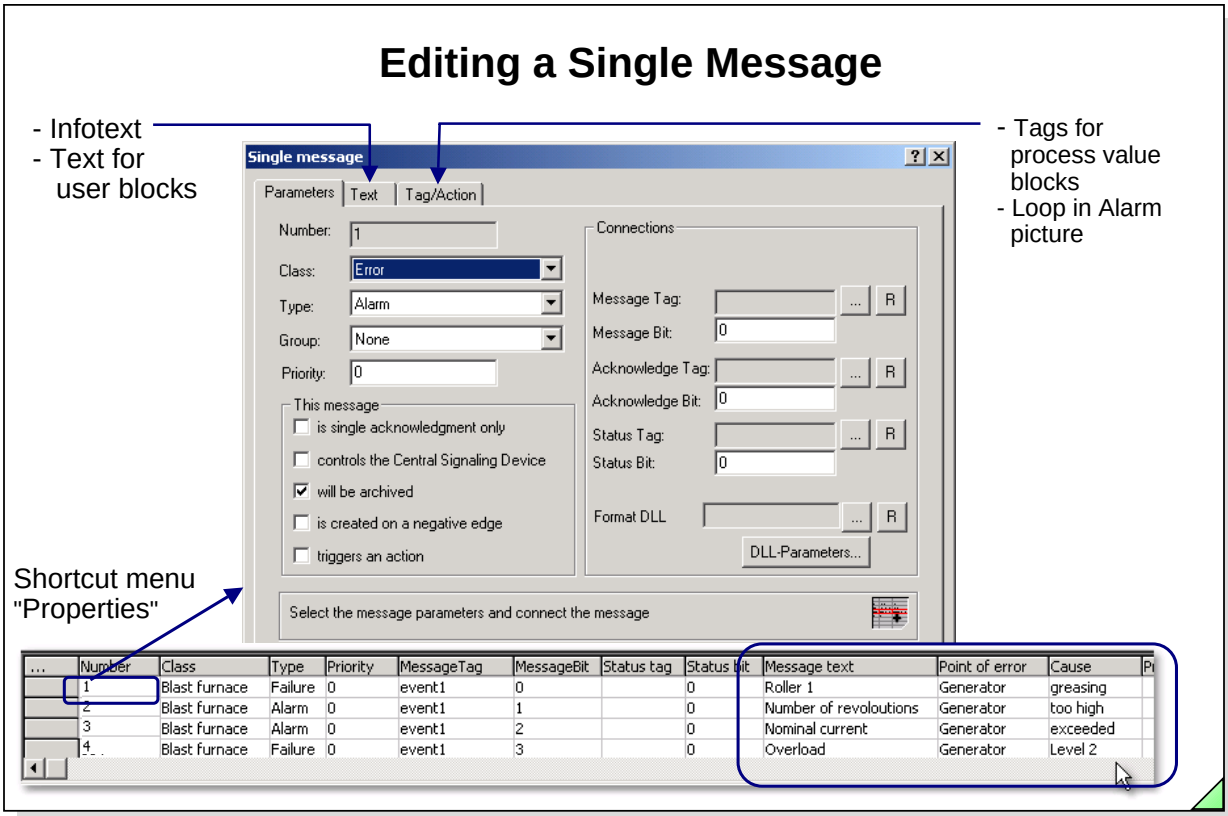
2. Connect databases.

3. Values are immediately available in RT: For Trends, you only have to scroll through the archive.
   For messages, you only have to read in the messages once more (such as, changing from short-term archive into the long-term archive).

4. When you don't need the data any longer, disconnect the database.

5. Data can then be removed from the local drive.

Editing a Single Message



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.20

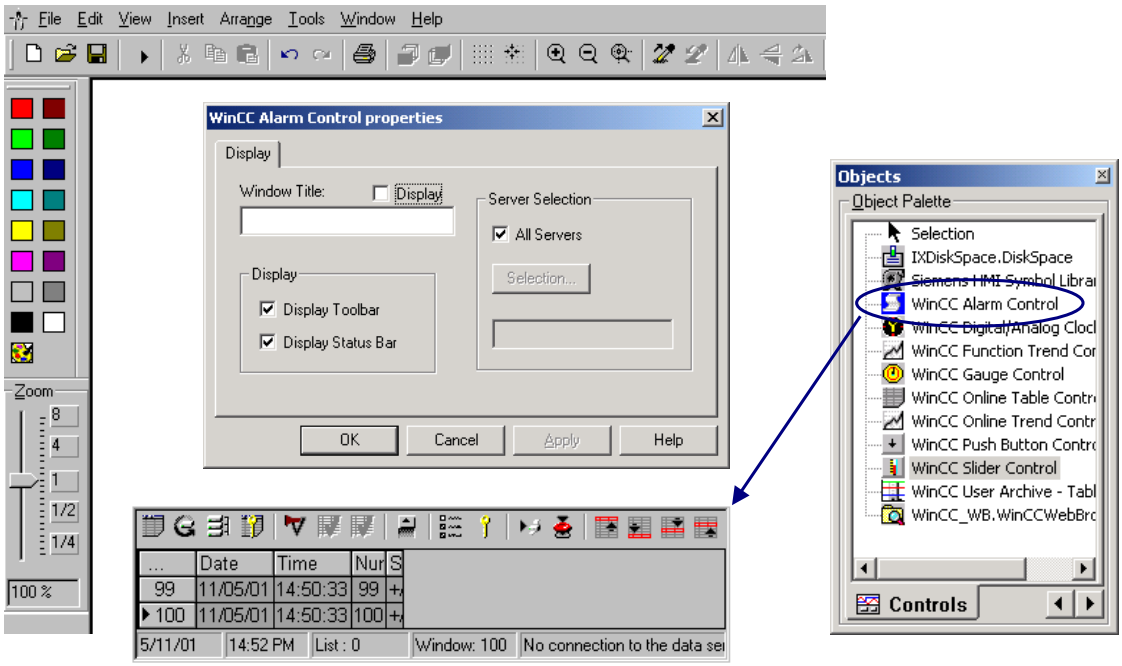
SITRAIN Training for Automation and Drives

You assign a message tag and a message bit to single messages. Single messages can be combined into group messages.

Editing a Single Message

1. Select a single message in the table window.
2. Call the shortcut menu and select the “Properties” instruction.
   You can set the parameters for the single message in the window that is called up. (See screen in the slide above).
3. Use the “OK” button to save.

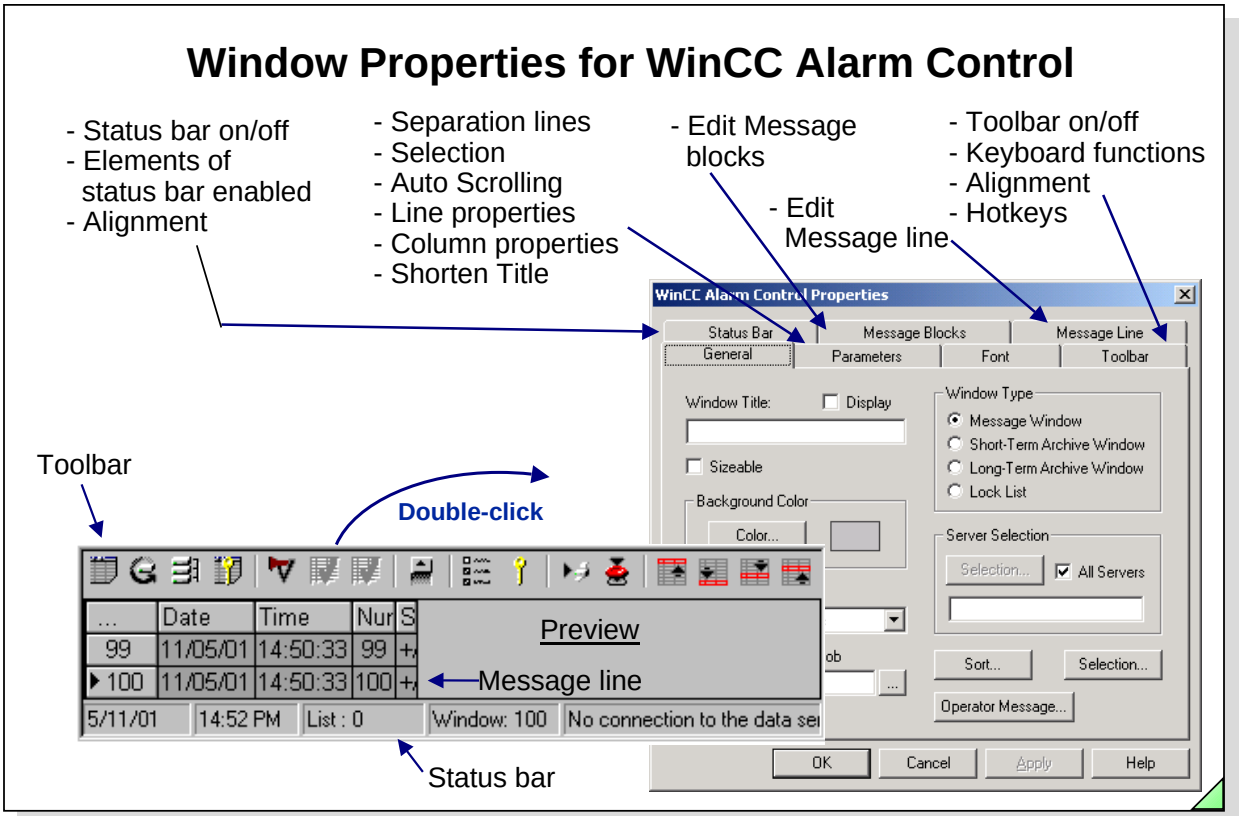
WinCC Alarm Control in Graphics Designer



The WinCC Alarm Control is the runtime window in which messages are displayed during runtime.

After the WinCC Alarm Control has been generated, the design, display and input possibilities of a message window template are assigned parameters.

After the Alarm Control has been edited, the Properties window automatically appears. You can assign additional parameters through the Properties window of the Control (see next page).



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.22

SITRAIN Training for Automation and Drives

With the help of the window properties for the WinCC Alarm Control, you establish the design and the display of a message window.

Call

-You can call the Properties window with a **double-click on the left mouse button**.

Window Preview

-To obtain a window preview, you must first save the picture and then call it up again.

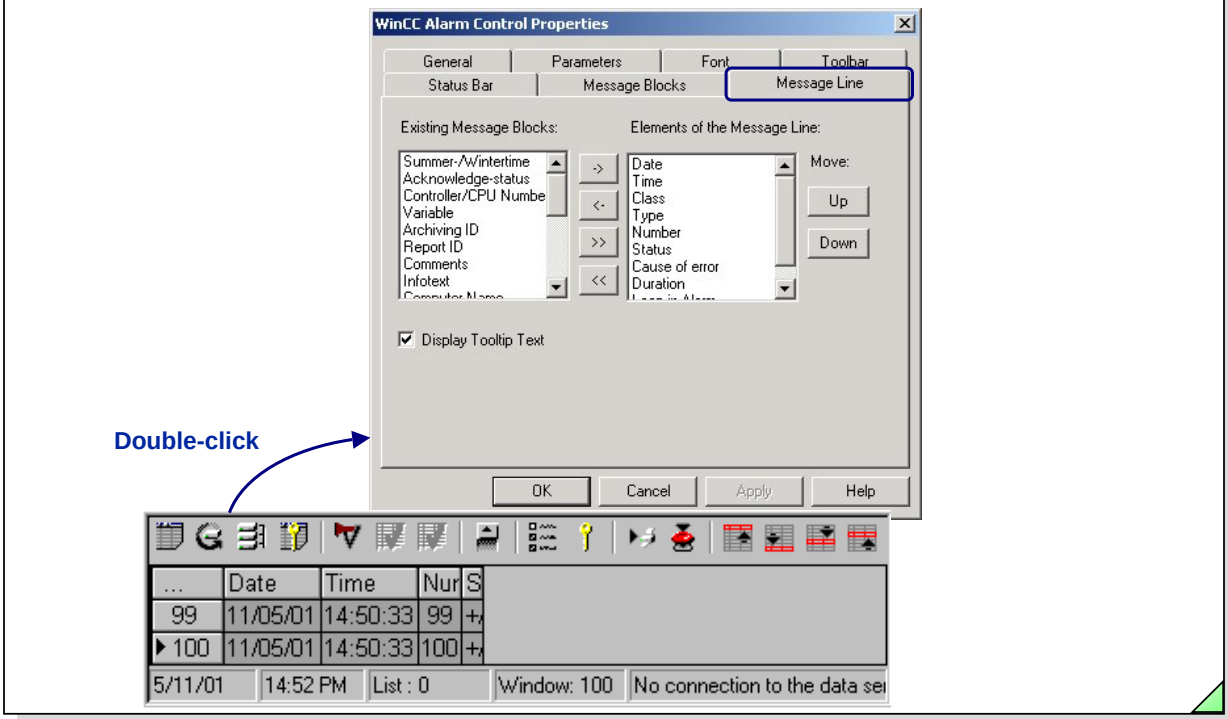
Selection

You can set Filters (according to message number, state, date...) using the Selection button. These filters take effect when the picture is selected.

Operator Messages

Here you define which events are to trigger an operator message. With this you can log the disabling, enabling and acknowledging of messages.

Message Line for WinCC Alarm Control



Through the Message Line tab, you can establish the sequence of the message blocks in a message line (format).

You choose the message blocks that are to be displayed in the message line from the configured message blocks. You can see the sequence in a preview. This sequence can be changed by shifting the blocks.

Message blocks that do not already exist can be edited in through the Message Blocks tab.

Tooltiptext

A longer message text from the message blocks in a line is displayed as a multi-line tooltip text (>=V5.1).

Note

For various views of message archive contents, you have to repeat the configuration displayed in the screen using the WinCC Alarm Control window.

Configuration Notes

Bit Message Procedure	"Reporting in the Correct Chronological Order"
- WinCC polls the message bits every 500ms	+ message telegrams are only sent when events occur
+ easy to configure	+ exact time-stamp for messages (length of the CPU cycle; e.g. 50ms)
- "central acknowledgement" through acknowledgement tags	- with PMC "central acknowledgement" without additional work
	- S5-PMC: higher capital expenditure (PMC channel; PMCPRO and special FBs)
	- S5-PMC: higher CPU scan cycle time and higher memory usage
	+ Advantage with S5 communication: S5-PMC does 'Blocking'
	+ S7 blocks integrated in the operating system

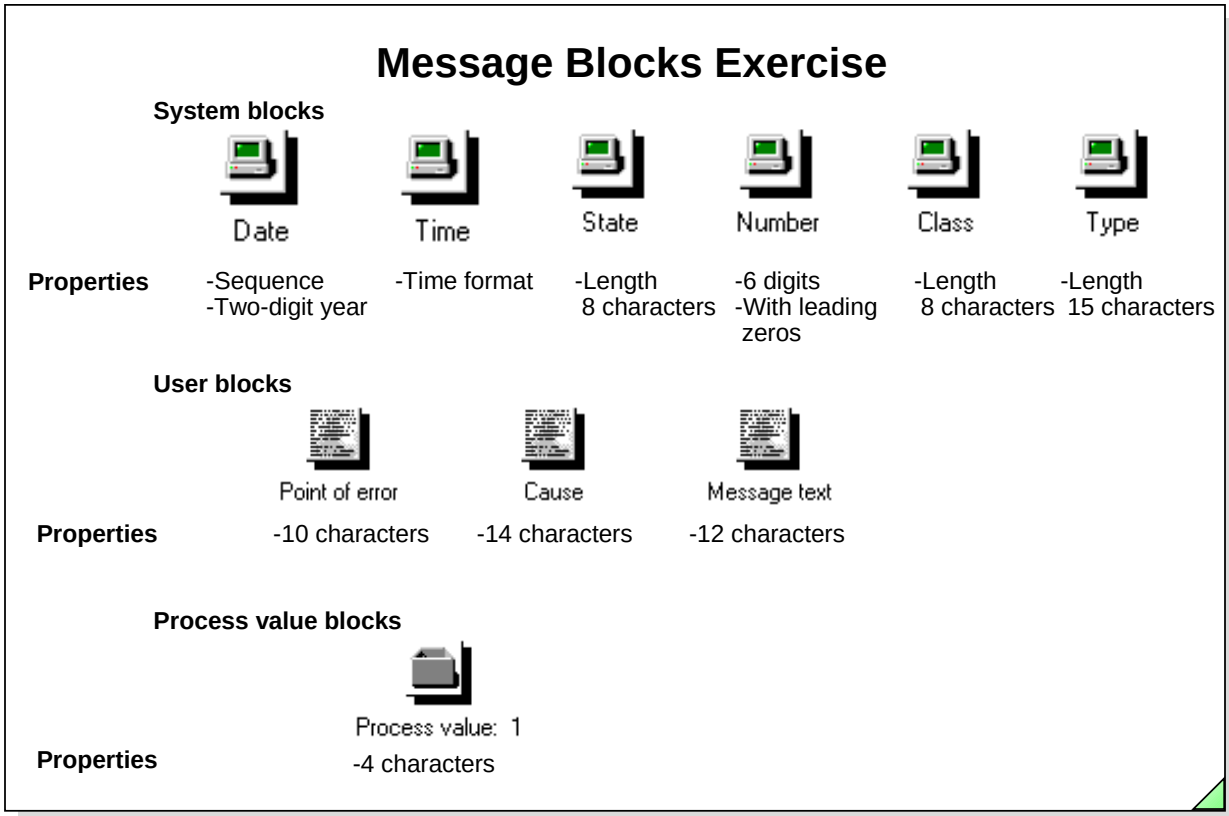
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.24

SITRAIN Training for Automation and Drives

Data Type You should use the largest possible data type for the bit message procedure. That way, you can reduce the amount of data for the transmission.

Analog Alarms If analog alarms are configured in WinCC, then these values are also requested cyclically.



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.25

SITRAIN Training for Automation and Drives

Exercise

Create the message blocks specified in the slide above.

Set the properties specified in the slide above.

Message Class, Message Types, and Archives Exercise

Configuring message classes

Blast furnace

General
- Name: blast furnace
- Message types
  Alarm, error

Acknowledgement
-Acknowledgement came in

Status texts
- ARR
- DEP
- ACK
- A/D

Message types

Alarm

Failure

Properties
- Arrived: red
- Departed: green
- Acknowledged: yellow

Archive

Short-term archive

-100 messages for reloading

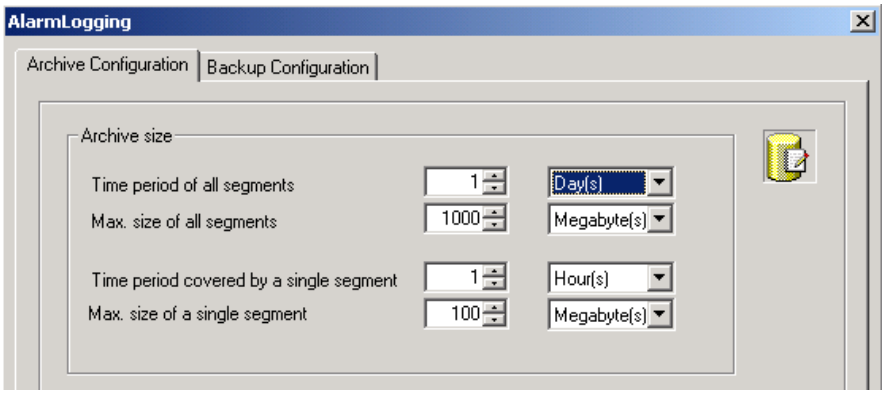
Long-term archive

Properties

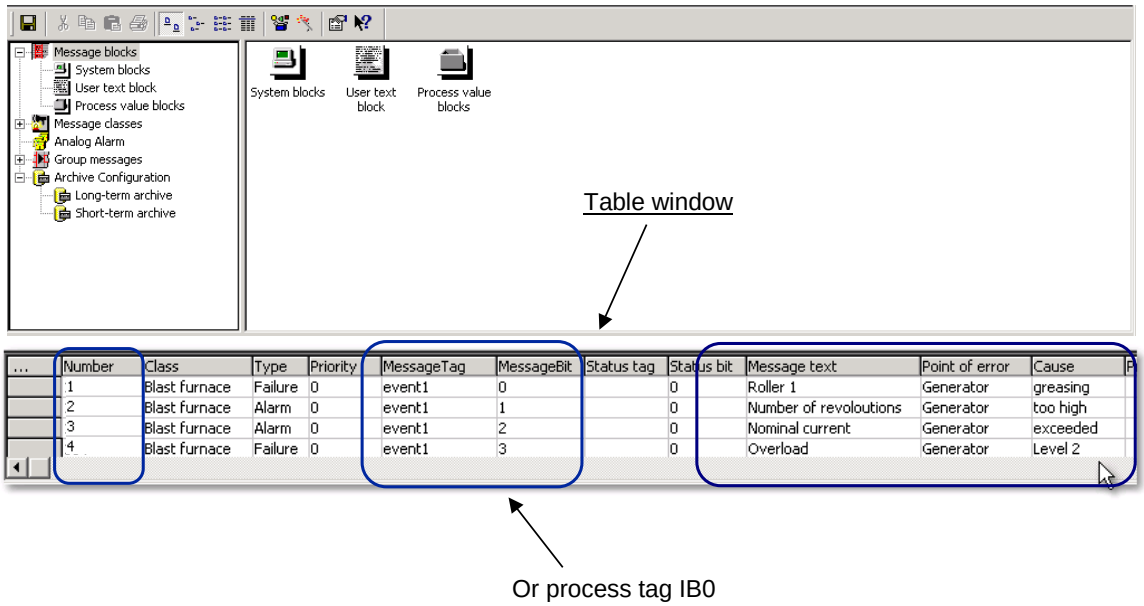
-See screenshot in text

Exercise

1. Create the message classes specified in the slide above. Adjust the specified values.
2. Create the message types specified in the slide above. Transfer the specified values.
3. Transfer the long-term parameters specified in the screenshot below.



Single Message Exercise



...	Number	Class	Type	Priority	MessageTag	MessageBit	Status tag	Status bit	Message text	Point of error	Cause
	1	Blast furnace	Failure	0	event1	0		0	Roller 1	Generator	greasing
	2	Blast furnace	Alarm	0	event1	1		0	Number of revoloutions	Generator	too high
	3	Blast furnace	Alarm	0	event1	2		0	Nominal current	Generator	exceeded
	4	Blast furnace	Failure	0	event1	3		0	Overload	Generator	Level 2

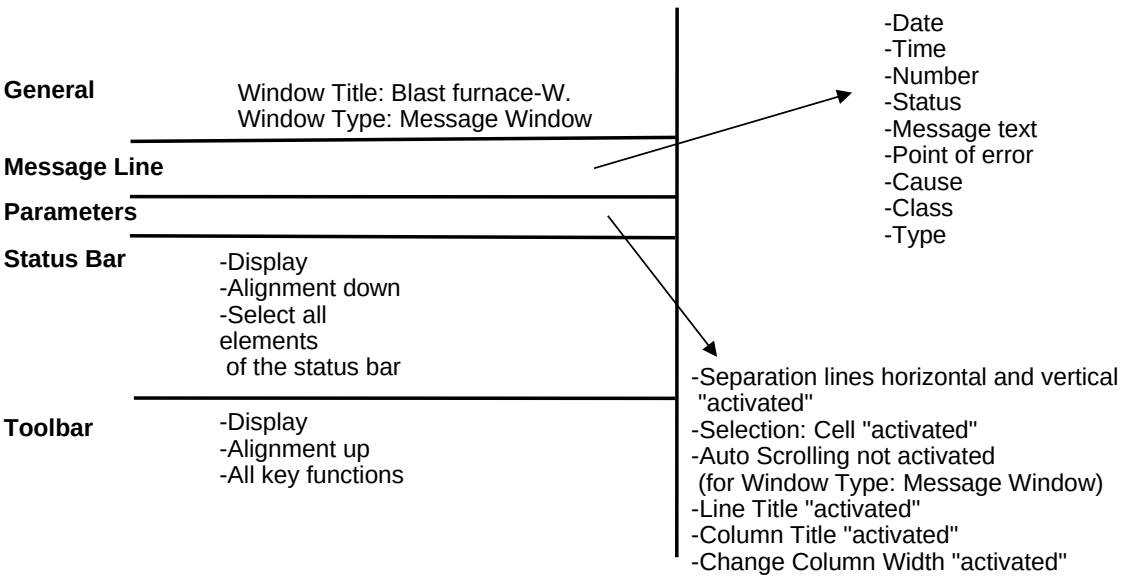
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.27

SITRAIN Training for Automation and Drives

Exercise Create four single messages with the numbers 1 to 4. Take the parameters specified in the table window in the slide above into consideration.

WinCC Alarm Control Exercise



Copy (if present) the picture melden.pdl into your project from the catalog ab_uebung.

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.28

SITRAIN Training for Automation and Drives

Objective Create a message picture that is called from the Start picture.

Exercise

- From the catalog ab_uebung, copy the picture Melden.pdl into your project with the subdirectory *GraCS* .

- In the Melden.pdl graphic screen generate a WinCC Alarm Control window.

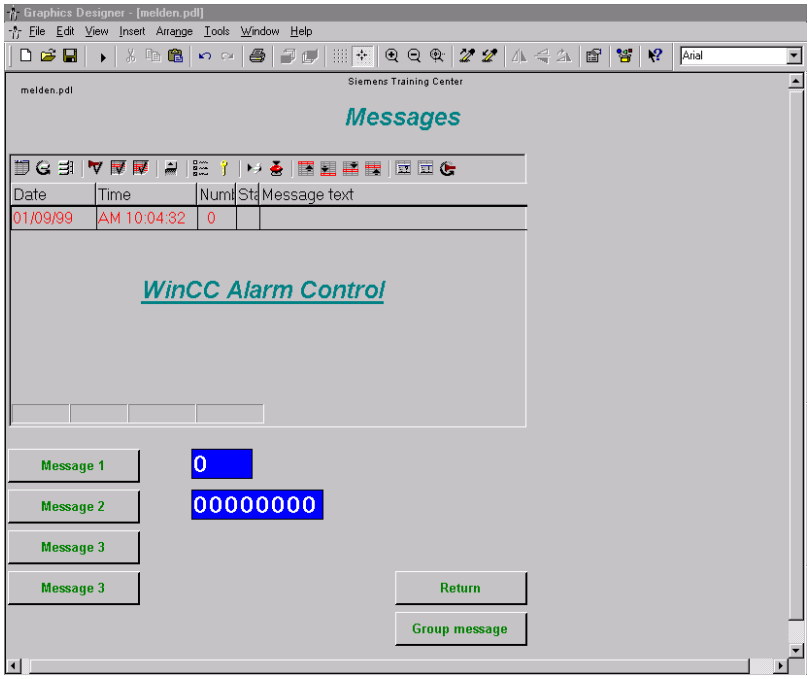
Note The properties of the WinCC Alarm Control window can be optionally changed.

Exercise

- Use the parameters given the the slide above for the WinCC Alarm Control window.

- Create the message line given in the slide above and set up the message blocks given.

Message Screen Exercise



In the start up list 1.) Alarm Logging RunTime and 2.) Text Library Runtime must be selected.

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.29

SITRAIN Training for Automation and Drives

Message Screen Function Test

Carry out a function test on the configuration you created. Test the attributes you assigned.

Exercise 1

Create a message with process value blocks and display the message. (Adapt the message line.)

Exercise 2

Create a message with a LOOP IN ALARM screen.

Exercise 3

Exporting and importing a message:

1. Export a message.
2. Use a text editor (such as Excel) to copy this message under a new message number.
3. Import this text file.

Exercise 4

Create a message screen with a fixed selection on the message class Blast Furnace.

Note

In the start up list 1.) Alarm Logging Runtime and 2.) Text Library Runtime must be selected.

Group Messages

1) From message classes:

Message class A | Message class B .... | Message class P

Message type 1 ....message type 16 .... Mess. type 1 .... mess. type 16

2) From user-defined messages:

Message numbers x to (x+2n) ..... Message numbers y to (y+2m)

Messages ....messages x to x+n .... x to x+2n

Messages ....messages y to y+m .... y to y+2m

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.30

SITRAIN Training for Automation and Drives

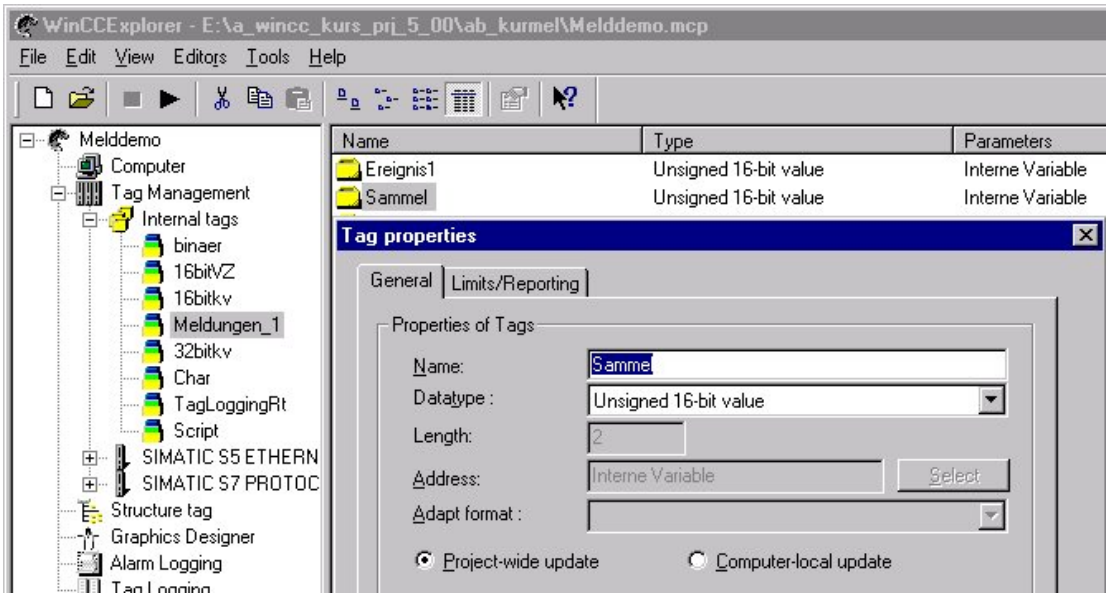
General

Group messages are used to collect events (OR-ing) and their displays. There are two different ways of creating group messages.

1) From message classes
2) From user-defined messages

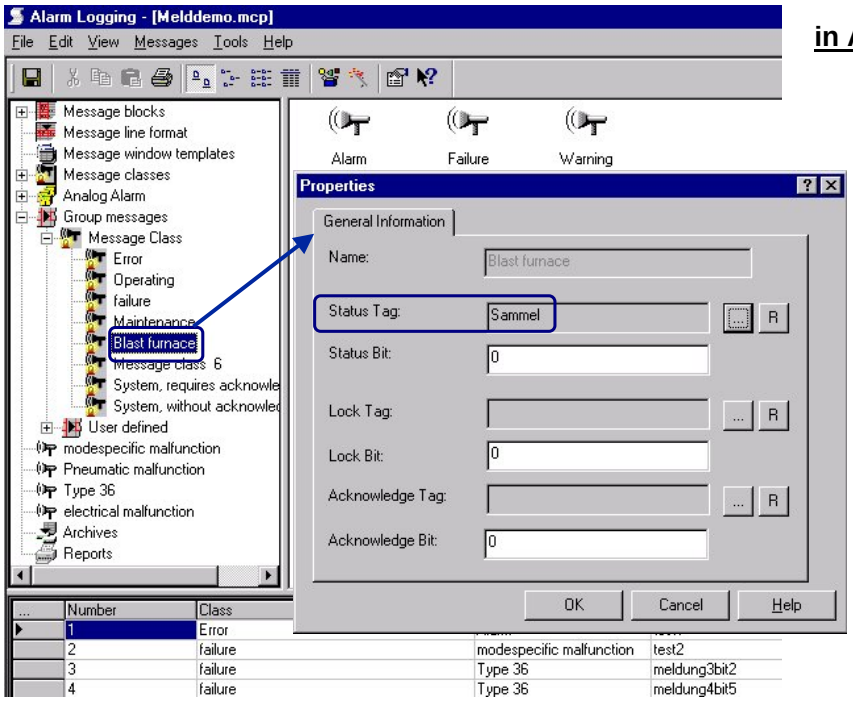
A tag that specifies the status of the group message must be defined for each group message.

Creating the Tag for a Group Message



Note The tag **'Sammel'** has to be of the data type "**Unsigned 16-bit value"** (8-bits / 16-bits / 32-bits).

Connect Tag with the Group Message (Message Class)



in Alarm - Logging

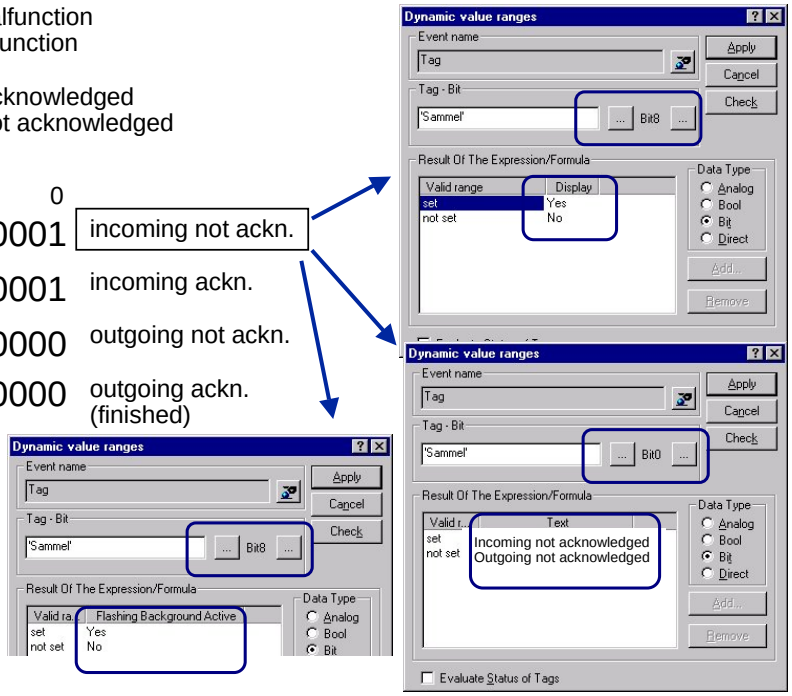
Note The tag ‘Sammel’ is to be created in the ‘Signed 8 bit value’ data type.

Evaluating the Status Tag for a Text Object

Bit 0: 0- there is no malfunction
1- there is a malfunction

Bit 8: 0- message is acknowledged
1- message is not acknowledged

15... 8... 0	
00000001 00000001	incoming not ackn.
00000000 00000001	incoming ackn.
00000001 00000000	outgoing not ackn.
00000000 00000000	outgoing ackn. (finished)



Evaluatation

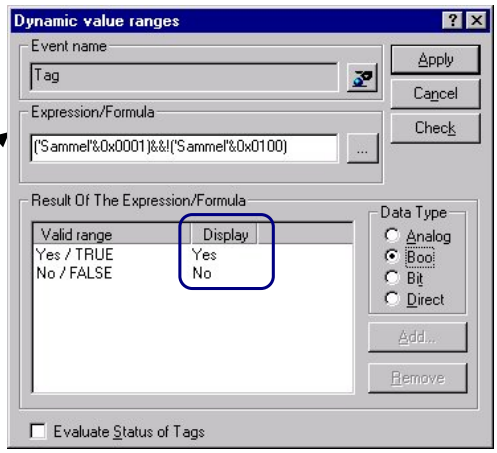
You use the Dynamic Dialog for the static text field “incoming not acknowledged“ to evaluate the Status Tag. The bit status of the tag is thus linked to the associated attribute.

Evaluating the Status Tag for a Text Object

Bit 0: 0- there is no malfunction
1- there is a malfunction

Bit 8: 0- message is acknowledged
1- message is not acknowledged

15... 8... 0	
00000001 00000001	incoming not ackn.
00000000 00000001	incoming ackn.
00000001 00000000	outgoing not ackn.
00000000 00000000	outgoing ackn. (finished)



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.34

SITRAIN Training for Automation and Drives

Evaluation

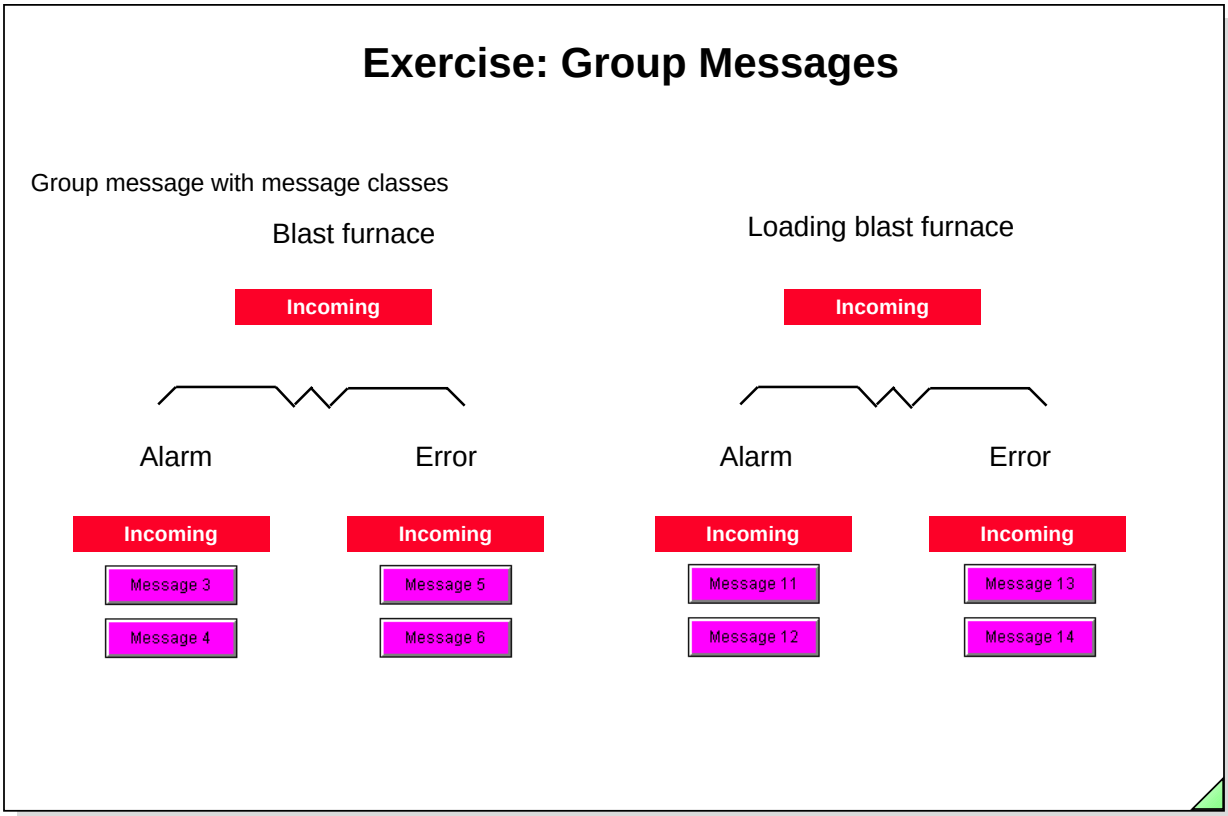
You use the Dynamic Dialog for the static text field “incoming not acknowledged“ to evaluate the Status Tag. Several bit statuses of a tag are thus linked to contants to form an expression.
The result of the expression is linked to an attribute.

Expression

('Sammel'& 0x0001)&& !('Sammel'& 0x0100)

& Binary AND
&& Logical AND

Exercise: Group Messages



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_05E.35

SITRAIN Training for Automation and Drives

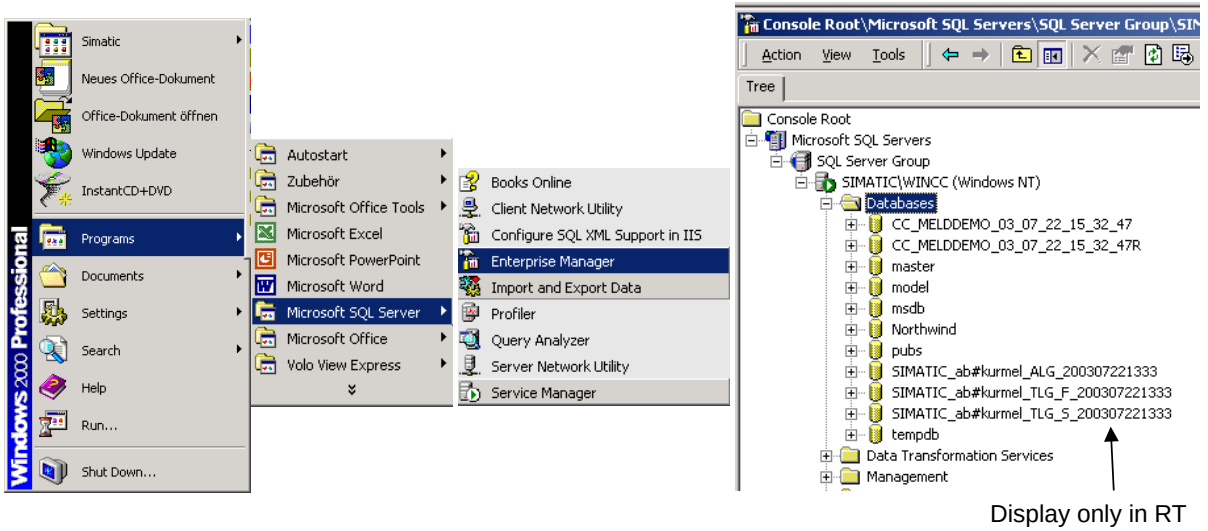
Exercise 5

You are to configure group messages for the message class ‘Blast Furnace’.

- From the Catalog ab_uebung, copy the picture Melden.pdl into your project with the subdirectory *GraCS*.
- Create the tags required for the group messages.

Carry out a function test on the configuration you created. Test the functionality of the exercises.

New: Database Tools 1/2

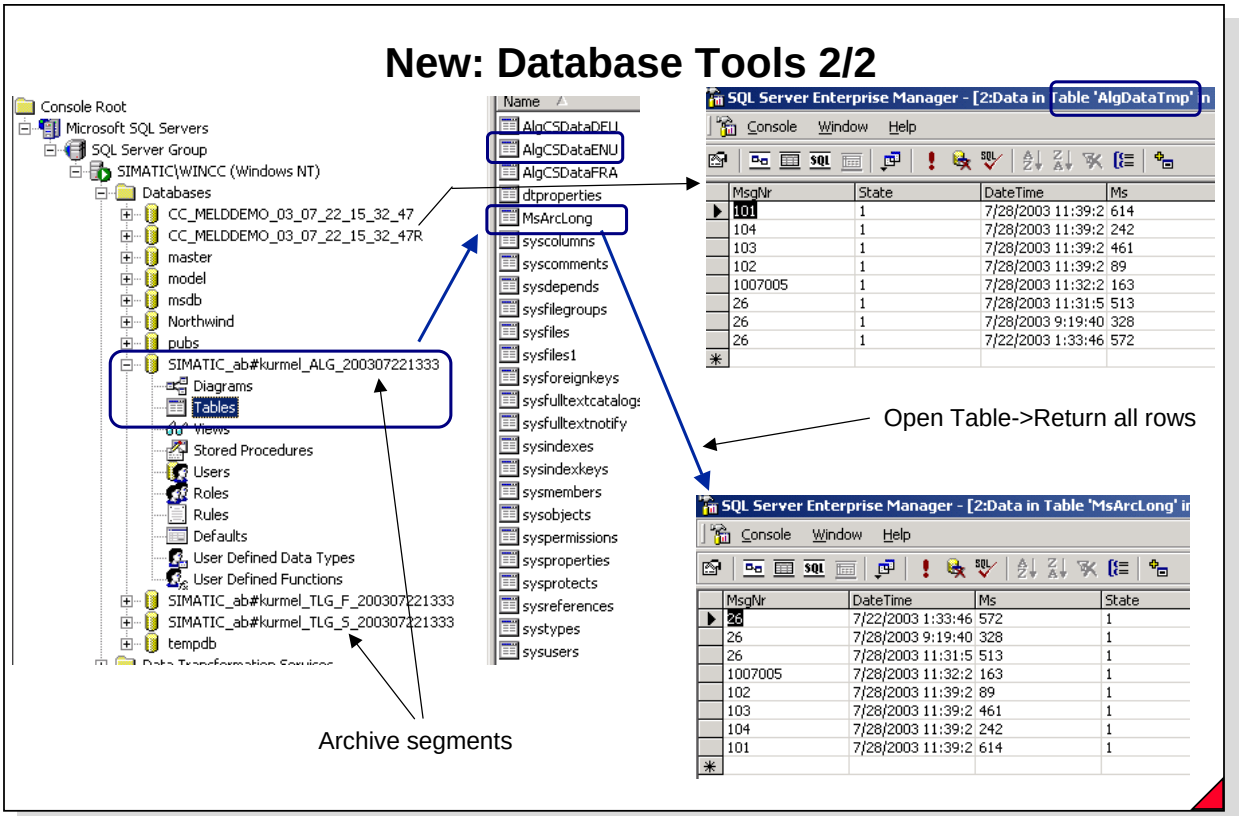


Date: 27.10.2003
File: SWINCC_05E.36

Delivery The Enterprise Edition contains all necessary functions for supporting company-wide data processing.

Optional Other editions: Standard Edition; Personal E.; Developer E.

New: Database Tools 2/2



Tables

AlgCSDataENU here you find the configured messages.

AlgDataTmp here you find the current messages.

MSArcLong here you find the archived messages from the single segments.

Note

You cannot make any changes to the database, because the database can become inconsistent.

Trend Display and Measured Value Archiving

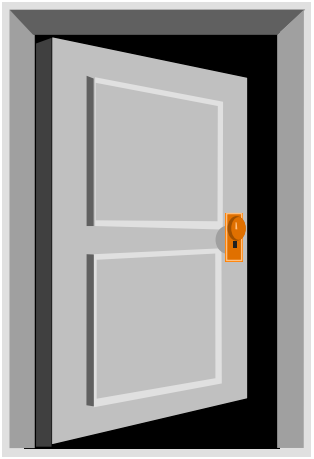
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_06E.1

SITRAIN Training for Automation and Drives

Contents Page

Objectives



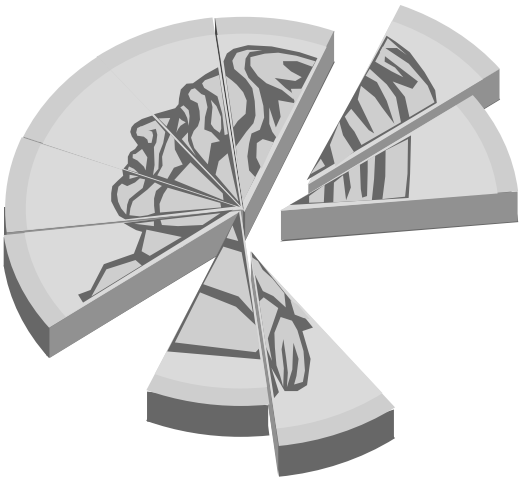
- Measurement Point
- Trend Display
- Trend Archive

Date: 27.10.2003
File: SWINCC_06E.2

⇨ **Upon completion of this chapter, the participant should be able to:**

- know the advantages of tag logging
- start tag logging
- adapt tag logging to the system requirements
- configure measurement points
- display measurement points in a picture as trends
- archive measurement points

Configuration Preparation



- Archive Description
- Archiving Time Period
- Trend Analysis
- Displaying Measured Values

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_06E.3

SITRAIN Training for Automation and Drives

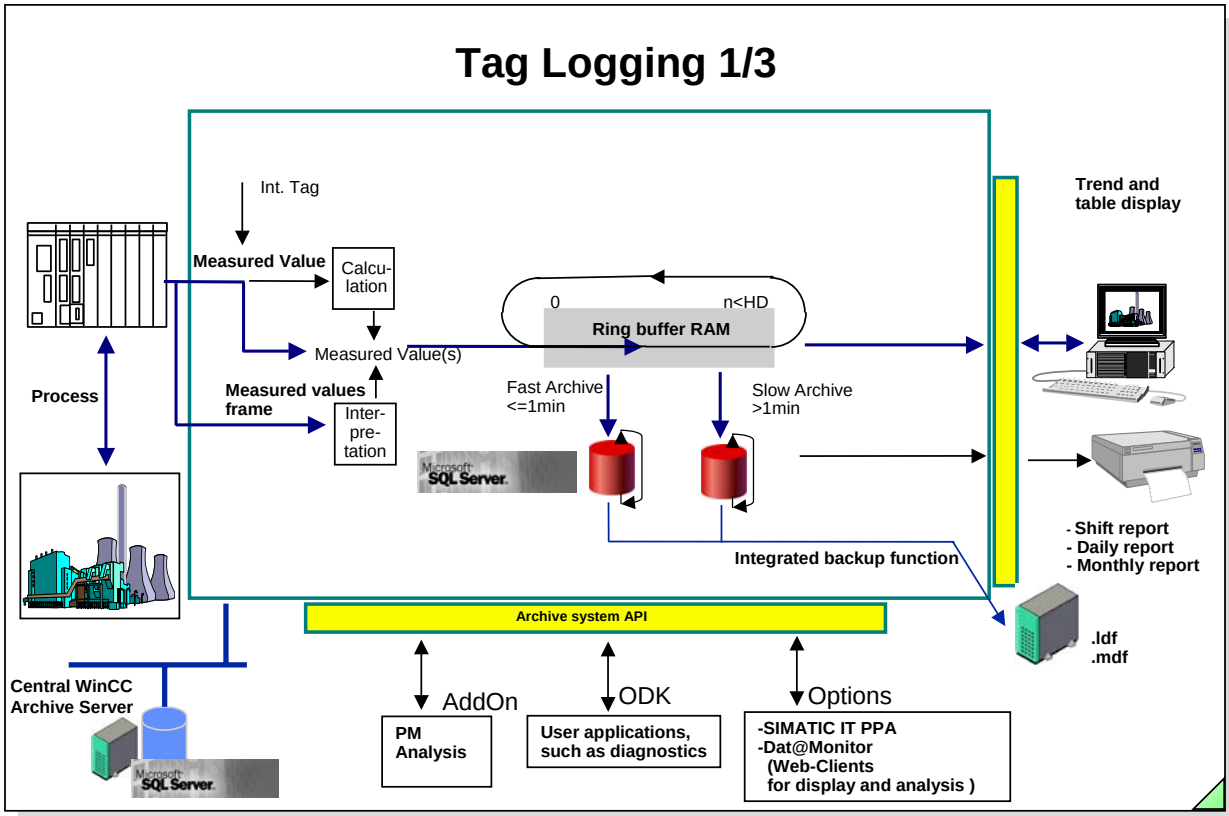
⇨ Configuration Preparation

❒ Process Connection

- Defining tags with the assignment of a signed 16-bit value

❒ Acquiring Measured Values

- Grouping all measurement points
- Specifying the archive description
- Specifying the archiving time period
- Specifying the analysis and evaluation routines
- Specifying the reports
- Specifying the process values
- Defining the contents and display for the trend window
- Specifying the trend display



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_06E.4

SITRAIN Training for Automation and Drives

General Information

By means of tag logging, tags from the PLC are displayed in WinCC in the form of trends and tables. The tags are archived and reported, if applicable.

To view the history, the measured values are saved on the hard drive in a ring buffer archive.

Tag Logging

- Measured value and measured value message frame
- Main memory (Real Time Trends)
- Ring buffer archive
- Opening for analysis programs and evaluation programs

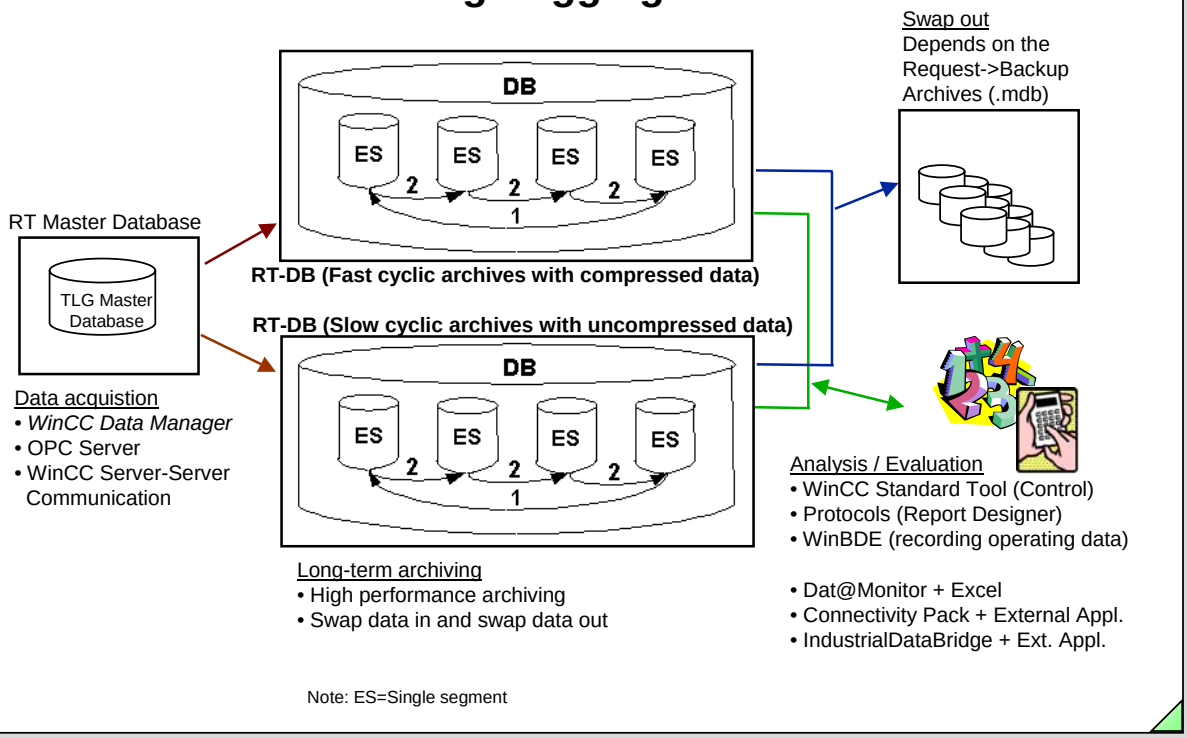
Number of Measured Values with Fast Archive

Archiving to the database for server/single-user of up to 5000 values/second.

License

The WinCC basic software contains 512 available archive tags. For extensive configurations, an expansion using powerpacks is needed.

Tag Logging 2/3



Archiving

For archiving measured values, WinCC uses cyclic archives (ring buffer) of a configurable size. You can configure them with or without backup.
The archive files are always stored on the local computer in the associated project.
The WinCC measured values archive consists of multiple single segments. You can configure both the size/time of the measured values archive and the size/time of the single segments in WinCC.
Example: The measured values archive archives all measured values that occur within one week (1), each single segment archives the measured values that occur within one day (2).
You can always configure both conditions at the same time. If either of the two criteria (time or size) is exceeded, the following happens:
Criteria for measured values archive (DB) is exceeded (1)-> the oldest measured values (that is, the oldest single segment) are deleted.
Criteria for single segment is exceeded (2)-> a new single segment (ES) is created.
A new single segment is also created when you configure TagLogging data online (for load Delta online).

Dat@Monitor

For visualization and evaluation, WinCC/Dat@Monitor provides a series of Internet-capable tools for viewing (view only) and online analysis. These support all the usual security mechanisms such as login/password, firewalls, encrypting etc..

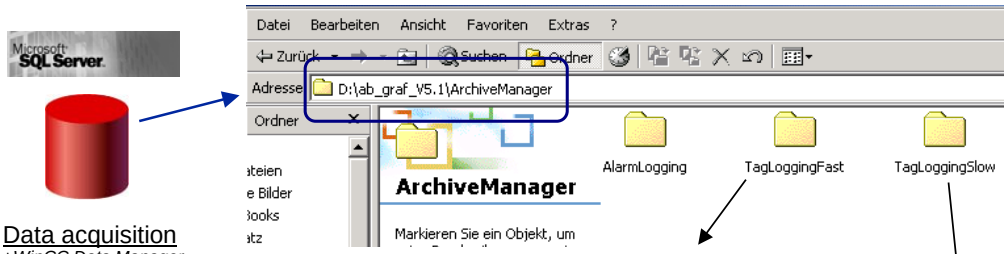
Connectivity Pack

Access to WinCC using OPC and OLE-DB. Using OLE-DB, you can directly access the archive data stored by WinCC in the MS SQL Server database. The OPC HDA 1.0 (Historical Data Access) and OPC A&E 1.0 (Alarm & Events) servers enable you to access historical WinCC archive system data and to pass on / acknowledge messages.

IndustrialDataBridge

Configurable connection to databases and IT systems.

Tag Logging 3/3



Data acquisition
+*WinCC Data Manager*
+OPC Server
+WinCC Server-Server Communication(max.11)

Runtime Data Tag Logging Fast Segment:
COMPUTER_PROJECT_TLG_Fast_<Time_from>_<Time_to>.LDF
COMPUTER_PROJECT_TLG_Fast_<Time_from>_<Time_to>.MDF
COMPUTER_PROJECT_TLG_Fast_YYYYMMDDHHMM.MDF
Example: DOZENT_ab#graf#V5.1_TLG_F_200306020629.mdf
DOZENT_ab#graf#V5.1_TLG_F_200306020901_200306021000.mdf

Runtime Data Tag Logging Slow Segment:
COMPUTER_PROJECT_TLG_Slow_<Time_from>_<Time_to>.LDF
COMPUTER_PROJECT_TLG_Slow_<Time_from>_<Time_to>.MDF
COMPUTER_PROJECT_TLG_Slow_YYYYMMDDHHMM.MDF
Example: DOZENT_ab#graf#V5.1_TLG_S_200306020629.mdf
DOZENT_ab#graf#V5.1_TLG_S_200306020901_200306021000.mdf

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_06E.6

SITRAIN Training for Automation and Drives

Name RT-DB

The runtime data are distributed amongst the master database and several runtime databases.

The master database manages the runtime databases and refers to the individual runtime databases. The master database is created in the project directory. The name of the master database consists of the project name with a trailing "RT".

The runtime databases each contain the archived data for a specific time period and are stored in the project directory in a subdirectory "ArchiveManager/TagLoggingFast or TagLoggingSlow ".

The name of the runtime database consists of:
+ computer name
+ project name
+ ID for the type of data (ALG = Alarm Logging, TLG_F = Tag Logging Fast, TLG_S = Tag Logging Slow)
+ date/time (year, month, day, hours, minutes).

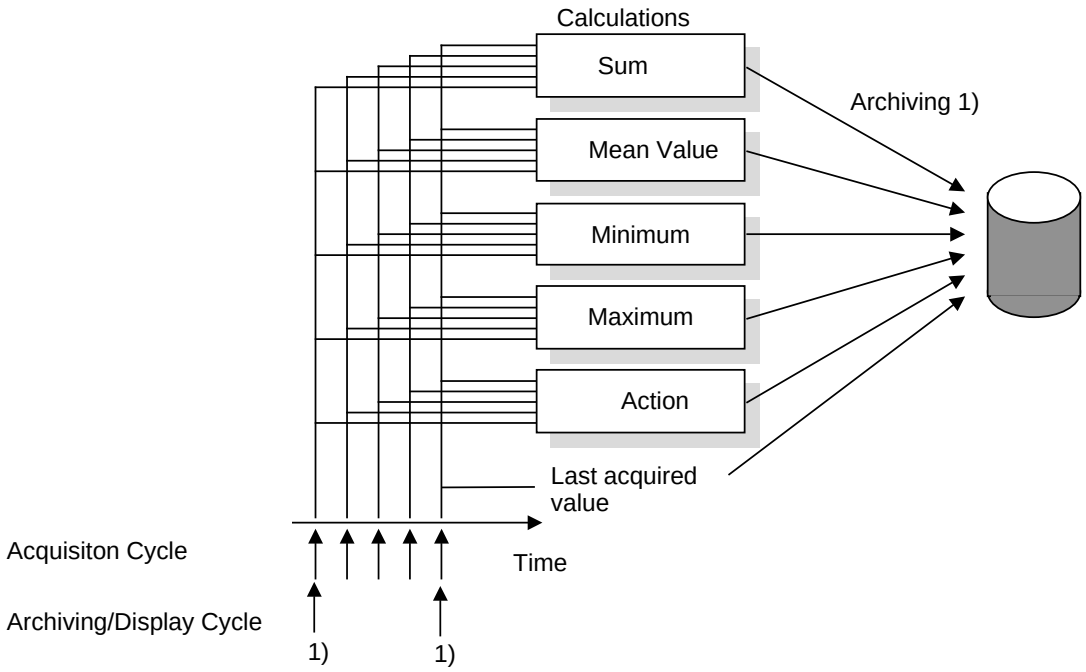
If the time of the segment change is less than the RT Start time, an additional file is created.
For example:
DOZENT_ab#graf#V5.1_TLG_S_200306021001_200306021101.mdf.
This file structure also indicates that measured values were recorded in this time period.
If the file only contains the time of the segment change, then no measured values were recorded in this time period.

Note

During the migration from V5.1->V6.0, the data of the original ring buffer (cyclic) archives and the sequence archives are stored in a common database.

Ldf=transaction Log File
mdf=file with the user data

Measured Value Archiving and Processing



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_06E.7

SITRAIN Training for Automation and Drives

General Information

The WinCC system records measured values in an acquisition cycle, and the values are saved during the archiving cycle to the hard drive. You have to make sure that only the values are written to the archiving cycle. Values recorded in the acquisition cycle are used only for the calculations named above, but they are not saved in the archive.

Processing

WinCC Tag Logging provides you with the following measured value processing:

- Sum generation
- Mean Value generation
- Minimum generation
- Maximum generation
- Action (free arithmetic)

Types of Archiving

WinCC Tag Logging provides you with the following types of archiving for measured values:

- Acyclic, with Start/Stop Events (controlled by the return value)
In acyclic archiving, a tag / measured value is stored when a start event is received or when the value of the tag changes.

- Cyclic - selective, with Start/Stop Events
Archiving is started dependent on an event being received and continues in constant time cycles until a second event is received. When there is a Stop Signal, the last acquired value is also stored

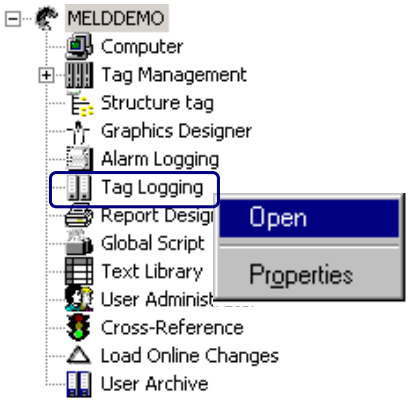
- Cyclic - continuous
Data acquisition begins when the system is started (Runtime) and continues in constant time cycles until the system is switched off

- Only upon change

FAQ 71278

FAQ 71278 describes the meaning and functionality of the archiving types

Tag Logging Call-Up



1. **Call the shortcut menu.**
2. **Use the mouse to click on “Open.”**

   Result: Tag Logging starts.
3. **Maximize the window and set up the interface as you want it.**

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_06E.8

SITRAIN Training for Automation and Drives

Task Definition

You are to configure the tag logging with measurement points, the process value archive, and trend window templates.

The configuration must then undergo a function test.

The settings necessary for the exercise are shown in the configuration explanation.

Requirements

If no tags have been configured as yet, then the tags must be configured now before additional configurations are carried out. (See Chapter 3 for configuration instructions.)

Tag Logging Call

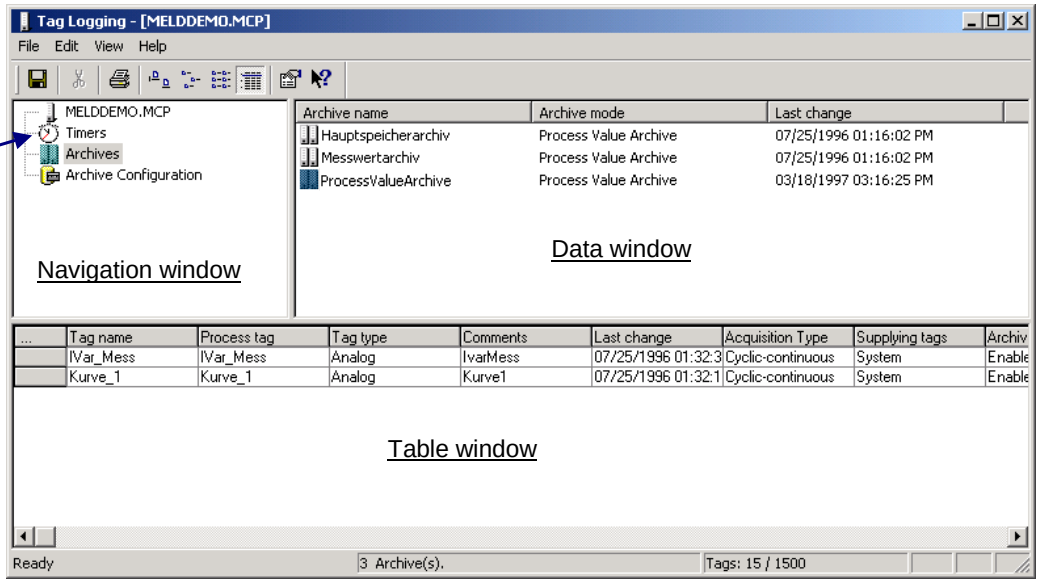
Refer to the screen in the slide above.

Exiting Tag Logging

1. In the **menu bar**, click on **“File.”**
2. In the **drop-down menu**, click on **“Exit.”**

If tag logging was open, you made changes, and you have not saved the changes yet, a dialog box for saving appears after the “Exit” command.

Tag Logging Configuration Interface



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_06E.9

SITRAIN Training for Automation and Drives

Configuration Interface

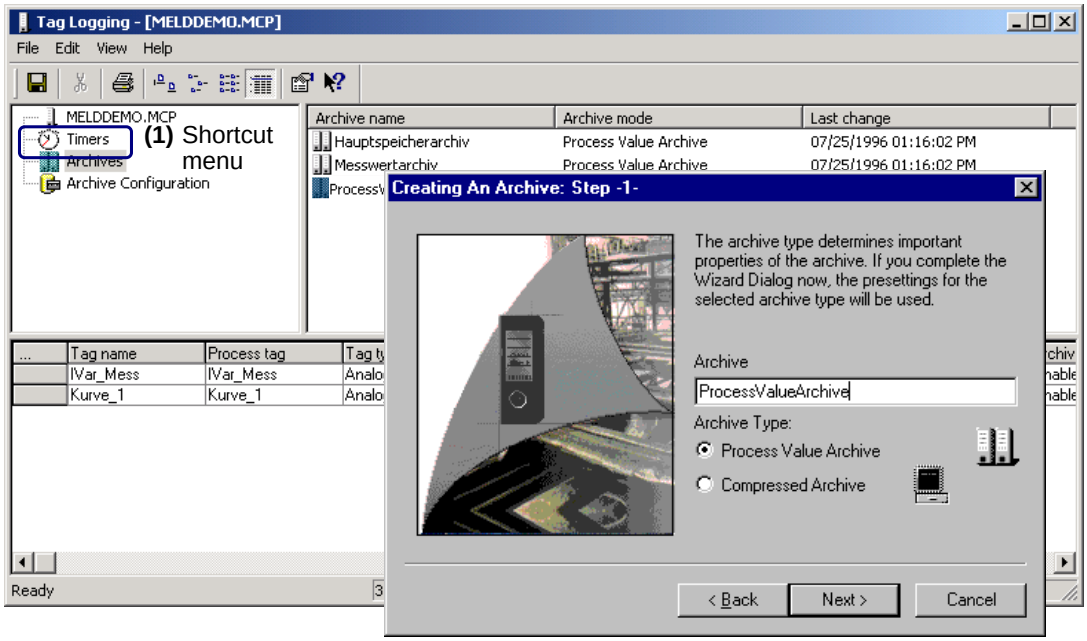
After you call tag logging, the division illustrated in the slide above is displayed on the monitor.

- The editors are displayed in a tree structure in the left window.
- The result of the configuration of the respectively selected editor is displayed in the right window.
- The table window in the lower part of the screen in the slide above displays the editing possibilities for texts, tags, and attributes for the measurement point.

Timers

(1) Shortcut menu, see slide above
With this you can freely create archiving and recording cycles.

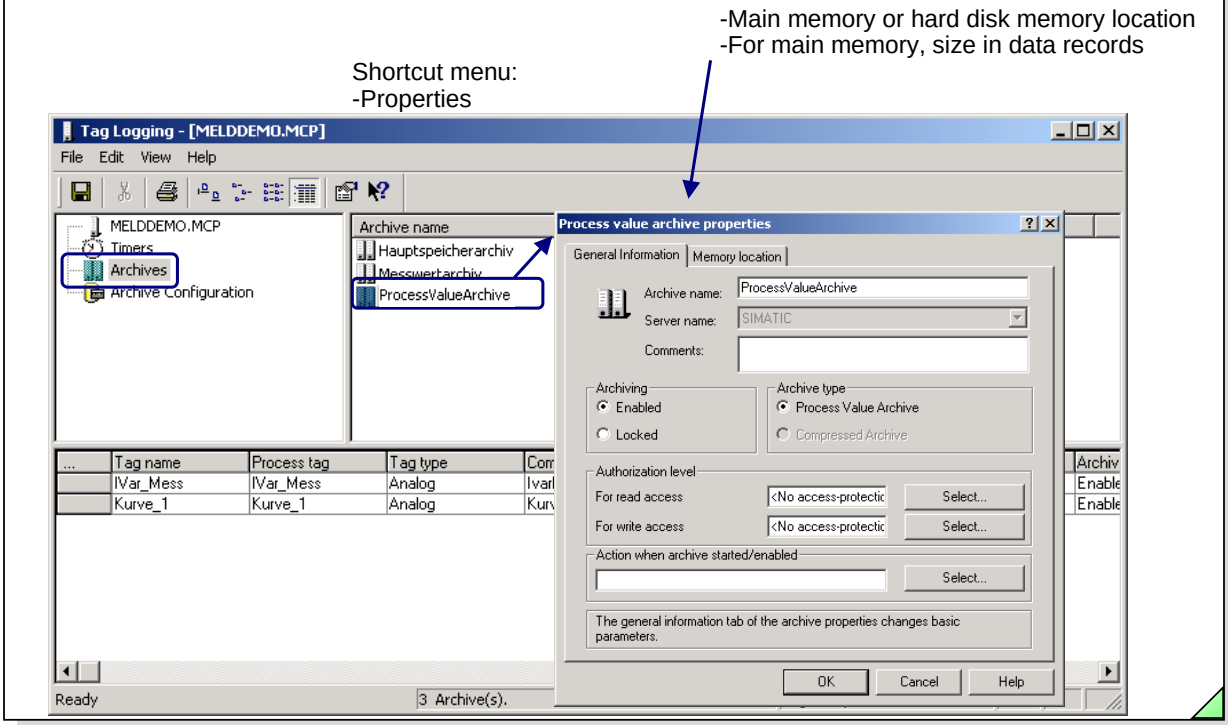
Archive Wizard



Start Archive Wizard **(1) Shortcut menu, see slide above**

To make it easier for you to begin configuring, the measured value system has a configuration assistant (Archive Wizard) that takes all important settings into consideration to set up an executable archive quickly. You can use the archive editor at any time to change the defaults.

Process Value Archive Properties



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_06E.11

SITRAIN Training for Automation and Drives

Archive Location

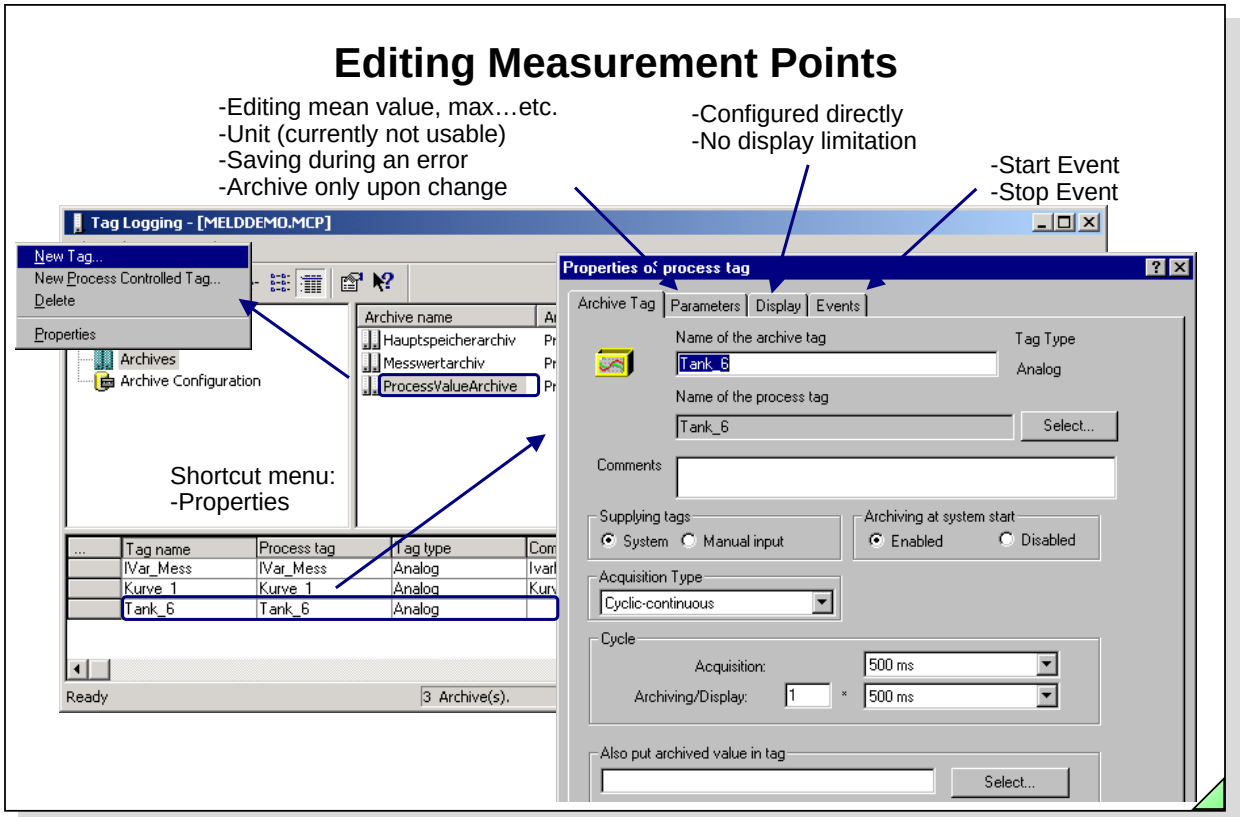
Every measurement point can be archived by the Tag Logging in two different archive types.

-in a main memory (RAM) - Ring Buffer.

-in a local harddisk - Ring Buffer.

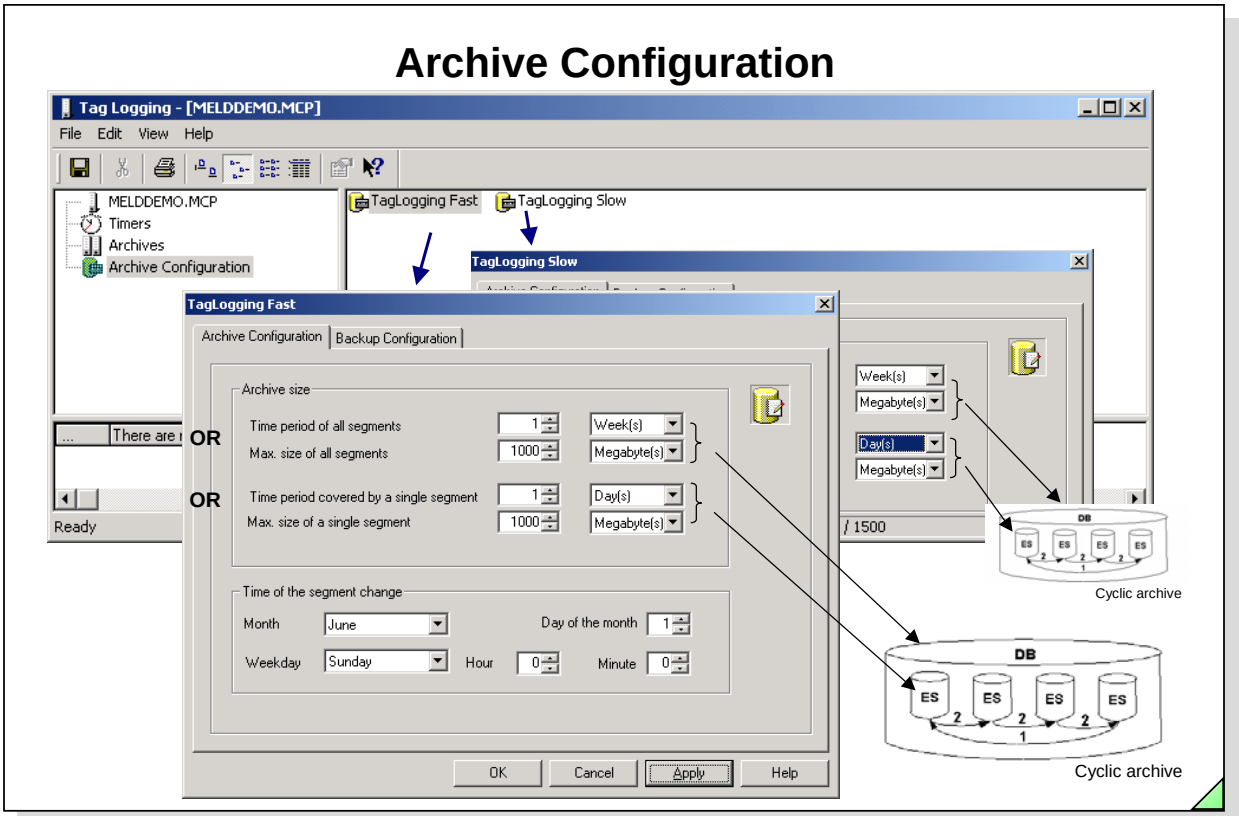
Calculation

The number of measured values multiplied by 28 bytes.



Measurement Points Measurement points can be added, changed, or removed from the archive. You can see the procedure when you look at the selection dialog in the screen displayed in the slide above.

Archive Configuration



Date: 27.10.2003
File: SWINCC_06E.13

WinCC >= V6 no longer differentiates between cyclic (ring buffer) archives and long-term archives. In Tag Logging, you define for which time period the data are to be archived and as of when the data are to be overwritten.
Depending on the archiving cycle, you create a fast or/and slow archive.

Note

By default, the backup begins one-quarter hour after the first time-dependent segment change. With every configuration change, a new segment begins in RT.

Archive Configuration

The "Time period of all segments" or "Max. size of all segments" refers to the entire archive. If the one or the other limit is exceeded, old segments are deleted (or moved to the Backup directory).

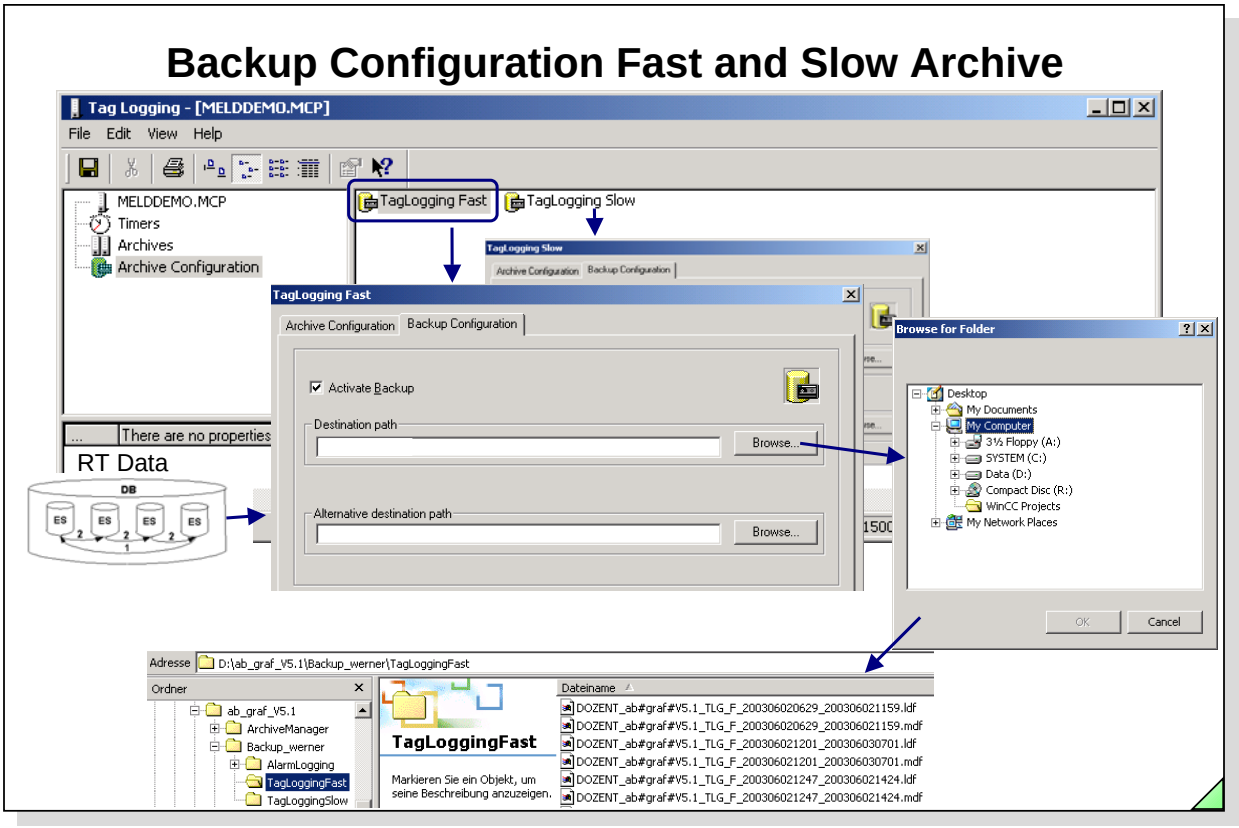
The "Time period for single segments" or "Max. size of a single segment" refers to archive slices. That way, the archive can be divided into several database files.

The "Time of the segment change" : Here you can enter a defined start time, for example for a daily segmentation, every segment is to begin at 0:00 o'clock. If the project is started for the first time at 16:00 o'clock, the first segment goes from 16:00 to 24:00 o'clock. After that, every segment covers the period from 0:00 to 24:00 o'clock.

+smallest unit for the time period of all segments: day
+smallest unit for the size of all segments: 1 megabyte

The archive configuration settings are used by all archives.

Backup Configuration Fast and Slow Archive



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_06E.14

SITRAIN Training for Automation and Drives

Swap Triggering

+Archive segment is completed (15 minutes after completion) or
+Maximum archive segment size is reached.

Example: RT Data

DOZENT_ab#graf#V5.1_TLG_S_200306021247_200306021254.mdf
DOZENT_ab#graf#V5.1_TLG_S_200306021301.mdf

DOZENT_ab#graf#V5.1_TLG_F_200306021247.mdf

Alternative Destination Path

The "Alternative destination path" is used if, for example, the network path for the backup is not available, for example, because of a network failure.

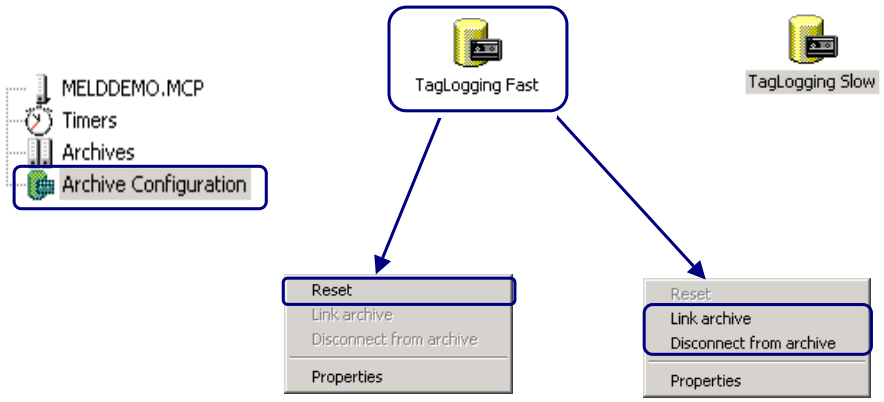
Evaluation

Access to the swapped data:
+such as, with optional WinCC tools
+selecting the archive segments in the associated subdirectory.
Such as: DOZENT_ab#graf#V5.1_TLG_S_200306021247_200306021254.mdf
and with WinCC "Connect" evaluate old archive.

Delete Backup

To delete an archive backup, delete the associated LDF file and MDF file in the Windows Explorer.

Resetting the Archive and Evaluating Old Archives



Shortcut menu for WinCC in CS mode

Reset:
All **D**ata**b**ase files in the long-term archive are deleted after check back.
In the first RT mode, DB files are automatically recreated.

Shortcut menu for WinCC in RT mode

Connect:
Necessary when old archive files are to be evaluated

Disconnect:
"Opposite" of Connect

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_06E.15

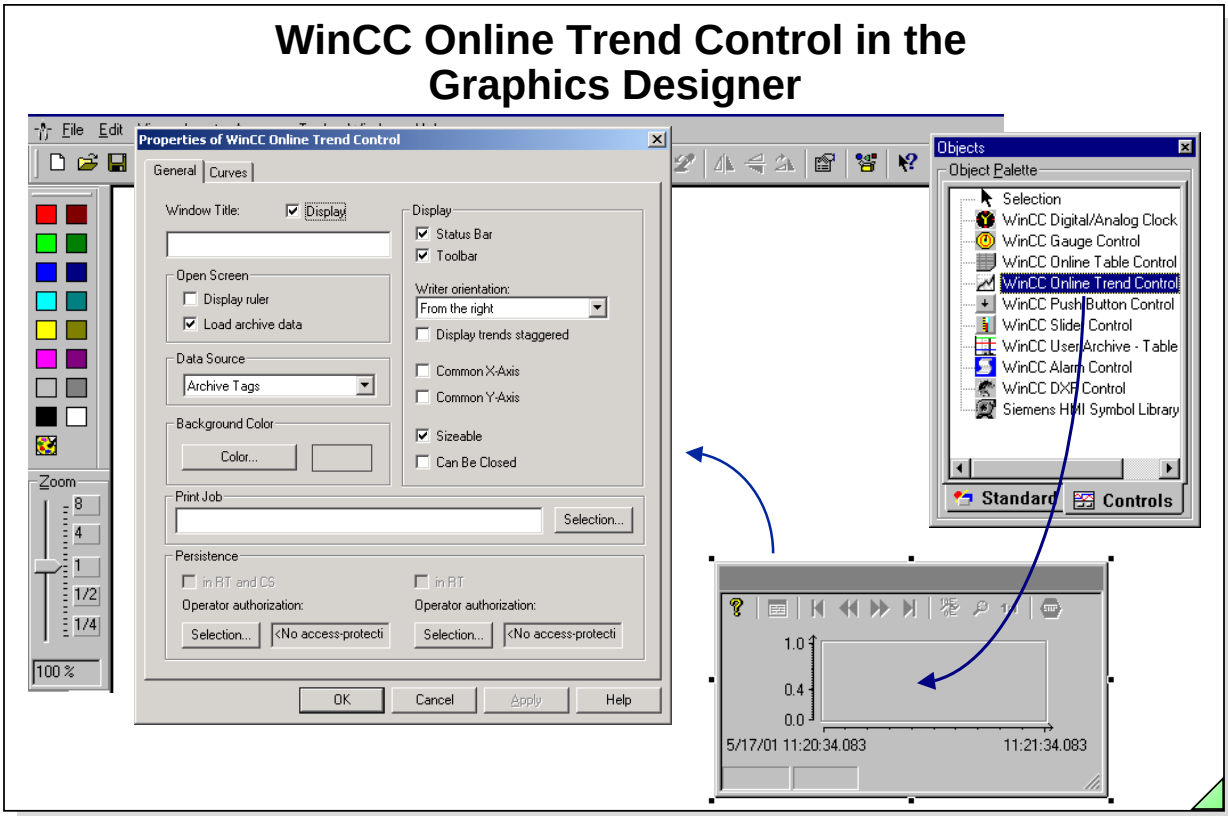
SITRAIN Training for Automation and Drives

Sequence for Connecting

1. Copy backup files to a local drive or insert memory medium (such as, CD).
2. Connect databases.
3. Values are immediately available in RT: For Trends, you only have to scroll through the archive.
   For messages, you only have to read in the messages once more (such as, changing from short-term archive into the long-term archive).
4. When you don't need the data any longer, disconnect the database.
5. Data can then be removed from the local drive.

You proceed in the same manner for Slow Archive.

WinCC Online Trend Control in the Graphics Designer



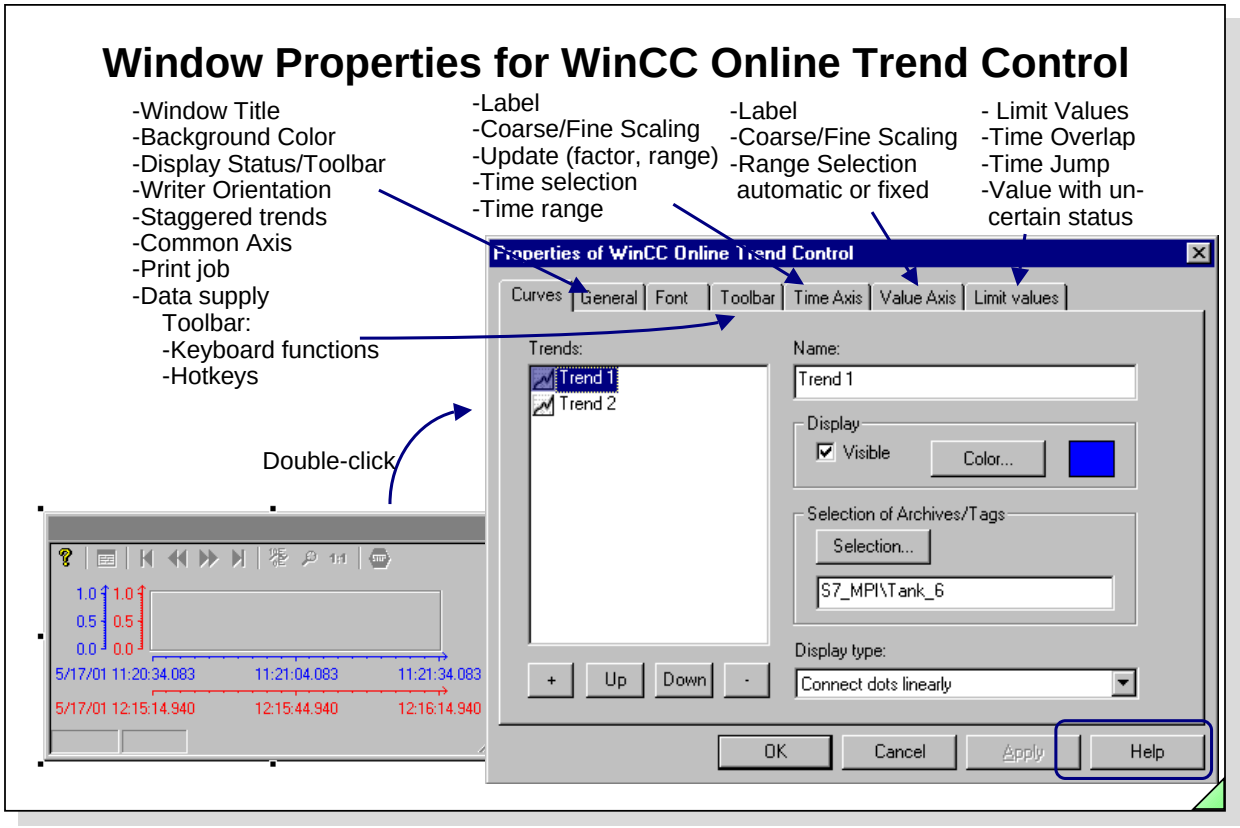
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_06E.16

SITRAIN Training for Automation and Drives

The WinCC Online Trend Control is the runtime window in which measured values are displayed during Runtime.

The Properties dialog appears automatically and allows you to make a selection of parameters for a fast dialog.



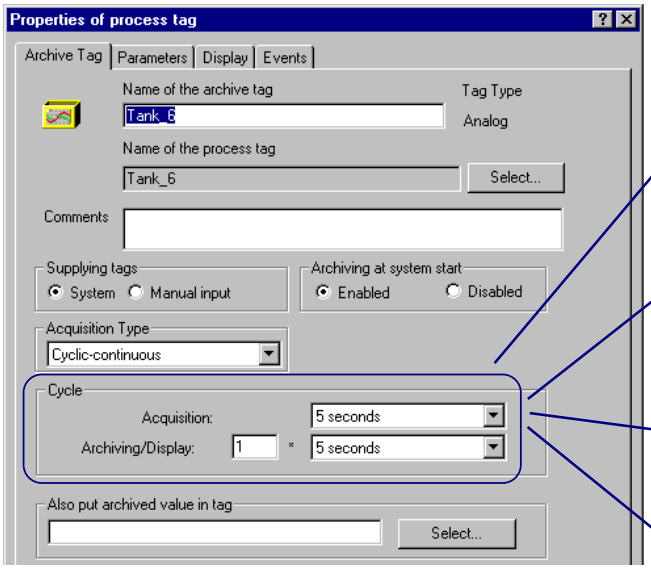
You specify the design and the display of a trend window with the help of the window properties for the WinCC Online Trend Control.

Call

You can call the Properties window with a **double-click on the left mouse button**.

You will find an exact description of the parameters in the Online Help.

Configuration Notes 1/2



Acquisition: 5 sec
Archiving/Display: 5 sec

This setting results in an optimized communication load

Acquisition: 1 sec
Archiving/Display: 5 sec

This setting results in an unnecessary communication load

Acquisition: 1 sec
Archiving/Display: 1 sec

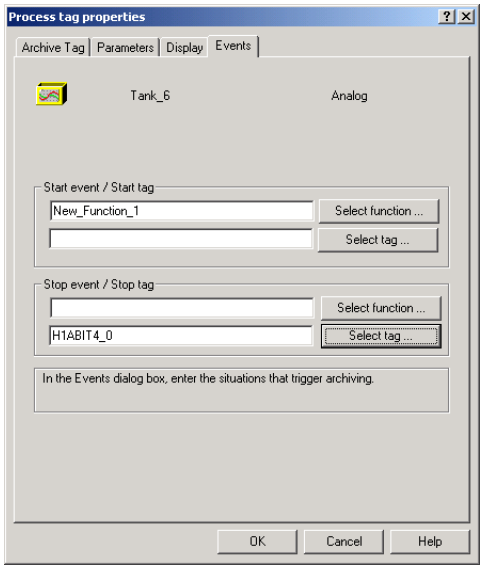
Archive becomes very large, unnecessary communication load

Acquisition: 2 sec
Archiving/Display: 10 sec

Makes sense for arithmetic, e.g. mean value

Note: You have to take into consideration that the acquisition cycle is a permanent load in the background (communication)

Configuration Notes 2/2



Start Events and Stop Events are functions that are called every 500 ms for every trend!
Better: Global Script Action (*.pas) with TLGLockArchiv

Example

If 20 trends are acquired using Cyclic-Selective, then
20 trends x 2 functions (Start/Stop) x 2 (processings 2x500ms) = 80 scripts/sec are processed. That means the processor load increases very quickly.

Better: configure an action (*.pas) in Global Script with tag trigger and the function TLGLockArchiv.

FAQ 2185 73

QUESTION:
How can you enable archiving per action

ANSWER:
You can enable or disable archiving during system start in the Tag Logging under Properties of Process Tag.

If you want to reverse this setting in Runtime, you have to configure the following action behind a key, for example:

```
#include "apdefap.h"
void OnClick(.....
{
TLGConnect(NULL,NULL);
TLGLockArchiv(NULL,"Prozesswertarchiv",FALSE,NULL);
TLGDisconnect(NULL);
}
```

Explanation:

"Prozesswertarchiv" --> Name of the process value archive

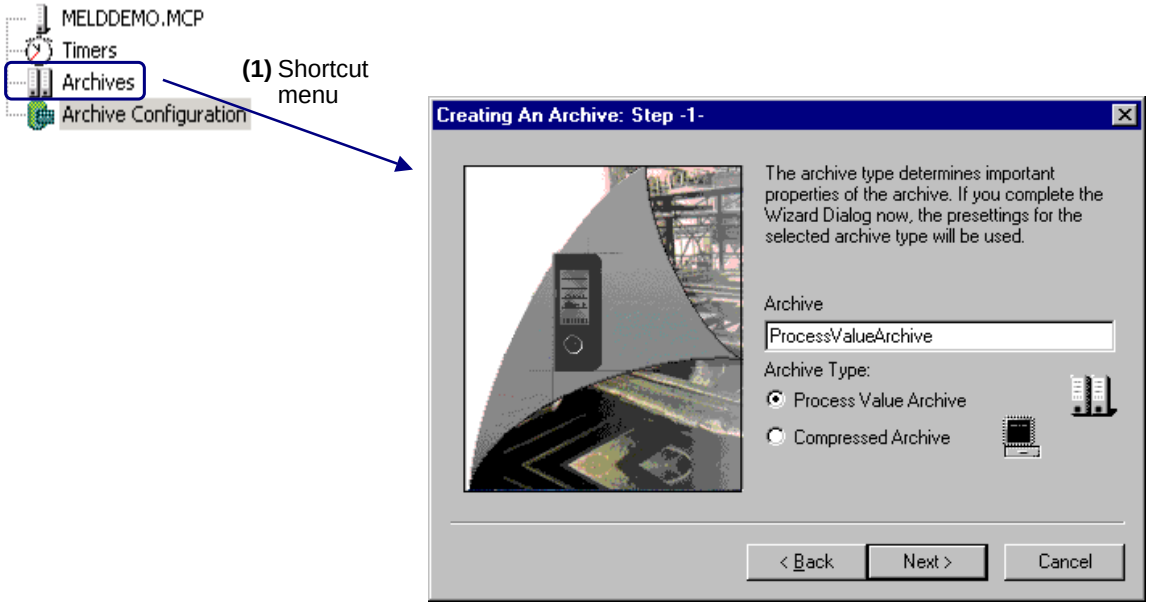
FALSE --> Unlock Archive

TRUE --> Lock Archive

Note

Group trends with Start/Stop Event in a separate archive.

Exercise Step 1/5 Archive Wizard



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_06E.20

SITRAIN Training for Automation and Drives

To make it easier for you to begin configuring, the measured value system has a configuration assistant (Archive Wizard) that takes all significant settings into consideration to set up an executable archive quickly.

Exercise 1/5

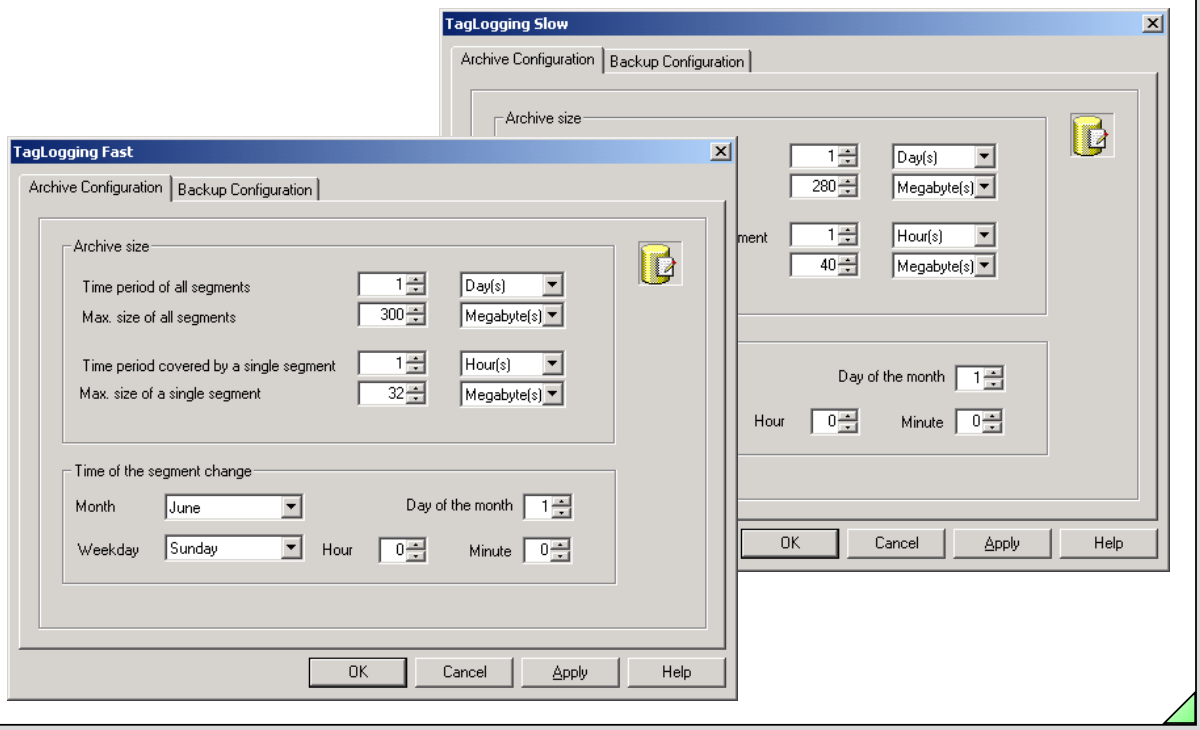
Please run through the Archive Wizard (1) and set the following parameters:

- Archive name: Measured_values_Station_1_10
- Archive type: Process value archive
- Tags: Trend_1
  IVar_Meas

Process Tag Properties

Trend_1: Acquisition 1 sec, archiving 1sec
Ivar_Meas: Acquisition 2 min, archiving 2 min.

Exercise Step 2/5 Archive Configuration



Date: 27.10.2003
File: SWINCC_06E.21

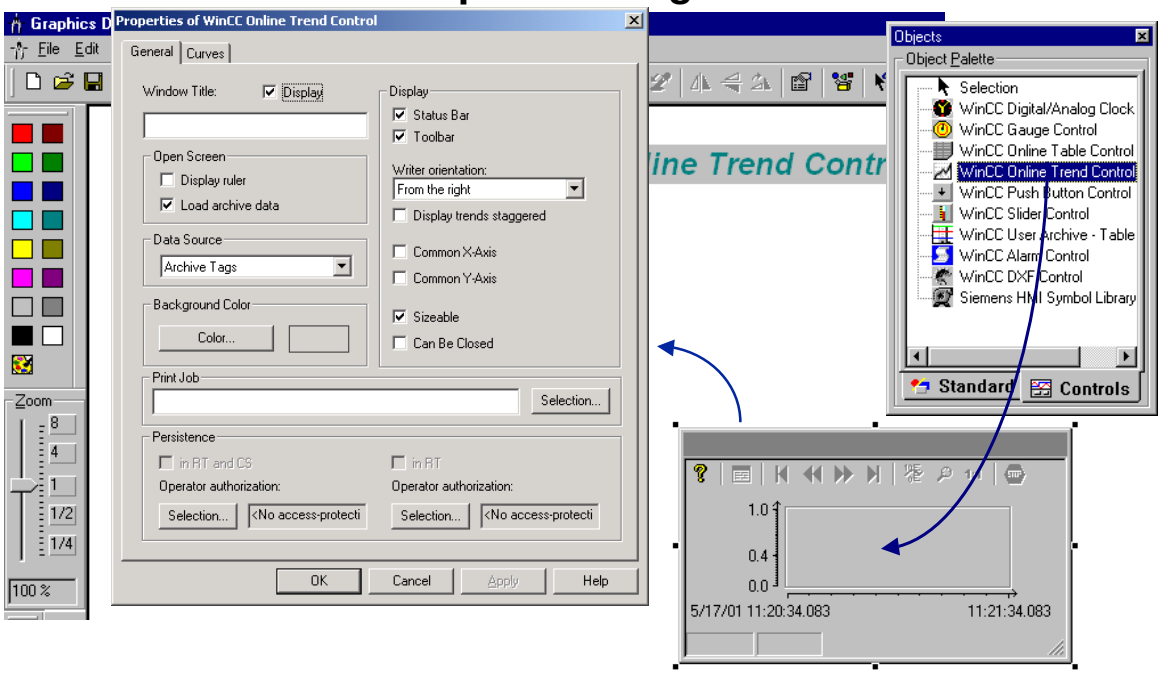
Task

Create an archive configuration that archives the measured values in a fast short-term archive for one day when the acquisition is <=1min and the size is 300 megabyte. When the acquisition >1min, the measured values are to be archived for 7 days in a slow short-term archive with a size of 280 megabyte. Choose the current date and time for the time of the segment change.

Exercise 2/5

Use the settings from the slide above.

Exercise Step 3/5 WinCC Online Trend Control in the Graphics Designer



Copy the picture "kurven.pdl" from the catalog "ab_uebung" into your project.

Date: 27.10.2003
File: SWINCC_06E.22

Objective

You are to create a trend window template in which you display the two trends from the measured value archive.

Exercise 3/5

From the catalog "ab_uebung", copy the picture "kurven.pdl" into your project with the subdirectory *GraCS*.

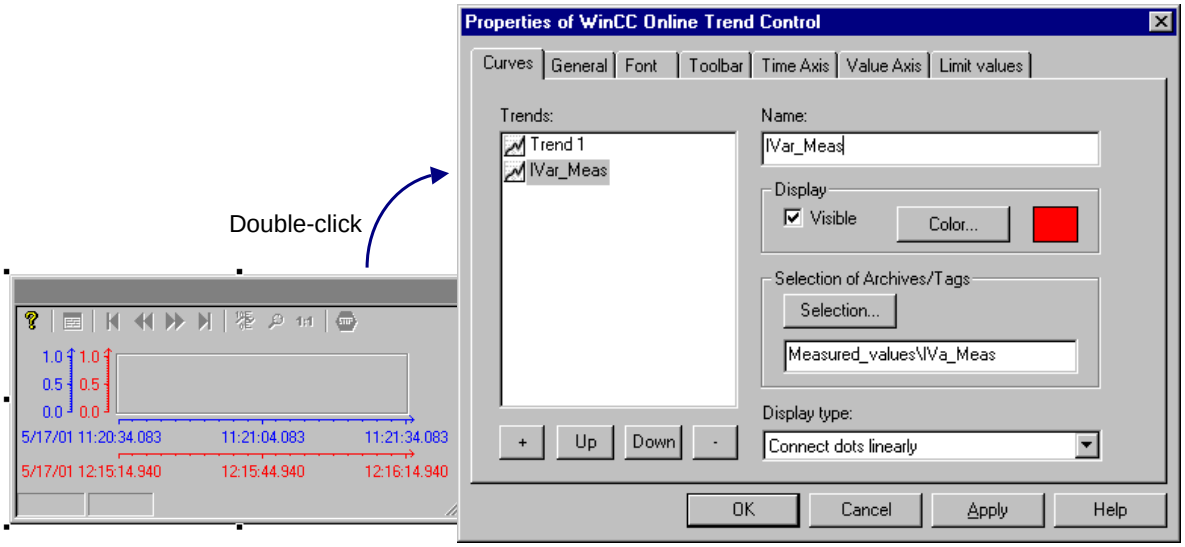
Please make the following settings in the window that appears

:

- **Trend 1**
  Display: visible
  Color: red
  Tag: Trend_1
  Connect dots linearly

- **Trend 2**
  Display: visible
  Color: green
  Tag: IVar_meas
  Connect dots linearly

Exercise Step 4/5 Window Properties for WinCC Online Trend Control



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_06E.23

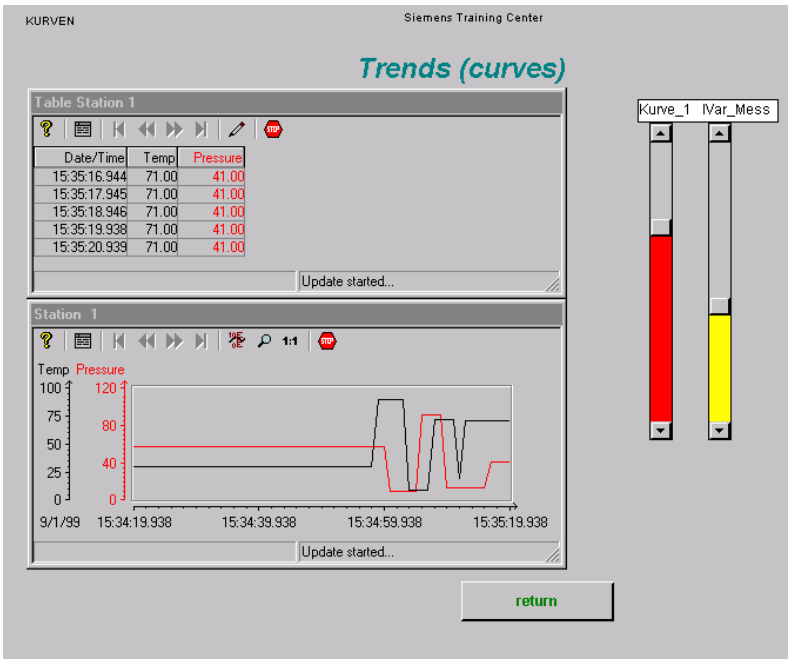
SITRAIN Training for Automation and Drives

Exercise 4/5 Please make the following settings in the window that appears:

- **Trend 1**	**- Trend 2**
- X axis	
Label: time	Label: time
Time range: 1 minute	Time range: 1 minute
Update: yes	Update: yes
- Y axis	
Label: Temp	Label: Pressure
Coarse scaling: 25	Coarse scaling : 20
Fine scaling: 5	Fine scaling: 5
Begin and end 0 - 100	Begin and end 0 - 120
Display decimal	Display decimal
Decimal places: 0	Decimal places: 0
Range selection automatic: no	Range selection automatic: no

- Call the trend picture from the Start screen.

Exercise Step 5/5 Function Test



"TagLoggingRuntime" must be selected in the start-up list!

Exercise 5/5 You are to carry out a function test on the configuration you created. Test the attributes you assigned.

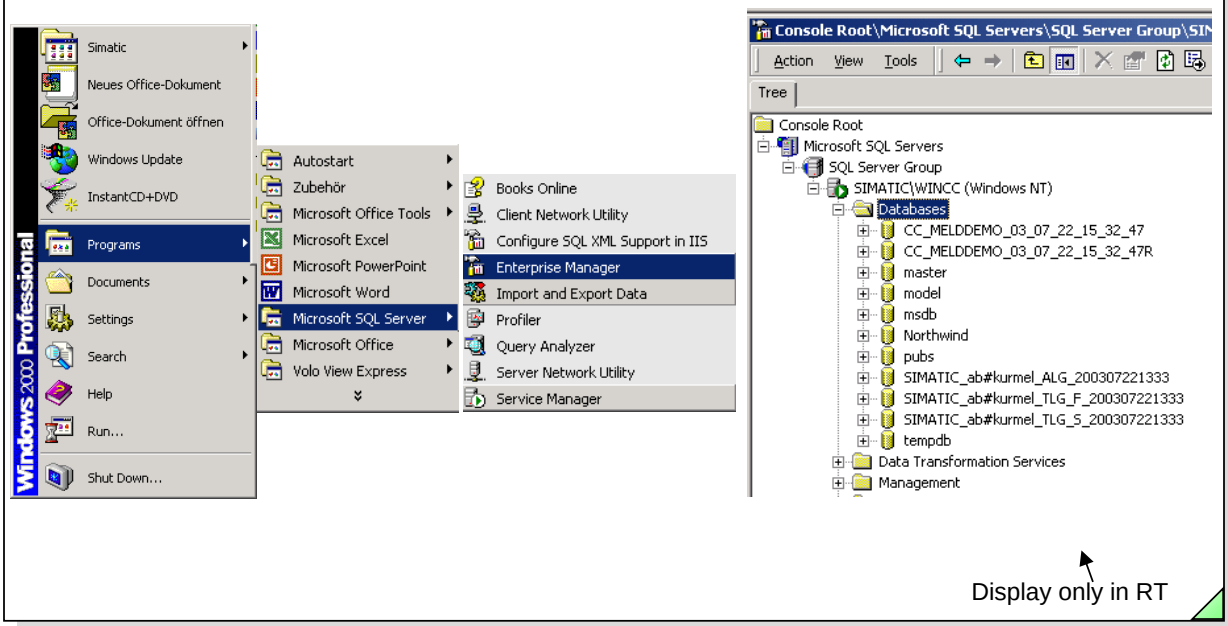
Note You **must** select "Tag Logging Runtime" in the start-up list!

Exercise 5 Carry out the following operations in the run time:

- Setting the value range for the Y axis
- Switching on/switching off the reading line
- Switching on/switching off the zoom
- Selection according to the time period
- Hide trends
- Recorder function (vertical trend window)

Exercise 6 Use a table window to configure the measured value output.

New: Database Tools 1/3



Delivery

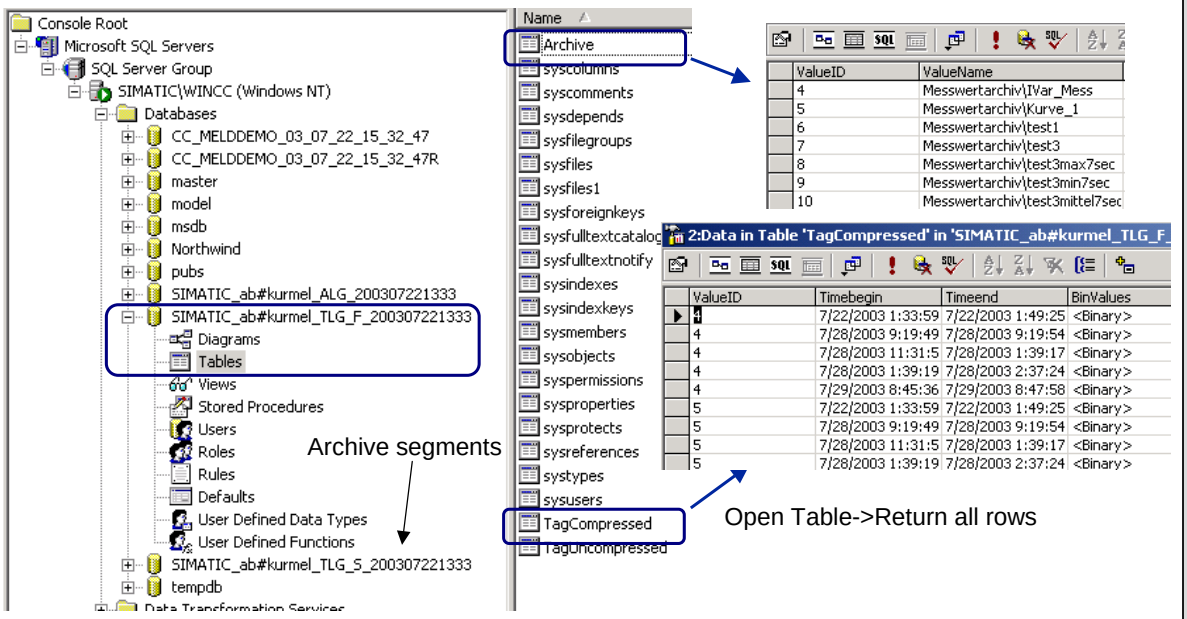
The Enterprise Edition contains all necessary functions for supporting company-wide data processing.

Other graphic tools that are supplied with the SQL Server:
+Import and Export Data
+Query Analyser
+Service Manager

Optional

Other editions: Standard Edition; Personal E.; Developer E.

New: Database Tools 2/3



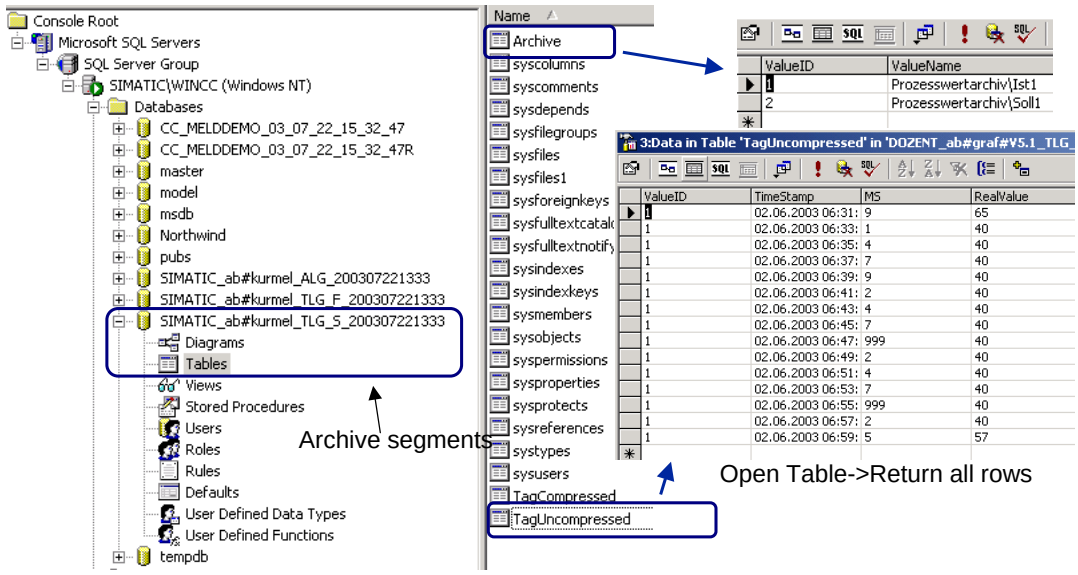
Tables TagCompressed: Is written in TagLogging Fast Archive. Data are stored as compressed blocks (BLOB, Binary Large OBject). Reading out only with WinCC Tools or WinCC OLE- DB Provider, which is a part of the Connectivity Pack option package.

Note It is not possible to uncompress the data with the MS OLE- DB Provider.

Trend Storage
+Different trends are stored in the same table
+Differentiation through ValueID
+Assignment of ValueID and trend names in the “Archive“ table

Note You cannot make any changes to the database, because the database can become inconsistent.

New: Database Tools 3/3



Tables TagUnCompressed: Is written in TagLogging Slow Archive. Data are stored uncompressed. You can also read out using standard interfaces.

Trend Storage
+Different trends are stored in the same table
+Differentiation through ValueID
+Assignment of ValueID and trend names in the “Archive“ table

Note You cannot make any changes to the database, because the database can become inconsistent.

User Archives

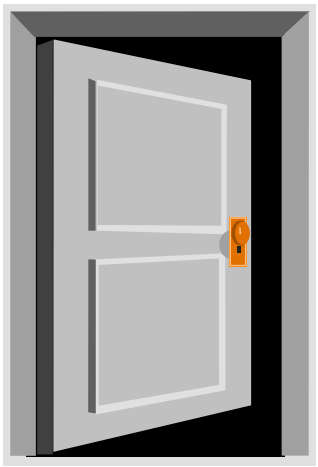
SIMATIC HMI

Date: 27.10.2003
File: SWINCCS_07e.1

SITRAIN Training for Automation and Drives

Contents Page

Objectives



- Creating a User Archive
- Creating a Table View
- Creating a Form View
- Accessing a User Archive
- Data Transfer to a Controller

SIMATIC HMI

Date: 27.10.2003
File: SWINCCS_07e.2

SITRAIN Training for Automation and Drives

Upon completion of this chapter, the participant should be familiar with:

- the advantages of the User Archive.
- the Table View, based on an example.
- the Form View, based on an example.
- the access possibilities, based on an example.
- the communication possibilities, based on an example.
- the application possibilities, based on an example.

User Archives

Advantages of user archives for:

- Recipes
- Batch data
- Production data
- Data storage for data from the controller
- Tool management
-
-

Access to user archives through:

- Table window
- Internal functions
- API functions (described in the WinCC Help, among others)
- ODBC / SQL
- MS Query
- Communication with the controller using tags
- Communication with the controller using raw data

SIMATIC HMI

Date: 27.10.2003
File: SWINCCS_07e.3

SITRAIN Training for Automation and Drives

General Information The user archives package is available as a WinCC option package.
It consists of
-user documentation
-internal functions
-keydisk for enabling the software.
The user archive software is already included on the WinCC CD.

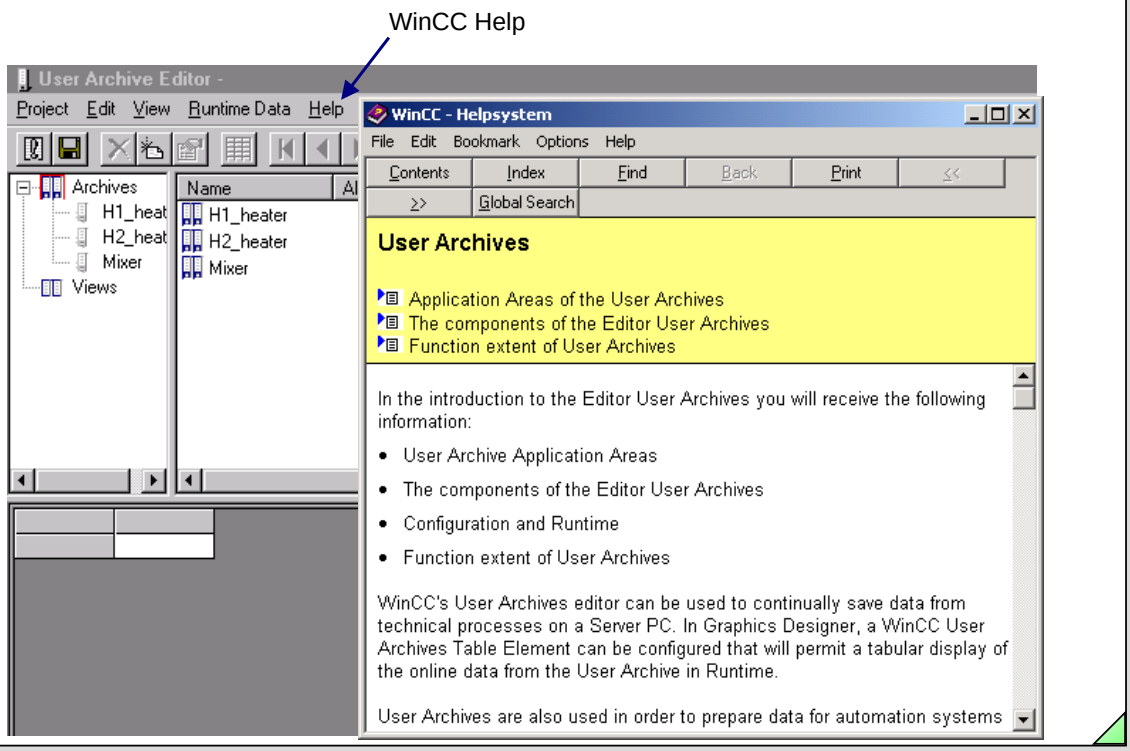
Applications The user archives can be utilized and applied in a great variety of applications (see slide above).

Basics WinCC User Archives is a database system which you can configure for your own requirements.
The WinCC User Archives has two types of database tables.

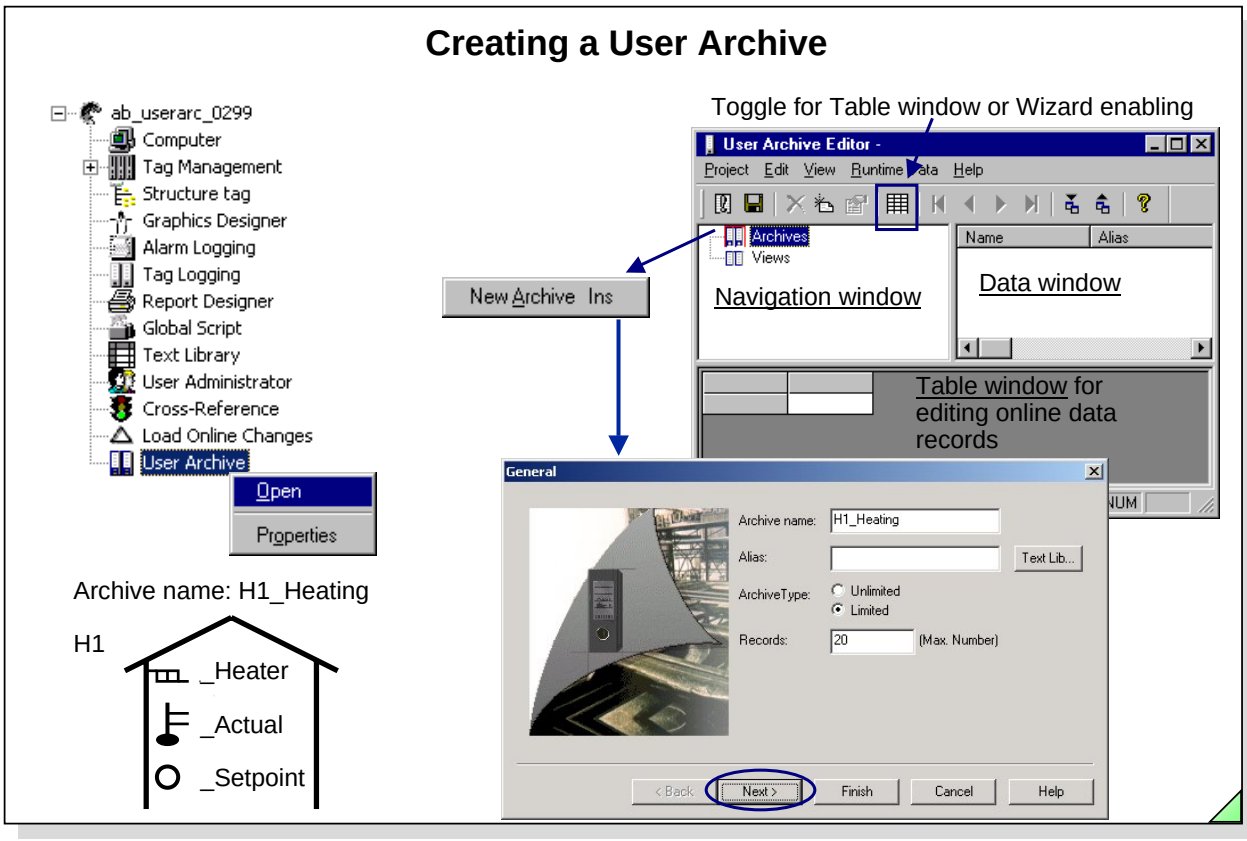
Archives Archives are database tables in which you can create your own data fields. Archives are used to store data and offer a database-type access to these data.

Views Views obtain data from the archives and are used to summarize data, in order to receive an overview of product groups, for example.
The data in a View can also be edited in Runtime. The changed data are transferred into the original archive.

WinCC Help (System)



WinCC provides you with a powerful help system for the configuration of the user archive.



Example

Temperature data from three components is to be stored in the visualization system.
For this, a user archive is configured with the fields:

-H1_Heater
-H1_Actual
-H1_Setpoint

Creating an Archive

You can create a user archive in the User Archive Editor, Navigation window.
You can use the Wizard to specify:
the archive name with the number of data records, communication parameters, control tag parameters, authorization and flag parameters.

Creating Archive Fields

In the navigation window “New Field”, you can define the three fields mentioned above with the help of the Wizard.
Field name, Value, Authorizations and Flags are assigned here.

Note

Any changes to user archives only take effect after **Saving**.

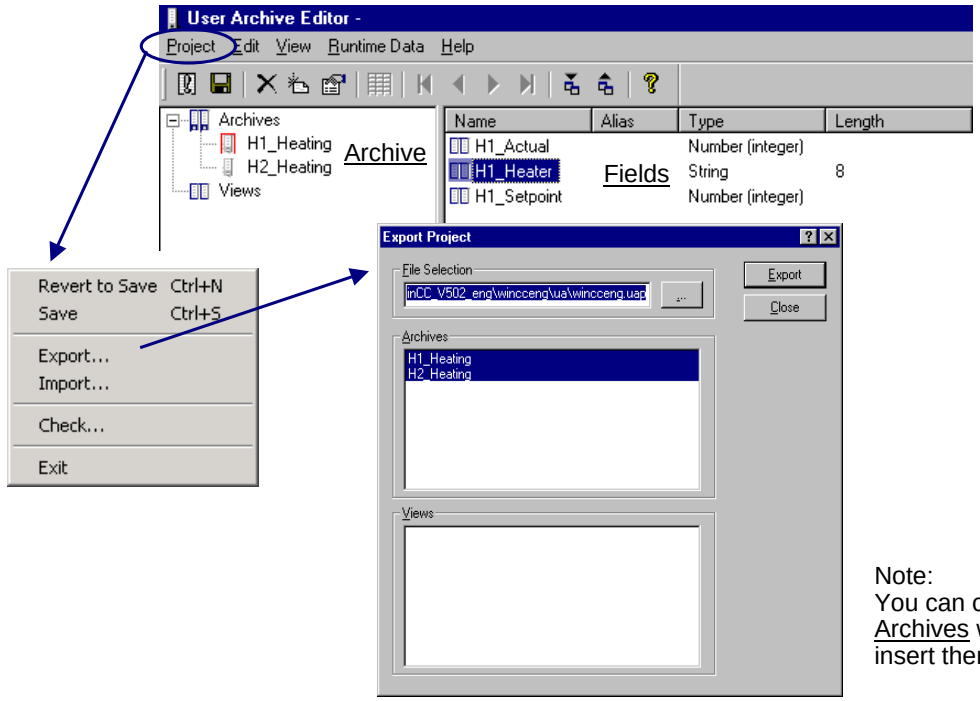
Table Window

You can switch on or switch off the table window by **clicking on Edit in the menu bar and then clicking on Runtime Data in the drop-down menu**. In the Table window, you can assign values to the three configured archive fields.

Control

The Control (WinCC User Archive Table Element) is required in the Graphics Designer for outputting the table element.

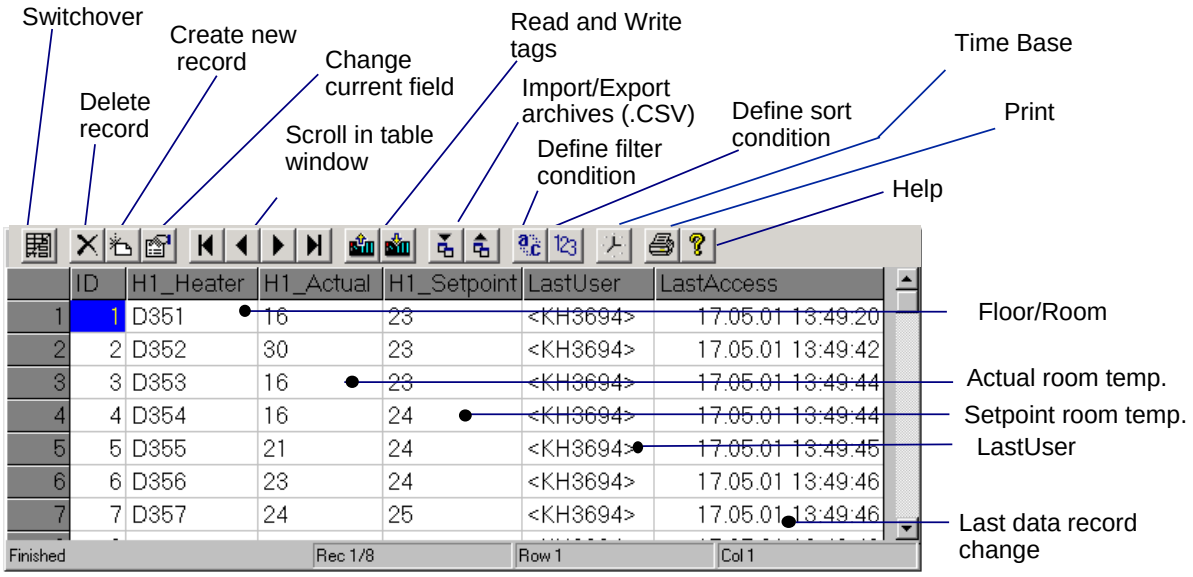
Exporting/Importing a User Archive



Note:
You can copy Fields and Archives with Ctrl+C, and insert them with Ctrl+V

Export/Import You can transfer the structure and the data for a new archive with the Export/Import function.

Table View of the User Archive Control



Note: The ID number is an internal reference of the database. The numbering does not have to be consecutive. It should not be used for selections.

Functions

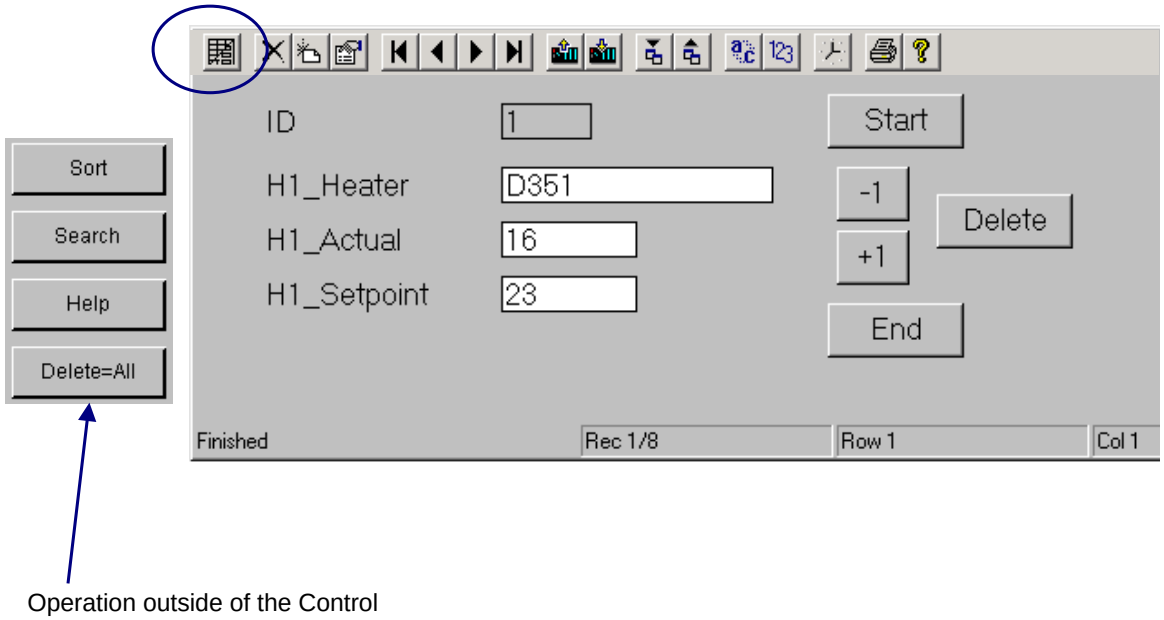
The table window makes it possible to edit the data of the user archive in various ways:

-Switchover	(switch between Form view and Table view)
-Delete record	(selected record is deleted)
-Create new record	(and confirm every time with the Enter key)
-Change current field	
-Scroll table window	(scroll forward and backward, as well as to the beginning or end of the archive)
-Read and write tags	(WinCC tags)
-Import/Export archives	(Attention! CSV format)
-Define filter condition	(Option! can be formulated directly in the database in SQL language)
-Define sort condition	(Option! can be given directly in the database in SQL language)
-Help	(request)

Advantages

The table window offers simple access to the data of the user archive.

Form View of the User Archive Control



Functions

The form window has a user interface which you can create according to your needs. The form view has three field types:

- static texts (any static text you wish)
- input fields (makes it possible for you to display and edit the data fields of the current archive)
- buttons (button allows you your own button instead of the symbols of the form field)

Advantages

The form window offers simple access to the data of the user archive which you yourself created.

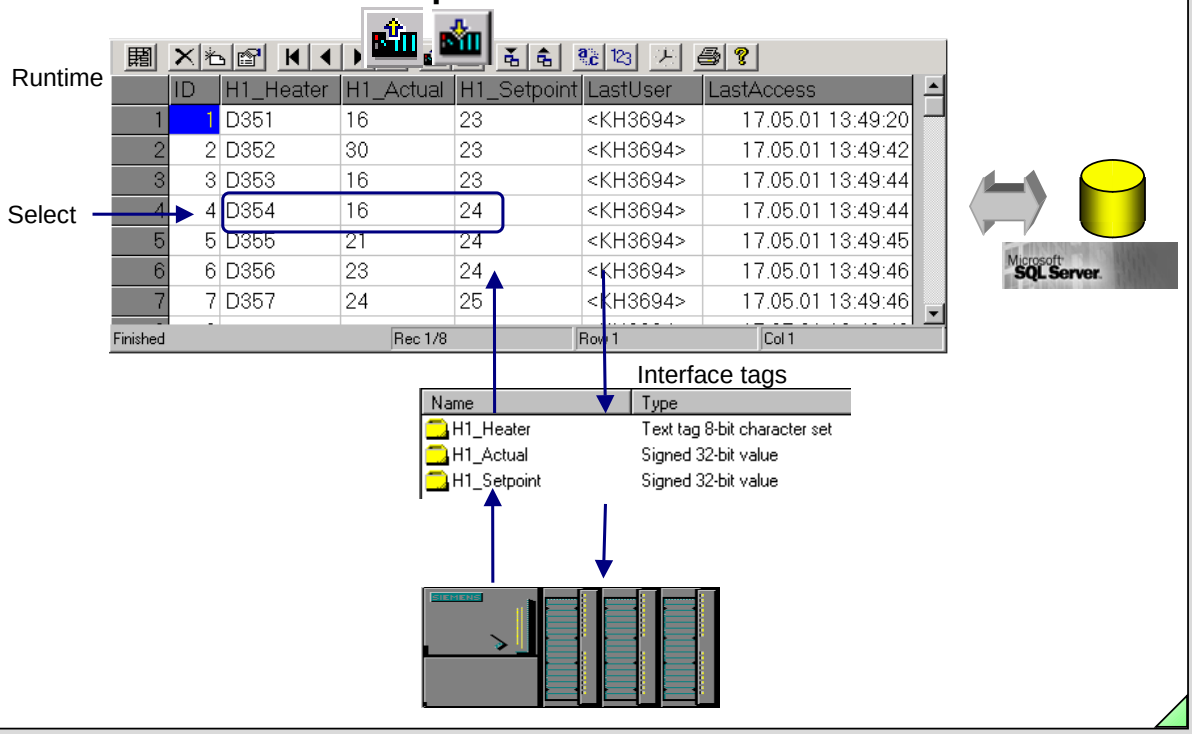
Note

It is also possible to place all toolbar buttons outside of the Control in the Graphics Designer.

Configuration: Create button
Object properties/ Event/ Mouse/ Direct connection/
Source: Enter constant values
Target: Object in Picture -- Select (Object: OLE Control, Property: **PressTBButton**

The Constants: 1 Form; 2 First; 3 Previous, 4 Next; 5 Last; 6 Delete; 7 New; 8 Edit; 9 ReadVar; 10 WriteVar; 11 Import; 12 Export; 13 Filter, 14 Sort; 15 Help;

Communication with the Controller using Tags, Operator Controlled

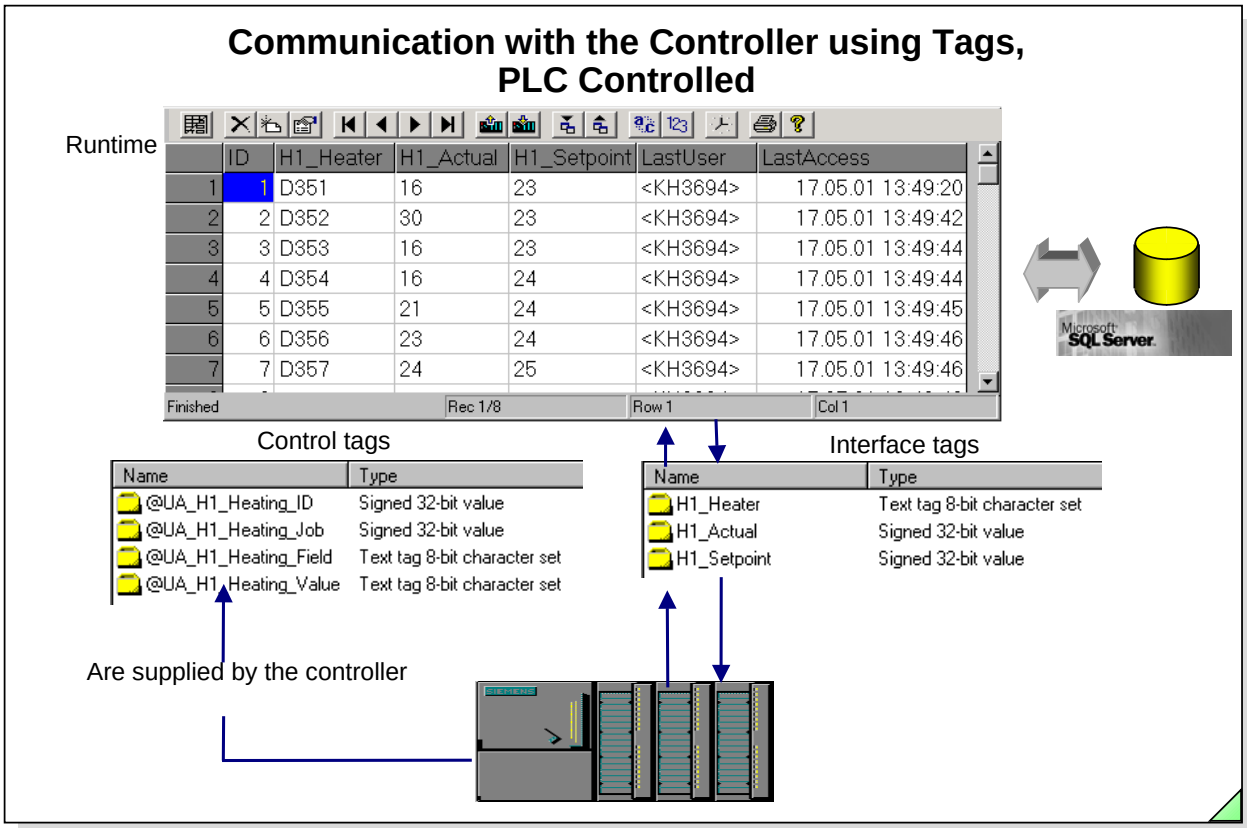


SIMATIC HMI

Date: 27.10.2003
File: SWINCCS_07e.9

SITRAIN Training for Automation and Drives

You can use tags to make a simple communication with the controller. You have to take this application into consideration when you assign parameters to the Wizard. The tags are then generated in the Tag Management.



You have to take this application into consideration when you assign parameters to the Wizard. The tags are then generated in the Tag Management.

Function of the Control Tags

ID The identifier (corresponds to the data record number) of the user archive

Job Three jobs are possible: reading, writing and delete:

SIMATIC to Archive = 6

Archive to SIMATIC = 7

Delete = 8

After the job has been executed there is an error identifier in this control tag:

No error = 0

Error = -1

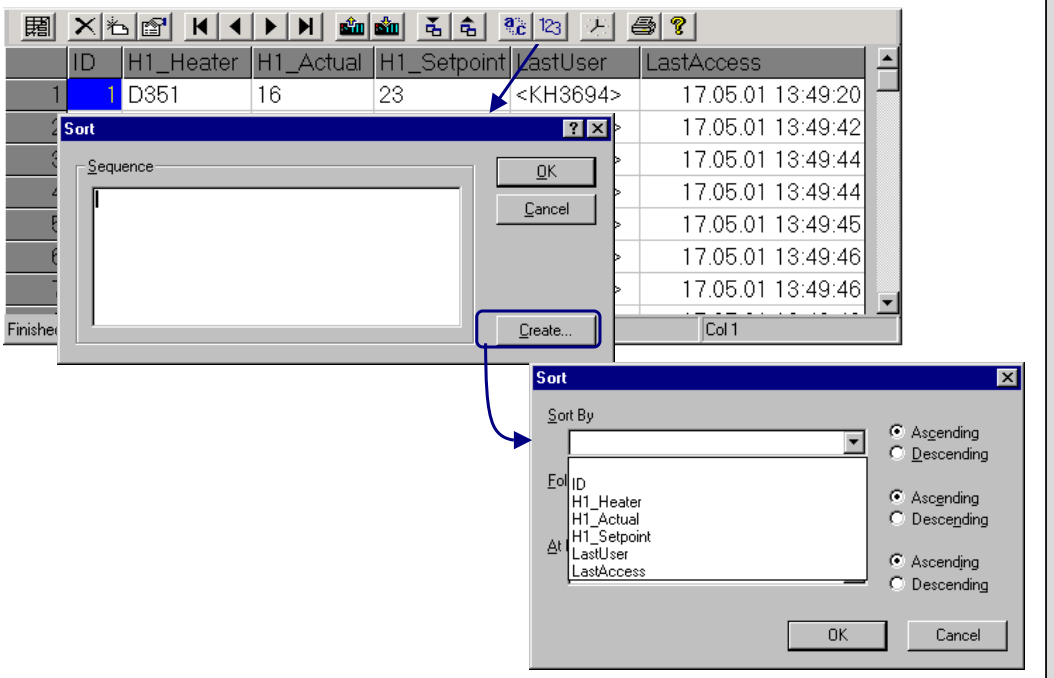
Field The archive field (Column labeling, Ingredients, for example)

Value The archive field value (for example, ‘Flour‘)

ID	Job = 6	Job = 7	Job = 8
-1	Add data record	-	Delete data record with the lowest ID
-6	Read data record with the lowest ID	Write data record with the lowest ID	Delete data record with the lowest ID
-9	Read data record with the highest ID	Write data record with the highest ID	Delete data record with the highest ID

Sort

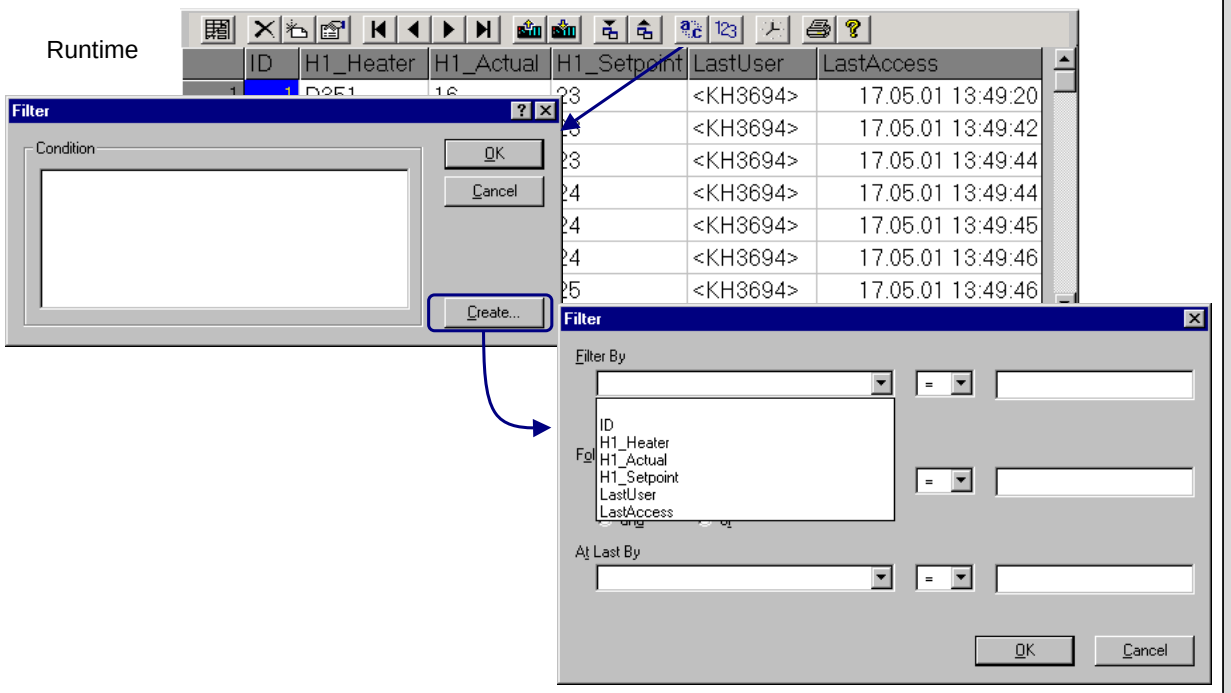
Runtime



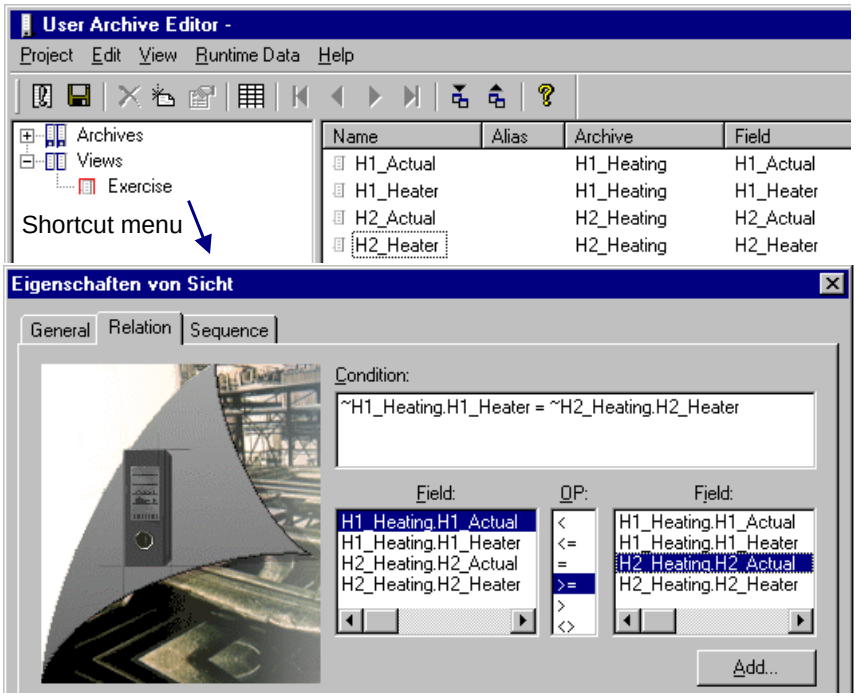
Date: 27.10.2003
File: SWINCCS_07e.11

Filter Condition

Runtime



Creating Views



Date: 27.10.2003
File: SWINCCS_07e.13

General Information

A view is an arithmetical logic linking of fields from at least two archives. These archives must have at least one thing in common or they must have a relationship.
If several conditions are given then these are **logically** gated with one another.
It must be noted that it only makes sense to link fields of the same type.
The following logic operations are possible:

- <
- <=
- 0
- >
- >=
- <>

Example

Two archives: H1_Heating, H2_Heating are to be gated with specific fields (H1_Heater = H2_Heater, as well as H1_Actual >= H2_Actual).
In the view formation, the actual value of H1_Heater is only displayed if it is >= the actual value of H2_Heater.

Advantages

Only values of the same type field are displayed, that are **equal to** (H1_Heating) or **greater than or equal to** the **archive H2_Heating**.

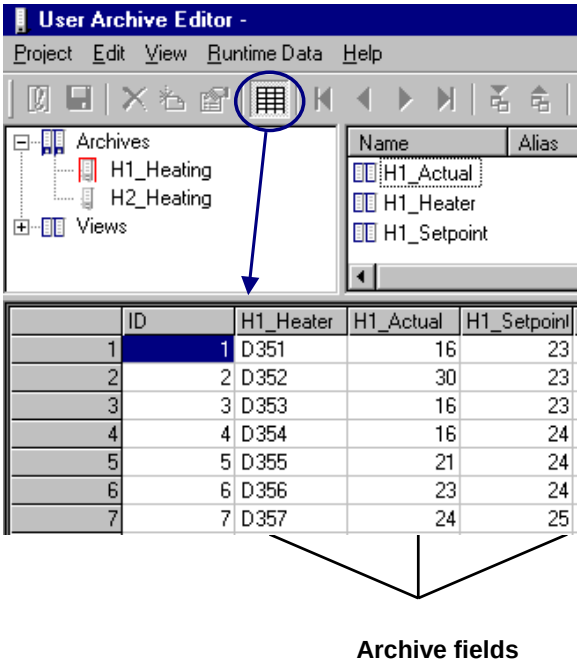
Note

The data in a View can also be edited in Runtime. The data that are changed are transferred to the original archive.

Exercise 1: Creating an Archive

User Archive Editor

- **Creating an Archive**
- **Creating Archive Fields**
- **Editing Table Windows**



Archive fields

Task

You are to create a user archive to store temperature data. For this, you will require three fields:
-H1_Heater (text tag 8bit character set / character string)
-H1_Actual (signed 32 bit value)
-H1_Setpoint (signed 32 bit value)

Creating an Archive

First of all, you create the archive “H1_Heating” in the User Archive.
-The archive type should only have 8 records.
-Communication takes place via WinCC tags.
-The control tags via the button “Create” (@UA_H1_Heater /internal),
-From the four buttons choose “Select”, the corresponding tag e.g. _ID etc.
-Authorizations and Flags, Button “OK”.

Creating Archive Fields

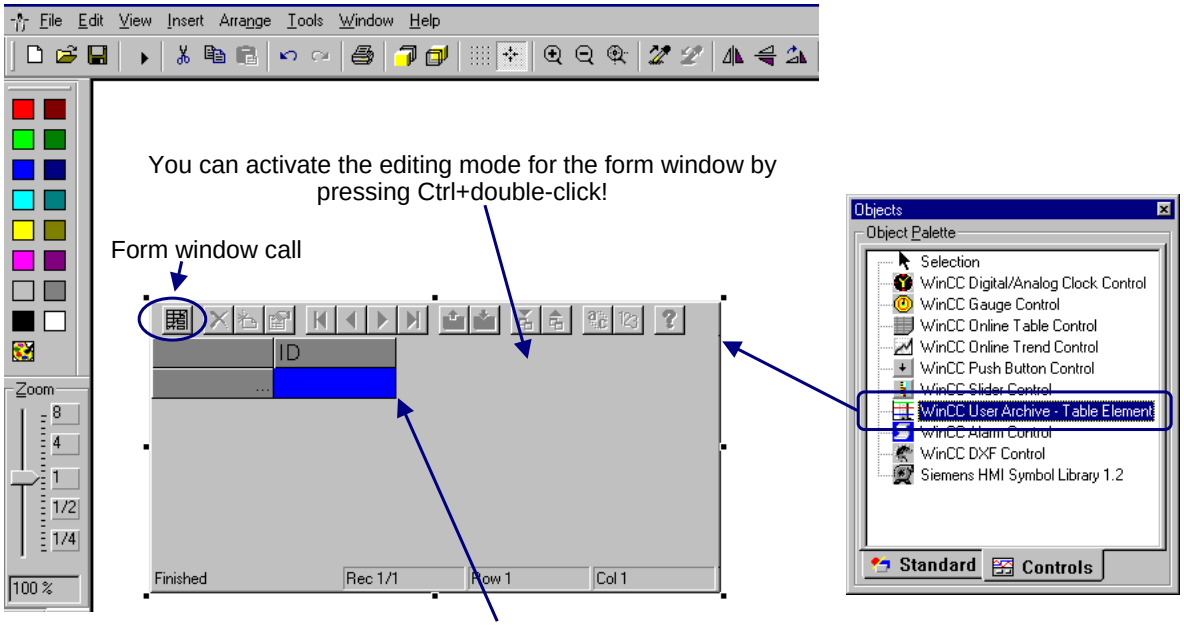
In the navigation window, create “New Field” with the help of the wizard.
-General: -Field name **“H1_Heater”**, -Type: character string, -length: 8
-Values: WinCC tag: select “H1_Heater”.
-Authorizations and Flags: “no entry”, Button “OK”.
Next Field: yes
-Field name: **“H1_Actual”**, -Type: Number double
-Values: WinCC tag: select “H1_Actual”, Start Value 0.
- Authorizations and Flags: “no entry”, Button “OK”.
Next Field: yes
-Field name: **“H1_Setpoint”**, -Type: Number double
-Values: WinCC tag: select “H1_Setpoint”, Start Value 0.
- Authorizations and Flags; “no entry”, Button “OK”.

Note Table Window

Any changes you make to user archives only take effect after **Saving**.
You can switch on or switch off the table field by **clicking on Edit in the menu bar and then clicking on Runtime Data in the drop-down menu**.

-by double-clicking on one of the fields, you can enter data. Here you can specify values.

Exercise 1: Integrating a Control



Date: 27.10.2003
File: SWINCCS_07e.15

Control

A Control is created in the Graphics Designer in which the table element configured above is displayed (the procedure is displayed in the slide above).

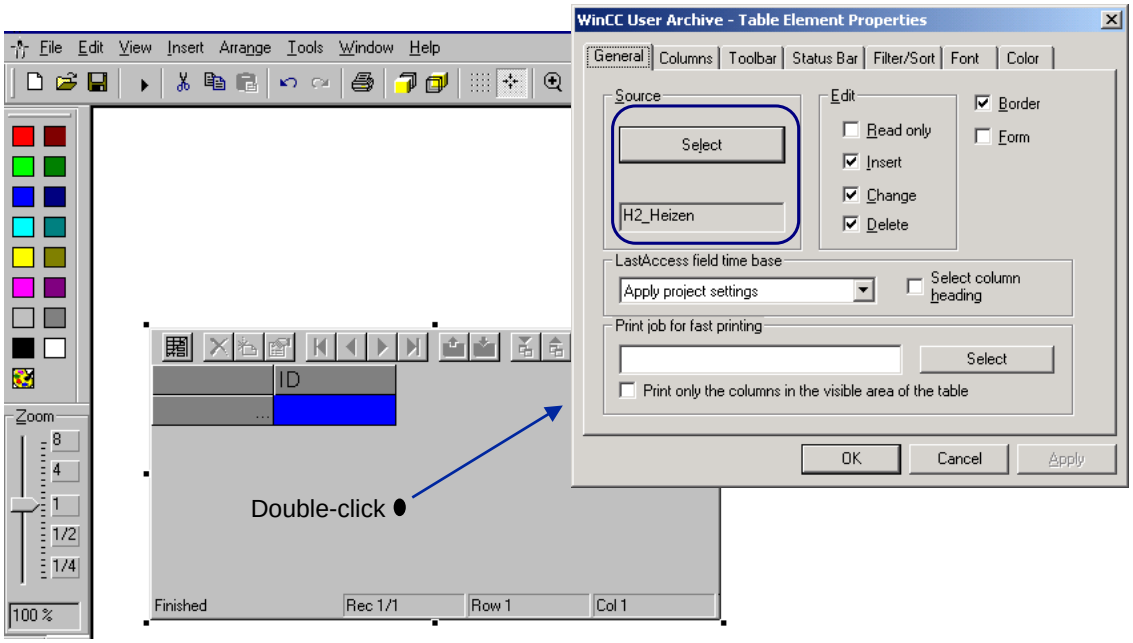
Note

In order to specify the properties of the “WinCC User Archive Table Element”, you must **double-click** within the Control area. Here you determine what is to be displayed in the Graphic screen (enter **Archive name** or **View name**). In addition, you can also set the authorizations of the access type: (-read only, -insert, -change, -delete), the toolbar, the status bar etc.

Inputting Values in RT

After you choose the picture with the table window, you can then create records and enter values in the archive.

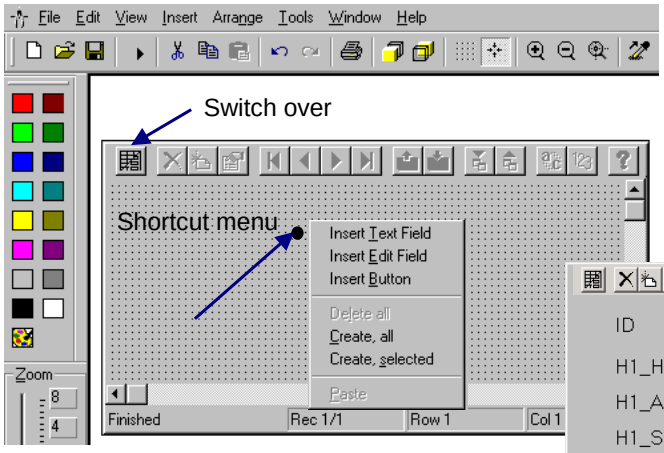
Exercise 1: Setting Control Parameters



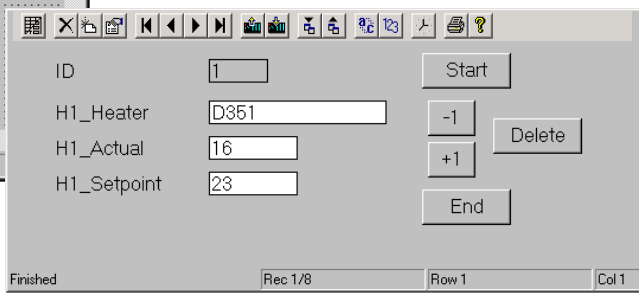
Note: Every time you make a change to the archive structure, you have to reselect the source (archive).

Exercise 2: Form View

- Expanding the exercise picture with the form view of the User Archive- Table Element



Configuration



Runtime

SIMATIC HMI

Date: 27.10.2003
File: SWINCCS_07e.17

SITRAIN Training for Automation and Drives

General Information You can configure the User Archive Control Form in the Graphics Designer yourself, according to your needs. It is used to display the user archive data during Runtime.

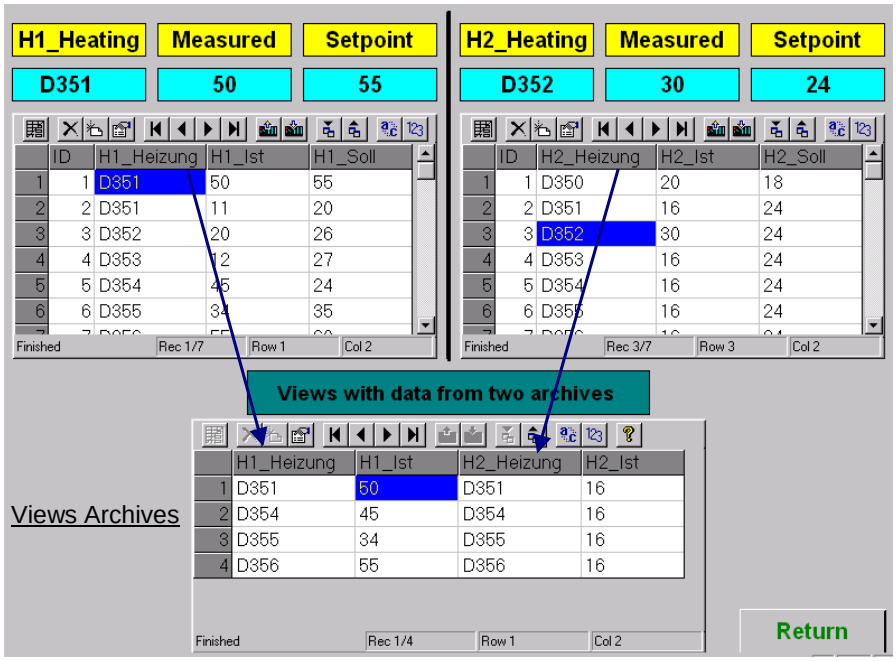
Task -The data of one record are to be displayed in the form view.
-Configure the necessary fields and test the function in Runtime operation.

Note In order to be able to edit the form field of the WinCC User Archive Table Element, **double-click** in the window of the User Archive Table Control **at the same time** as you press the **Ctrl key**. The table view of the Control then appears.

Switch over You use this icon to switch between the form view and the table view.

Exercise 3: A Second User Archive

- Expanding the exercise picture



Date: 27.10.2003
File: SWINCCS_07e.18

Task

You are to create a second user archive with the three fields:
-H2_Heater (text tag 8bit character set / character string)
-H2_Actual (signed 32 bit value)
-H2_Setpoint (signed 32 bit value)

- Use the same procedure as in Exercise 1 (**Attention!** only with H2_)

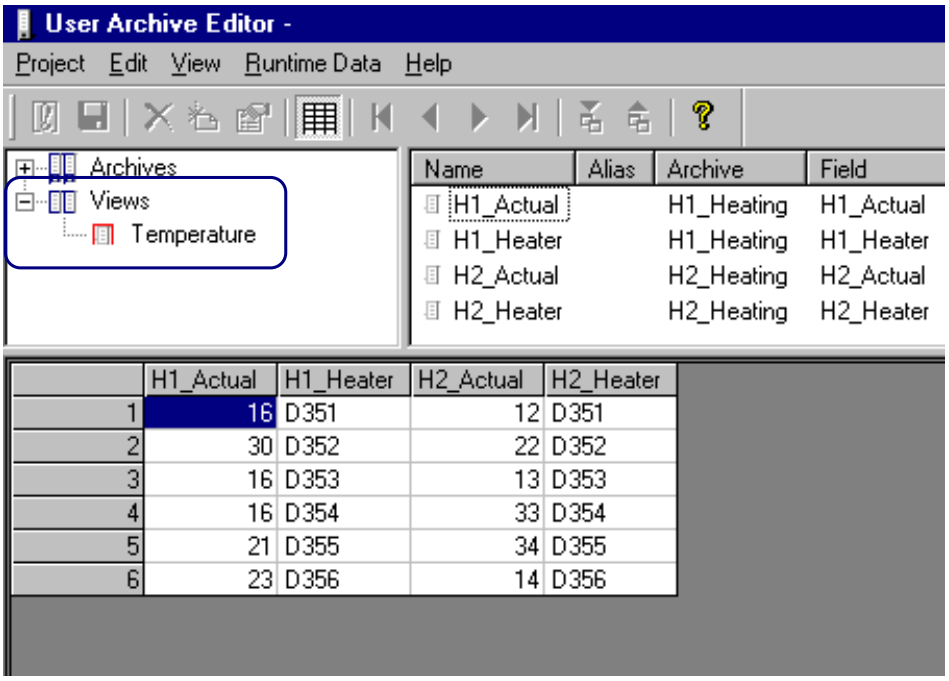
- then integrate the Control in the Graphics Designer, as in exercise 1 (Archive H2_Heater).

Runtime

Test the function in Runtime operation.

Exercise 4: Views of the Two Databases

- Expanding the exercise picture



Task

The two databases (H1_Heating and/or H2_Heating) are to be linked with specific fields (H1_Heater = H2_Heater, as well as H1_Actual >=H2_Actual) using **Views**.
In the View formation, the value of H1_Heater is only displayed if it is equal to or greater than H2_Heater.

Creating Views

First of all, create the view “Temperature” in the User Archive with the help of the archive wizard.

Creating Columns

You can choose the user archives through the “General” dialog box.

- Archive: H1_Heating
- Field: H1_Heater
- Column name: H1_Heater

New Column:
- Archive: H2_Heating
- Field: H2_Heater
- Column name: H2_Heater

New Column:
- Archive: H1_Heating
- Field: H1_Actual
- Column name: H1_Actual

New Column:
- Archive: H2_Heating
- Field: H2_Actual
- Column name: H2_Actual

Relation

Views/ Temperatures/ Properties (Properties of View)/ Relation/
Condition: H1_Heating. H1_Heater = H2_Heating. H2_Heater
H1_Heating. H1_Actual >= H2_Heating. H2_Actual

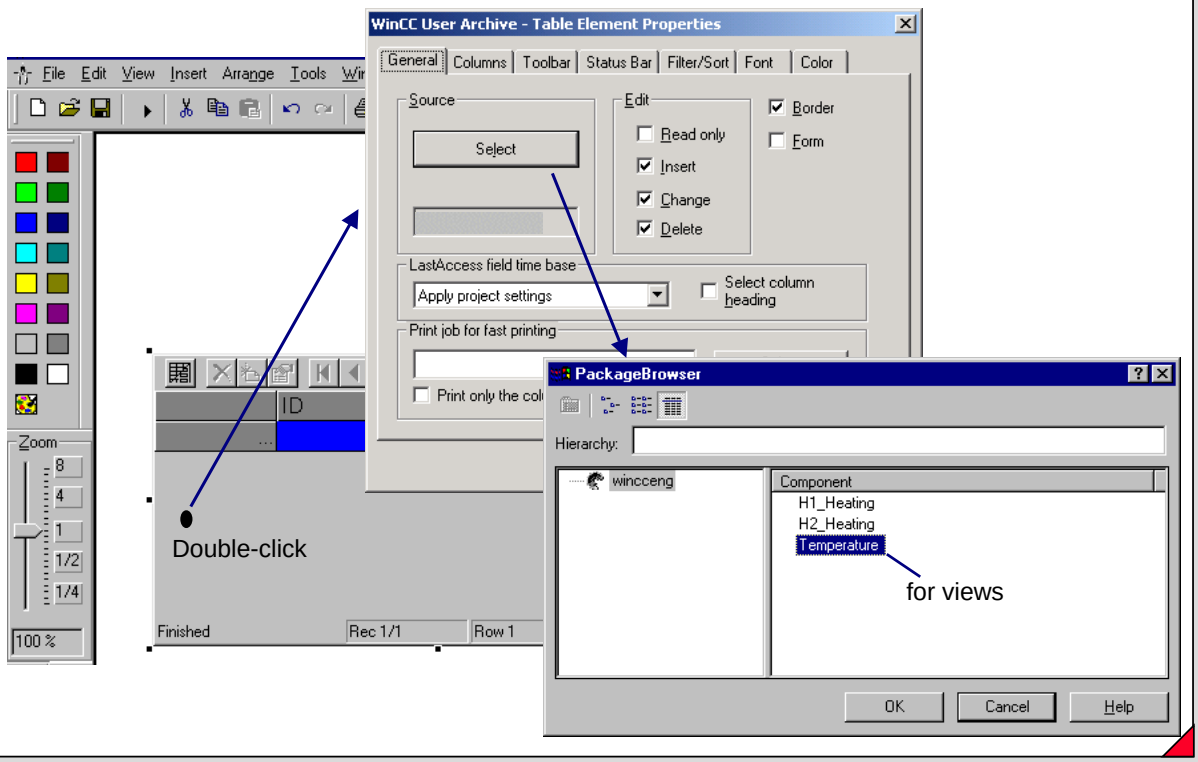
Note

Any changes you make to views only take effect in the database after **Saving**.

Table Window

You can switch on or switch off the table field by **clicking on Edit in the menu bar and then clicking on Runtime Data in the drop-down menu.**
This completes the entries in the User Archive Editor.

Exercise 4: Integrating a Control



SIMATIC HMI

Date: 27.10.2003
File: SWINCCS_07e.20

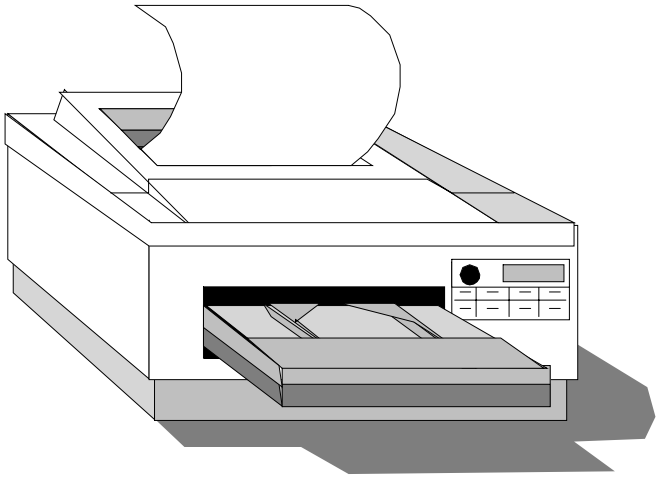
SITRAIN Training for Automation and Drives

Control A Control is created in the Graphics Designer in which the table element configured above is displayed (the procedure is displayed in the slide above).

Note In order to specify the properties of the “WinCC User Archive Table Element”, you must **double-click** within the Control area. Here you determine the view in the Graphic screen.
In addition, you can also set the frame, form, toolbar, status bar etc.

Runtime Test the function in Runtime operation.

Report System (Report Designer)



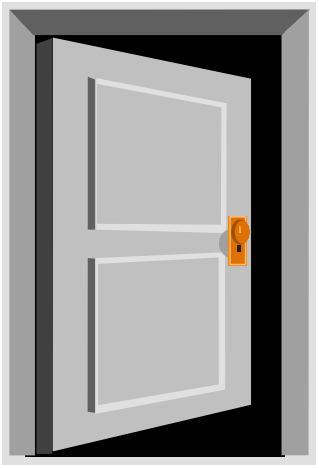
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_08e.1

SITRAIN Training for Automation and Drives

Contents Page

Objectives



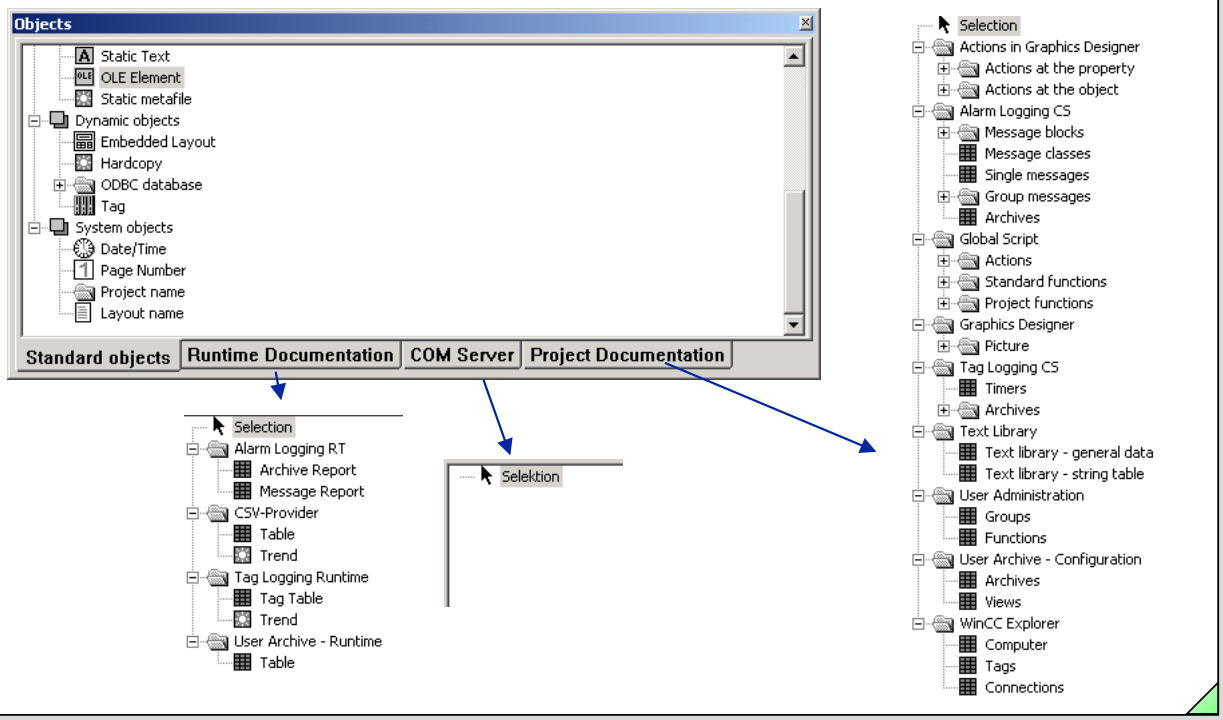
- Message sequence report
- Message archive report
- Measured value report
- Screenshot picture area (x,y)
- CSV Provider
- Feedback documentation
- User report

Date: 27.10.2003
File: SWINCC_08e.2

⇨ **Upon completion of this chapter, the participant should be familiar with:**

- the advantages of the Report Designer.
- a "Message Sequence Report"
- a "Message Archive Report"
- a "Measured Value Report"
- a "Screenshot Report"
- a "CSV Provider Report"
- the "Feedback Documentation"
- a "User Report"

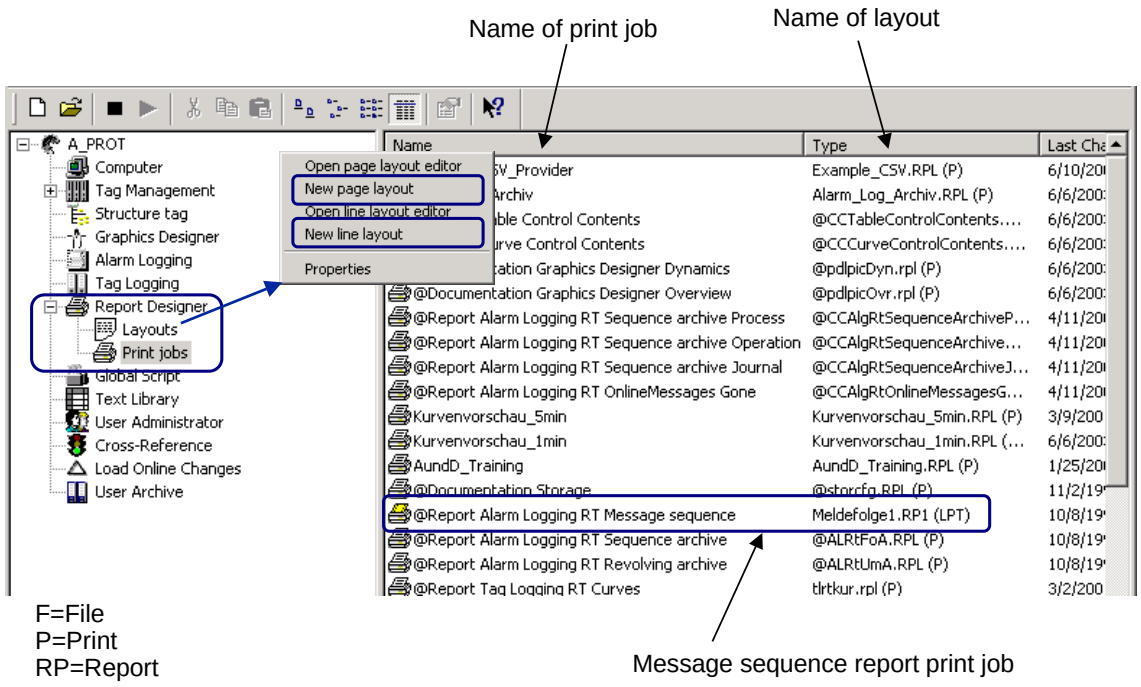
Object Palette of the Report Designer



General

You can create the report objects for the necessary reports using the Report Designer's object palette. These report objects exist, in part, in the many standard layouts for the direct print job.

Report Designer



General Information

The Report Designer is a component of the WinCC basic package and offers functions for creation and output of reports.

"Creation" in this description refers to creating report layouts. "Output" refers to printing these reports. During the output, the configured wildcards are dynamically filled (supplied) with the relevant data.

You have two editors available for creating the reports:

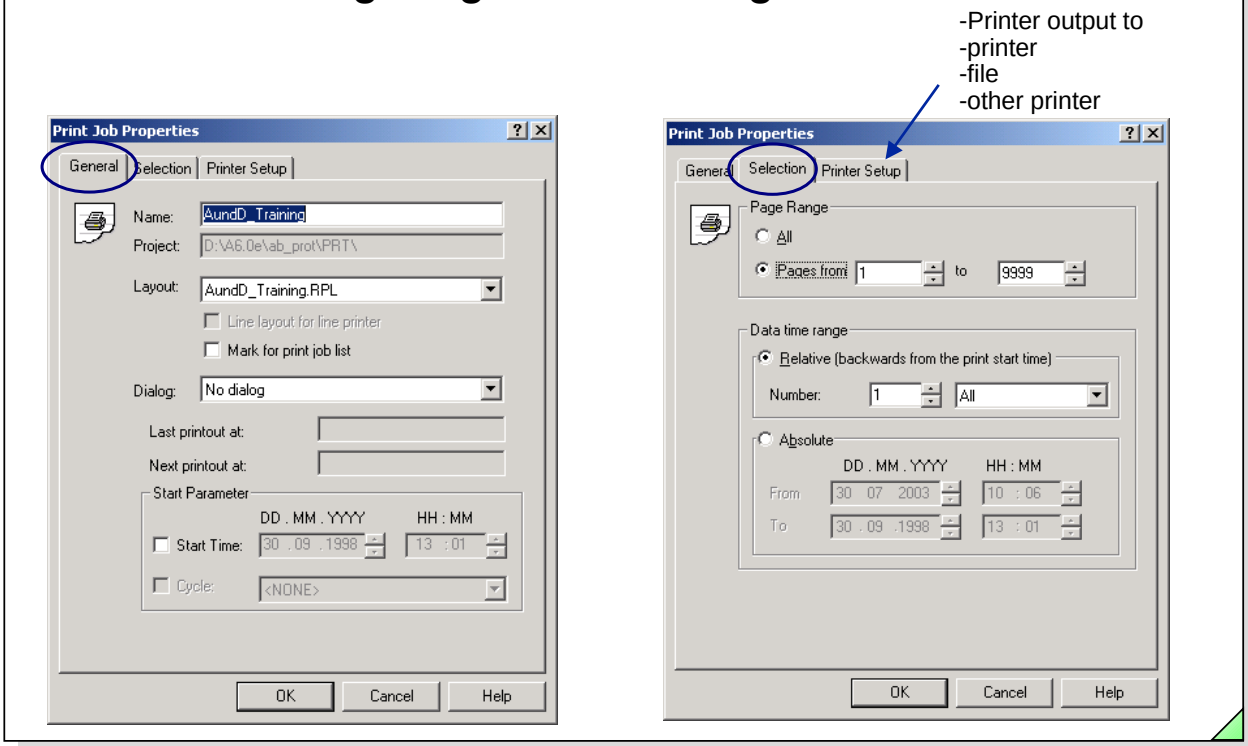
- Editor for page layouts
- Editor for line layouts (message Sequence reports)

There is one entry available per application in the selection dialog for the definition of the data (configuration) to be output in the reports:

- Documentation Graphics Designer — Feedback doc. of the Graphics Config.
- Documentation Tag Logging — Feedb. doc. of the Trends Config.
- Documentation Text Library — Feedb. doc. of the Message Text Blocks
- Documentation Control Center — Feedb. doc. of the project (such as the tag list)
- Documentation Global Script — Feedback doc. of the Action Config.
- Documentation Alarm Logging — Feedb. doc. of the Message Config.
- Storage — Feedb. doc. of the Storage Config.
- Report Alarm Logging Runtime — Feedback doc. Message Reports
  Tag Logging Runtime Curves — Feedback doc. of the Trends
  -
  -
  -

You use the dynamic objects in the Report Designer for the data output. These dynamic objects must be linked with the relevant applications.

Configuring and Selecting a Print Job



Date: 27.10.2003
File: SWINCC_08e.5

Print Output

The data selected for the task depends on the application. You select data when you create the layout, carry out the print job, or trigger printing.

Not all applications offer all three possibilities for selecting data.

The dynamic objects are supplied with the current values during the report output.

Static objects and system objects are available for the visual layout of the reports.

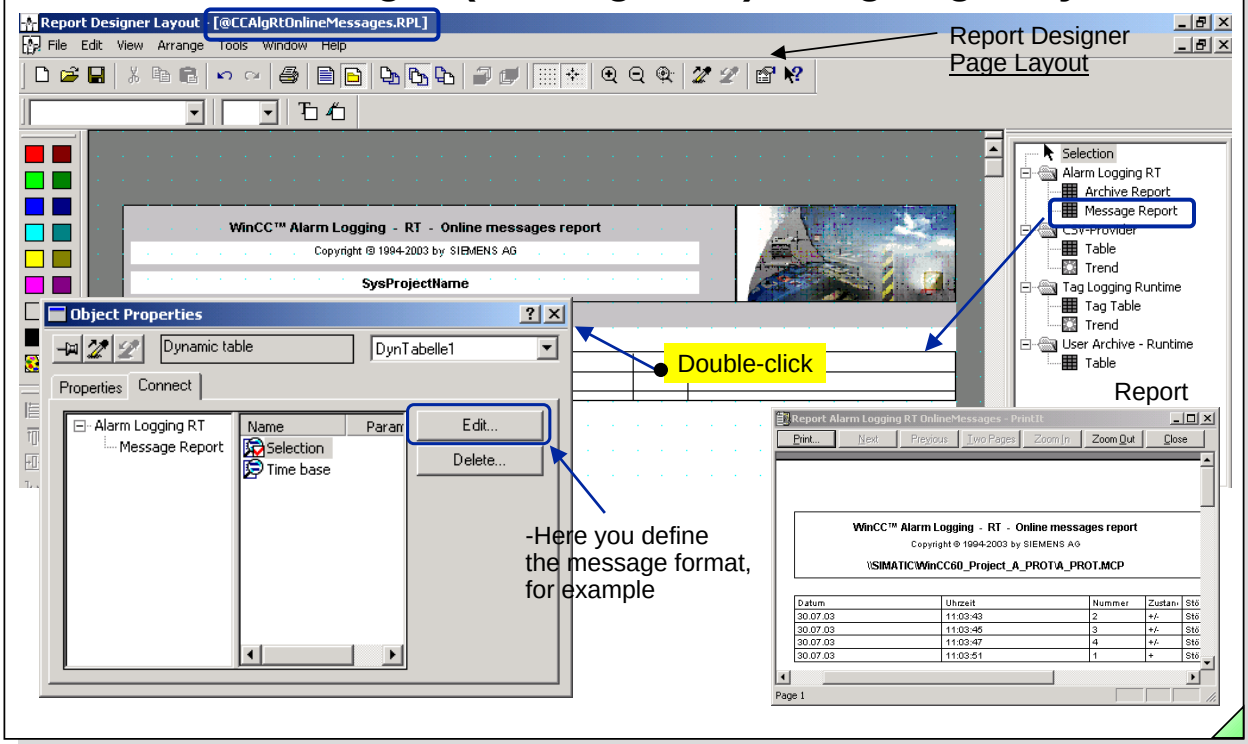
The time of the print output and the output medium must be specified for the output of the reports. The system offers the following selection possibilities:
- Operator-triggered printing
- Printout at a preselected time
- Cyclic output
- Output on the screen
- Output to a preselected printer
- Output to a file (*.emf format)

Note

You can directly use or copy the provided system layouts (all system files always begin with the @ character) and then adapt the layouts for your requirements.

You can use the WinCC Documentation Viewer to view and also print out the files in the *.emf format. You can find the WinCC Documentation Viewer on the WinCC installation CD under Tools.

Online Messages (Message List) using Page Layout



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_08e.6

SITRAIN Training for Automation and Drives

Task

The incoming messages are to be reported. You can use the message sequence report or the message list for this purpose. There are two types of message reporting:

1. Online Messages with output by pages, see slide above
2. Message sequence report output by lines, see next page

Preparation

You must also be sure that “This message will be archived” (default setting) is selected under Message Parameters for the single report in the Alarm Logging Editor.

Report Designer

The Report Designer contains a complete page layout and a print job with parameters assigned for the message sequence report:

- Page layout: @ **CCAlgRtOnlineMessages.rpl**
- Print job: @ **Report Alarm Logging RT OnlineMessages**

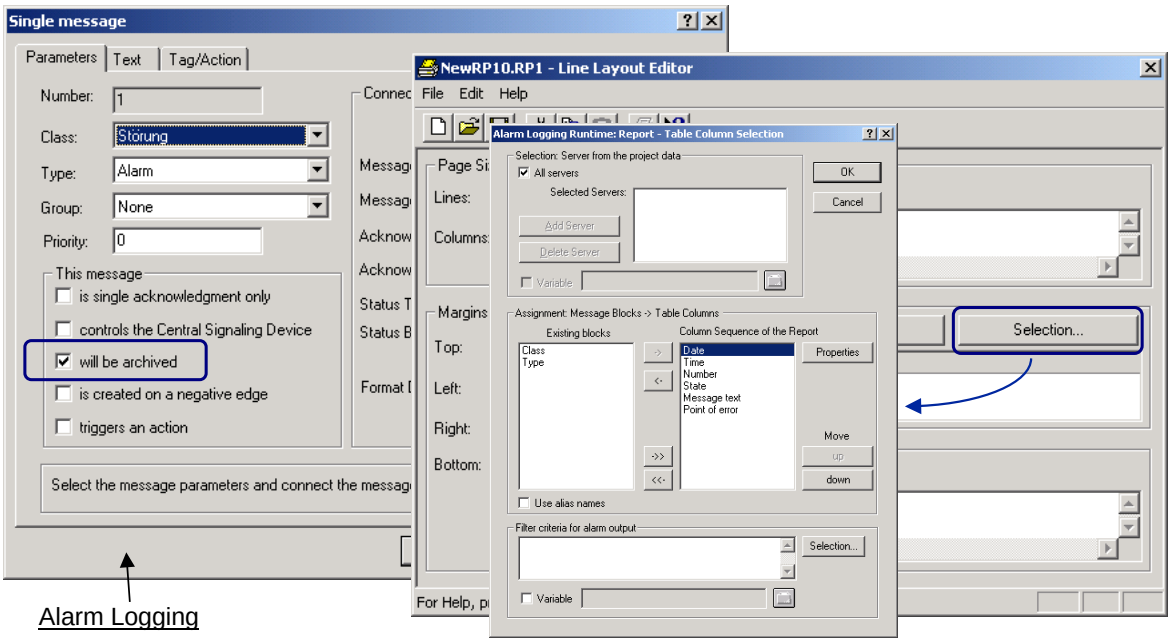
3. Now you only have to adapt the page layout to your message system. You use the Report Designer to edit the dynamic table. You use Object Properties/Connect/Selection/Edit to edit the message blocks to be output.
4. For the print job, for example, @ Report Alarm Logging RT OnlineMessages, you must select the appropriate layout - @CCAlgRtOnlineMessages.rpl.

Online Messages Report

When you trigger the print job @Report alarm Logging RT OnlineMessages, the report shown above is output.

Message Sequence Report using Line Layout

Usable matrix printers: all Epson LQ printers
-all HP printers (Deskjet, Laserjet, Paintjet), all Cannon BJC printers



Note: The "Message Sequence Report" must be entered in the computer startup list!

Task

The incoming messages are to be reported. You can use the message sequence report or the message list for this purpose. There are two types of message reporting:

1. Online Messages with output by pages, see slide above
2. Message sequence report output by lines, see next page

Report Designer

You create a line layout in the Report Designer and link it to the print job for the message sequence report:

- Line layout: **must be created by you!** For example, xyz.rpl.
- Print job: **@ Report Alarm Logging RT Message sequence**

3. The line layout must created once more and adapted to your message system.
4. For the print job, for example, @ Report Alarm Logging RT Message sequence, you must select the appropriate layout - xyz.rpl (they must be linked).

Message Sequence Report

When you activate the project (starting the RUNTIME mode), you also automatically start the message sequence report.

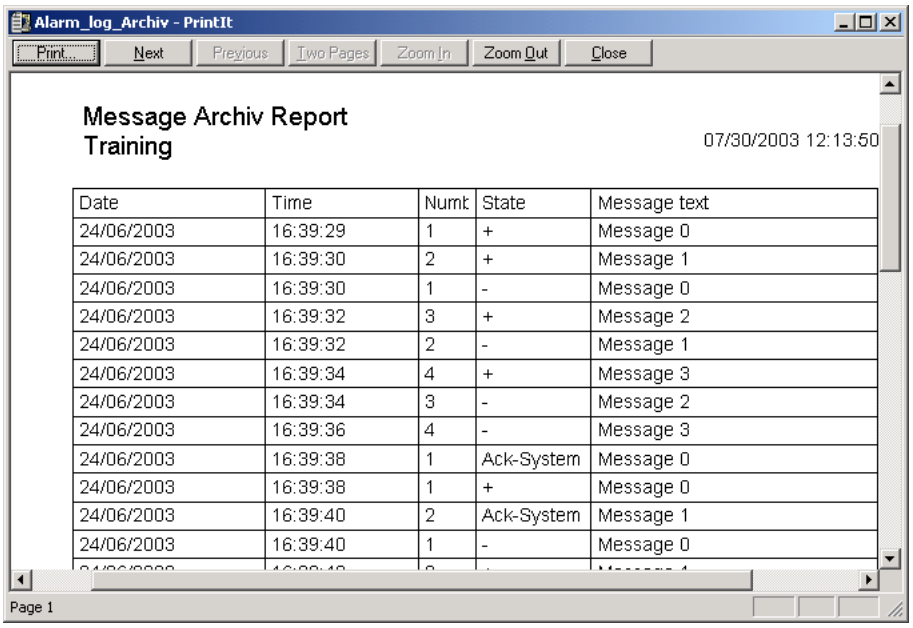
Note

The message sequence report does not have an "Off" button. The message sequence report runs either until it is removed from the message system or until the project is deactivated. You can, however, switch the report on and off (>=V4) with a mouse click in Runtime mode by triggering the function MSRTActivateMProt(TRUE,NULL); MSRTActivateMProt(FALSE,NULL).

For example: HP printers only print the message sequence report when a complete page of current messages was acquired. Special page printers report the current message immediately.

The line printer must be connected to the LPT1 parallel interface.

Message Archive Report 1/2



Date	Time	Numb	State	Message text
24/06/2003	16:39:29	1	+	Message 0
24/06/2003	16:39:30	2	+	Message 1
24/06/2003	16:39:30	1	-	Message 0
24/06/2003	16:39:32	3	+	Message 2
24/06/2003	16:39:32	2	-	Message 1
24/06/2003	16:39:34	4	+	Message 3
24/06/2003	16:39:34	3	-	Message 2
24/06/2003	16:39:36	4	-	Message 3
24/06/2003	16:39:38	1	Ack-System	Message 0
24/06/2003	16:39:38	1	+	Message 0
24/06/2003	16:39:40	2	Ack-System	Message 1
24/06/2003	16:39:40	1	-	Message 0

Task

The messages from the message archive are to be reported. You use the message archive report for this purpose. (You see the print preview in the screen in the slide above.)

Preparation

You must also be sure that “This message will be archived” (default setting) is selected under Message Parameters for the single report.

Report Designer

In the Report Designer, you have to create a page layout with the associated print job.

- Page layout: for example, Alarm_Log_Archive.rpl
- Print job: for example, Alarm_Log_Archive

The print job, for example, Alarm_Log_Archive must be linked with the appropriate layout, for example, Alarm_Log_Archive.rpl.

Message Archive Report

You can trigger the message archive report in different ways
- through the message window‘s toolbar: Icon 

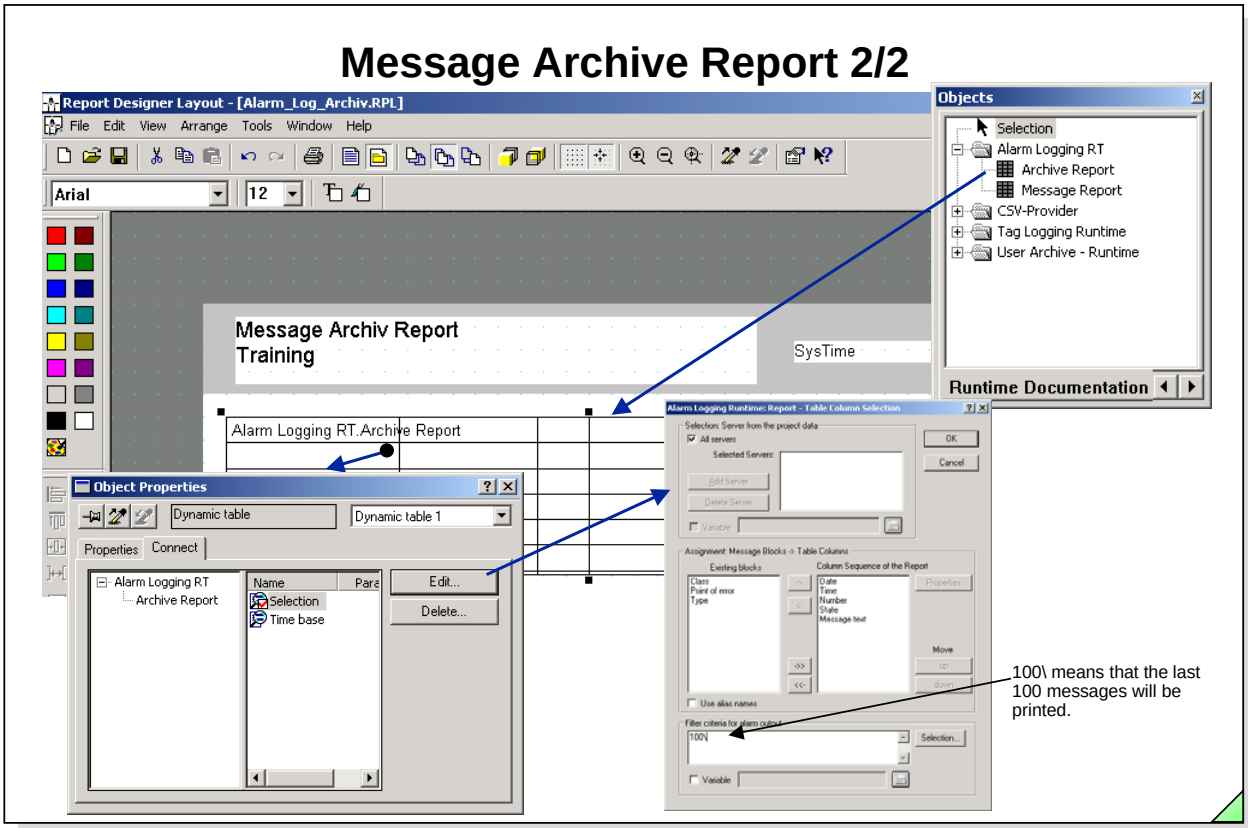
- through the Control Center: shortcut menu Report Designer/ trigger Print Job.
- using different trigger settings at a specific time or even cyclically.
- through the Runtime Application window All jobs shortcut menu.

Note

You select the printed messages either from the current display in the message window 

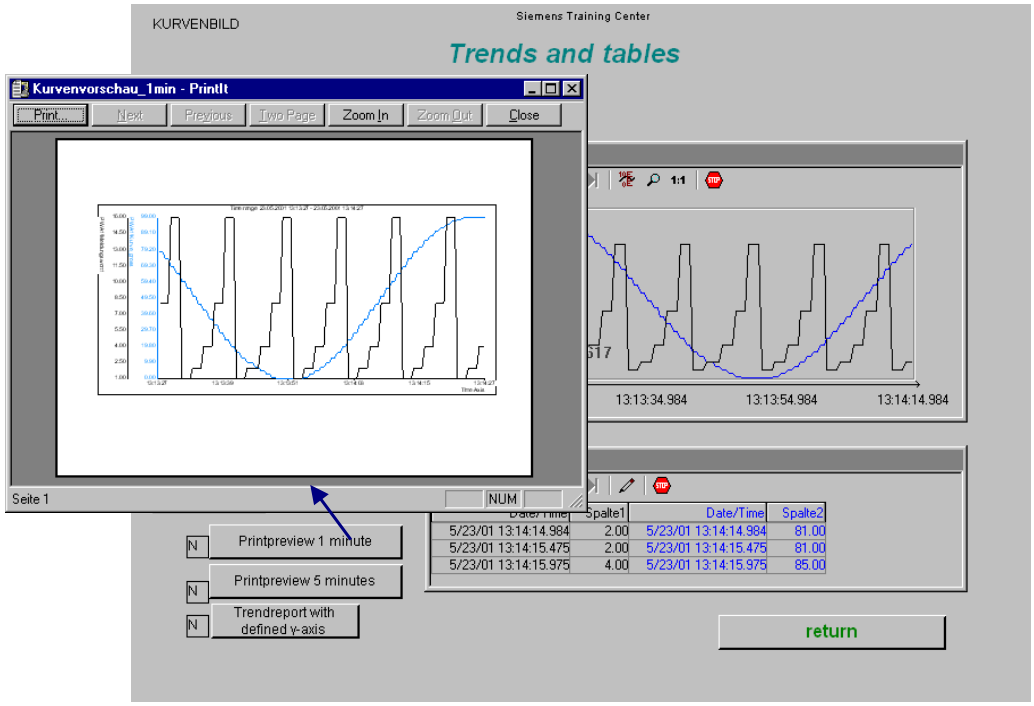
or you select a time period, or the number of messages (the same selection criteria as in Alarm Control can be used here). All selections can be made dynamic using a tag. You can also redirect the printout to be in a file or on the screen.

Message Archive Report 2/2



Print Job After you create the layout, the print job is set up.

Measured Value Report (Trends) 1/4



Date: 27.10.2003
File: SWINCC_08e.10

Task

The measured values in the measured value archive are to be reported. You use the measured value archive report for this purpose. (You see the print preview in the screen in the slide above.)

Report Designer

You are to create a page layout with the Report Designer as well as a print job with parameters assigned for the message value archive report:

- Page layout: Kurvenvorschau_1min.rpl
- Print job: AundD_Training2

1. The creation of the page layout is shown on Page 10.
   You are also to edit the dynamic metafile in the Report Designer.
2. For the print job, for example, AundD_Training2, you must select the appropriate layout - Kurvenvorschau_1min.rpl (they must be linked).

Measured Value Report

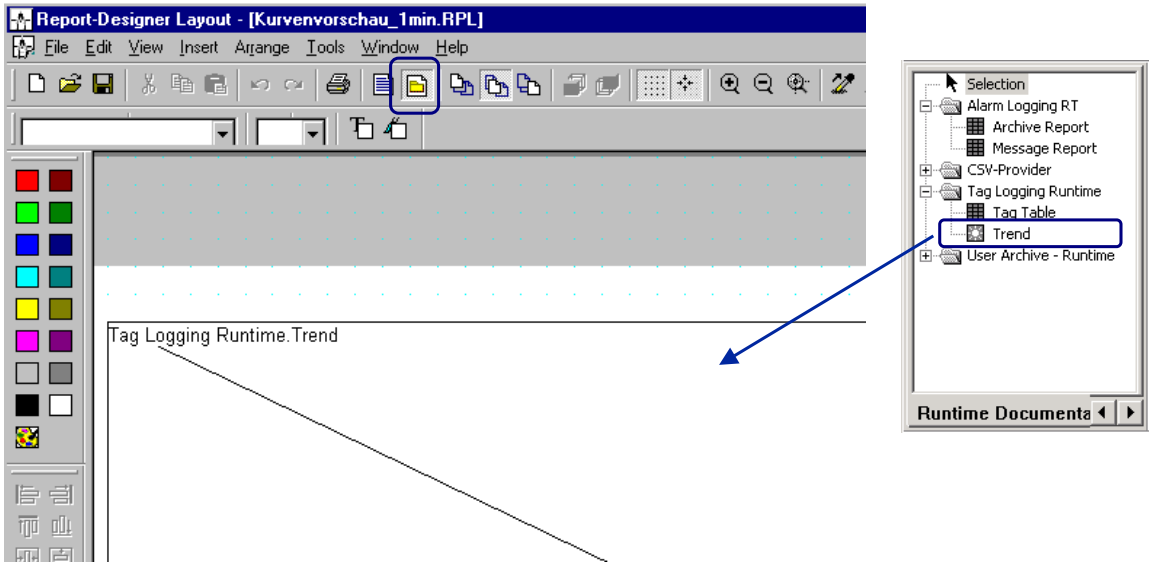
You can trigger the measured value archive report in different ways:

3.1 With the WinCC Explorer: trigger shortcut menu Report Designer/ Print job.
3.2 With various trigger settings at a certain time or even cyclically.
3.3 With the Runtime Application window All jobs shortcut menu.
3.4 With an Event Mouse and calling the function
RPTJobPreview("AundD_Training2"); or
RPTJobPrint("AundD_Training2");

Note

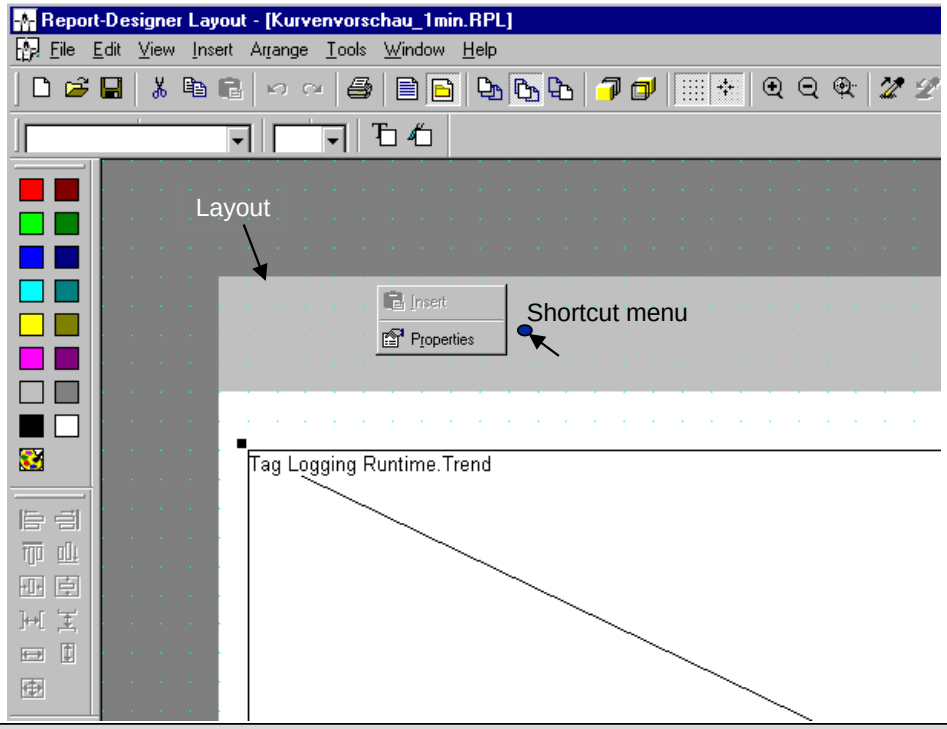
You select the printed measured values by selecting a time period from the WinCC Explorer. You can also redirect the printout to be in a file or on the screen.

Measured Value Report (Trends) 2/4

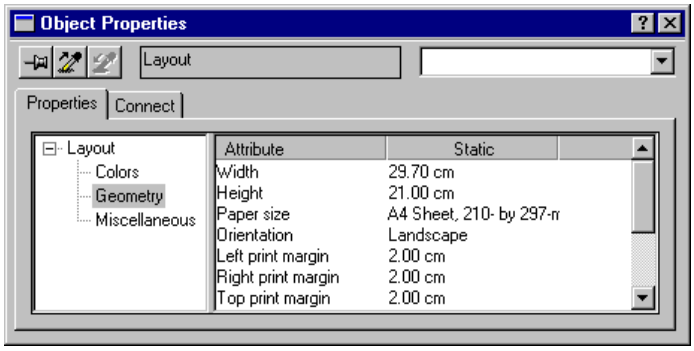


Date: 27.10.2003
File: SWINCC_08e.11

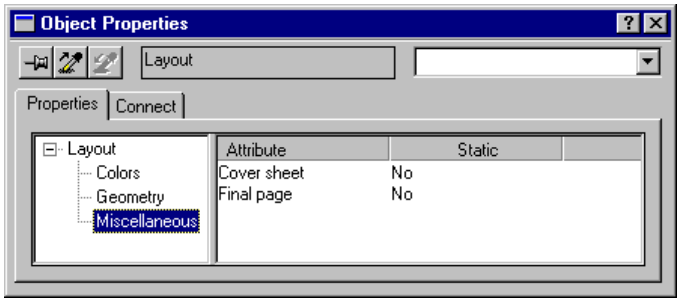
Measured Value Report (Trends) 3/4



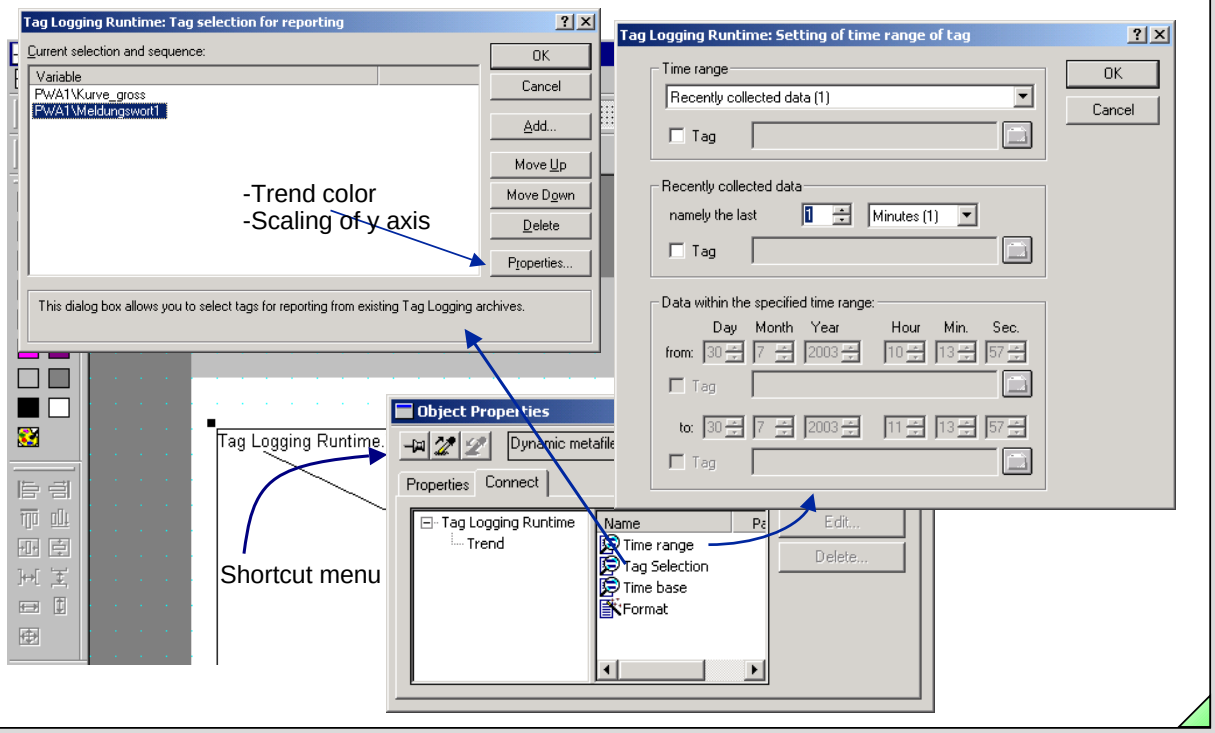
You use the Landscape setting for a horizontal format.



You can also print the report with a cover sheet and a final page.



Measured Value Report (Trends) 4/4

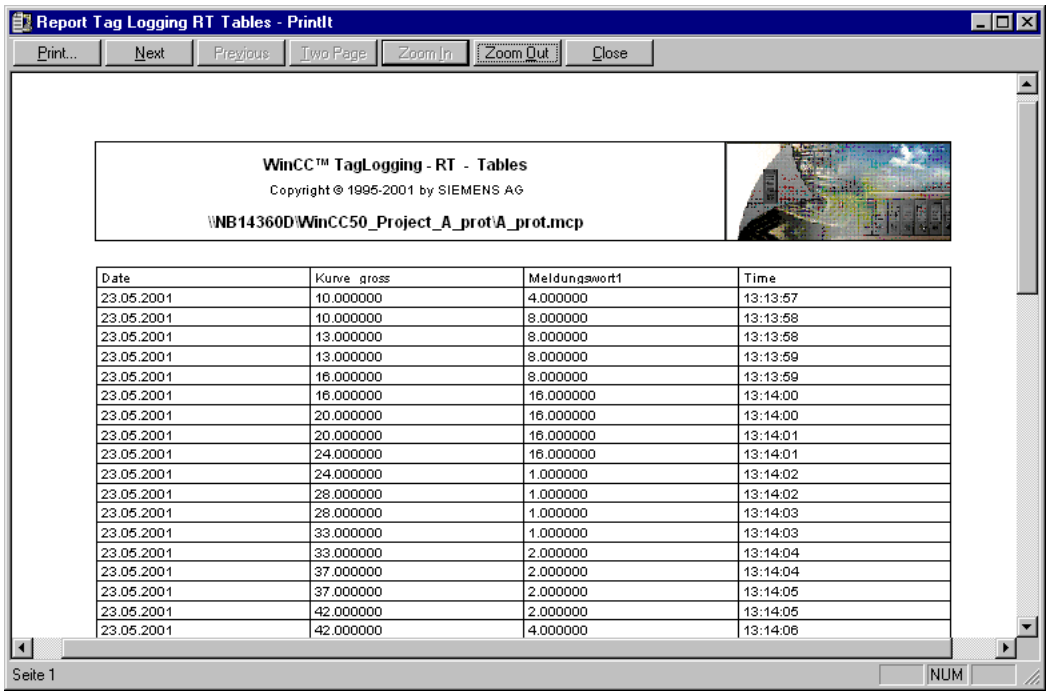


Date: 27.10.2003
File: SWINCC_08e.13

First of all you must connect (link) the archive. Then you can edit the time range, the archive tags, the format for the trends and the fonts.

The y axis labeling is displayed either fixed or dynamic.

Measured Value Report (Tables)



Date	Kurve_gross	Meldungswort1	Time
23.05.2001	10.000000	4.000000	13:13:57
23.05.2001	10.000000	8.000000	13:13:58
23.05.2001	13.000000	8.000000	13:13:58
23.05.2001	13.000000	8.000000	13:13:59
23.05.2001	16.000000	8.000000	13:13:59
23.05.2001	16.000000	16.000000	13:14:00
23.05.2001	20.000000	16.000000	13:14:00
23.05.2001	20.000000	16.000000	13:14:01
23.05.2001	24.000000	16.000000	13:14:01
23.05.2001	24.000000	1.000000	13:14:02
23.05.2001	28.000000	1.000000	13:14:02
23.05.2001	28.000000	1.000000	13:14:03
23.05.2001	33.000000	1.000000	13:14:03
23.05.2001	33.000000	2.000000	13:14:04
23.05.2001	37.000000	2.000000	13:14:04
23.05.2001	37.000000	2.000000	13:14:05
23.05.2001	42.000000	2.000000	13:14:05
23.05.2001	42.000000	4.000000	13:14:06

Date: 27.10.2003
File: SWINCC_08e.14

Task

The measured values in the measured value archive are to be reported. You need the measured value archive report for this purpose. (You see the print preview in the screen in the slide above.)

Report Designer

The Report Designer contains a complete page layout as well as a print job with parameters assigned for the measured value archive report.

- Page layout: **@CCTlgRtTables.rpl**
- Print job: **@ Report Tag Logging RT New**

\+ You now only have to adapt the page layout to your measured value archive. You edit the dynamic table in the Report Designer. You use Object Properties /Connect /Tags/Edit to edit the measured values you want to output and the fields for the date and time-of-day.

Note

The standard print job is already linked to the standard report layout.

Measured Value Report

You can trigger the measured value archive report in different ways:

\+ With the WinCC Explorer: trigger shortcut menu Report Designer/ Print job.
\+ With various trigger settings at a certain time or even cyclically.
\+ With the Runtime Application window All jobs shortcut menu.

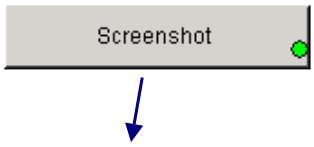
Note

You select the printed measured values either from the current display in the Control 

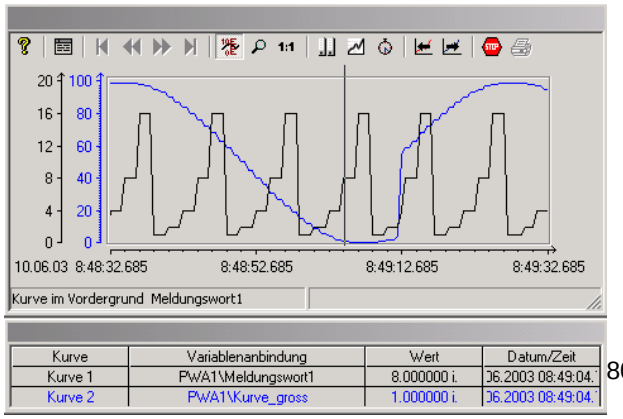
or you select a time period. All selections can be made dynamic using a tag. You can also redirect the printout to be in a file or on the screen.

.

Screenshot of a Picture Area




```
{
#pragma code("prtscrn.dll")
BOOL PrtScreenPart (ULONG left, ULONG top,
ULONG width, ULONG height);
#pragma code()

ULONG ObenX,ObenY,Breite,Hoehe;

ObenX=GetLeft(lpszPictureName,"Control1");
ObenY=GetTop(lpszPictureName,"Control1");
Breite=GetWidth(lpszPictureName,"Control1");
Hoehe=GetHeight(lpszPictureName,"Control1");

PrtScreenPart
(ObenX,ObenY,Breite,Hoehe+80);
}
```



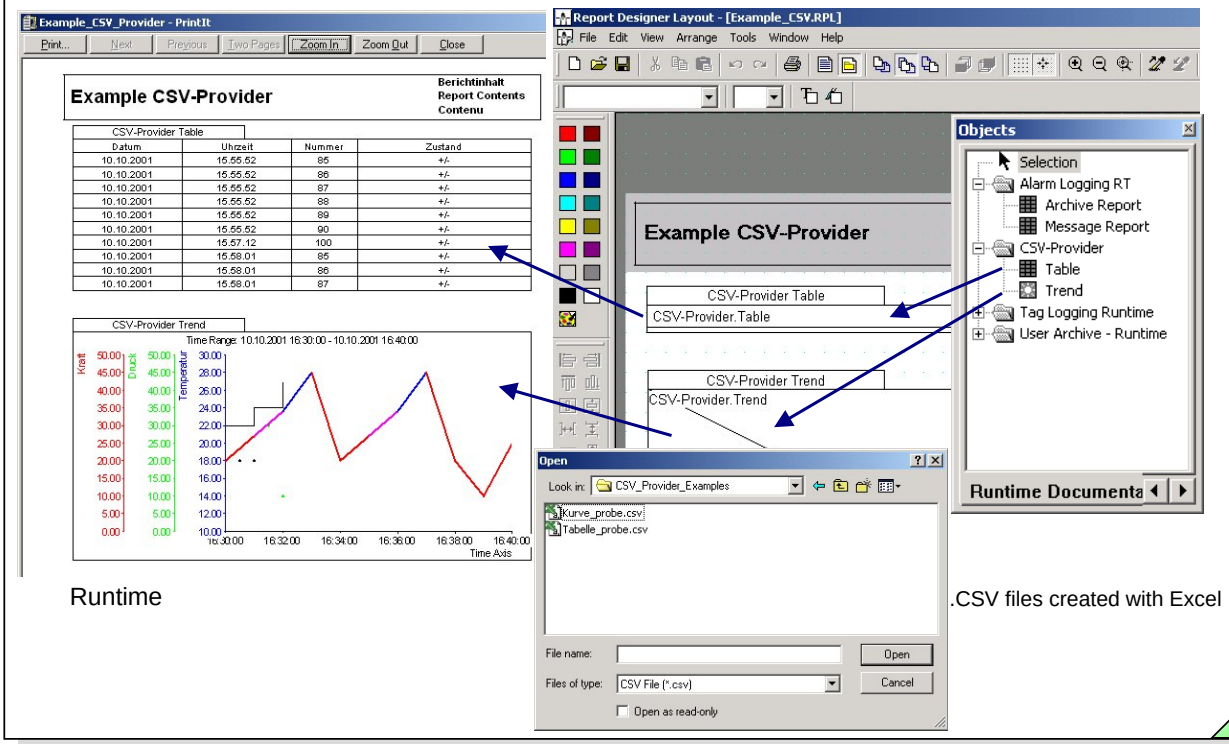
80

Date: 27.10.2003
File: SWINCC_08e.15

With this application, you can define a picture area for the printout in a graphic picture.
The PrtSreenPart function is supplied with the Size and Position object parameters. This can then be made dynamic using internal tags.

Note PrtScreenPart cannot be seen on the screen, it can only be printed out.

CSV Provider



Runtime

.CSV files created with Excel

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_08e.16

SITRAIN Training for Automation and Drives

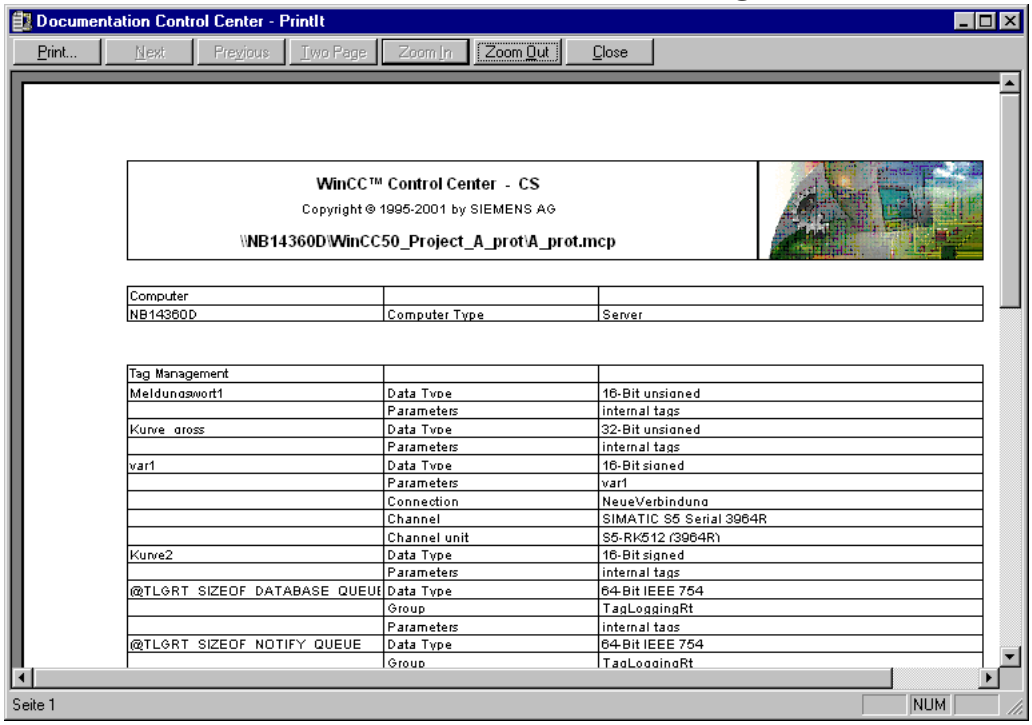
Task

The measured values, from laboratory or plant samples, for example, are to be prepared for a report.

Report Designer

1. The measured value samples are to be entered in a .CSV format using Excel. (the structure of a CSV file is described in the WinCC IS Help, Index CSV file, Requirements).
2. Create a Report Layout f.e. Example_csv.rpl.
3. Edit the Runtime Document CSV Provider, Table and Trends from the Object palette.
4. Link the two dynamic objects, for example, with the Kurve_probe.csv and Tabelle_probe.csv file.
5. Create a print job and link the print job to the layout: Example_csv.rpl.
6. View the print job on the screen with Print Preview and print it out.

Feedback Documentation of the Configuration Data 1/2



Computer		
NB14360D	Computer Type	Server

Tag Management		
Meldungswort1	Data Type	16-Bit unsigned
	Parameters	internal tags
Kurve_gross	Data Type	32-Bit unsigned
	Parameters	internal tags
var1	Data Type	16-Bit signed
	Parameters	var1
	Connection	NeueVerbindung
	Channel	SIMATIC S5 Serial 3964R
	Channel unit	S5-RK512 (3964R)
Kurve2	Data Type	16-Bit signed
	Parameters	internal tags
@TLGRT_SIZEOF_DATABASE_QUEUE	Data Type	64-Bit IEEE 754
	Group	TagLoggingRt
	Parameters	internal tags
@TLGRT_SIZEOF_NOTIFY_QUEUE	Data Type	64-Bit IEEE 754
	Group	TagLoggingRt

Date: 27.10.2003
File: SWINCC_08e.17

Task

You are to document the configuration data from the various editors. You have different standard layouts and print jobs available.

Print Job:	**Page Layout:**
@Documentation Control Center	@mcpcs.rpl
@User 1	@Control Center CS (compact).rpl
@Documentation Graphics Designer	@pdlpic.rpl
@Documentation Graphics Designer Dynamics	@PDLPicDyn.rpl
@Documentation Global Script Project function	@gsc_rpfc.rpl
@Documentation Global Script Standard function	@gsc_rsfc.rpl
@Documentation Global Script Actions	@gsc_ract.rpl
@Documentation Tag Logging	@tlgcs.rpl
@Documentation Alarm Logging	@algcs.rpl

-
-
-

Report Designer

1. You can adapt the respective page layout as you want to have it. You edit the static and dynamic components in the Report Designer and save the new page layout under another name, if applicable.
2. For the print job, for example, **@Documentation....**, you must select the appropriate layout - **@.....rpl** (they must be linked).

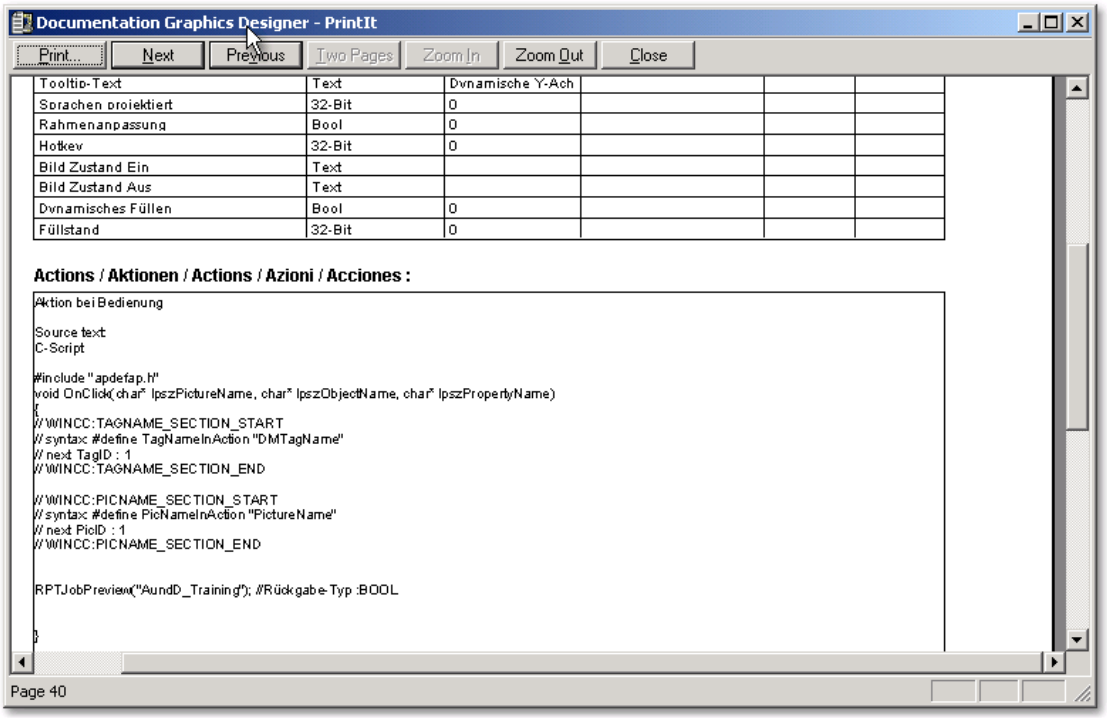
Feedback Documentation

You can trigger the feedback documentation in various ways.
3.1 With the WinCC Explorer: trigger shortcut menu Report Designer/Print job.
3.2 With various trigger settings at a certain time or even cyclically.

Note

You can also redirect the printout to be in a file or on the screen.

Feedback Documentation of the Configuration Data 2/2



Extension

In addition to the general feedback documentation functions in the Report Designer, you can also document the actions directly from the Editor. The functions in Report Designer are used to specifically document individual functions that were just changed or were newly created and were included (transferred into) in the project.
To start this type of documentation, there are internal print jobs and standard layouts available.

Print Job:	**Page Layout:**
@internal Graphics Designer Actions at the object	@akt_obj.rpl
@internal Graphics Designer Actions at property	@akt_prop.rpl
@internal Global Script Project function	@gsc_pcf.rpl
@internal Global Script Standard function	@gsc_sfc.rpl
@internal Global Script Actions	@gsc_act.rpl

Report Designer

1. You can adapt the respective page layout as you want to have it.
   You edit the static and dynamic components in the Report Designer and save the new page layout under another name, if applicable.
2. For the print job, for example, **@internal G....** , you must select the appropriate layout - **@.....rpl** (they must be linked).

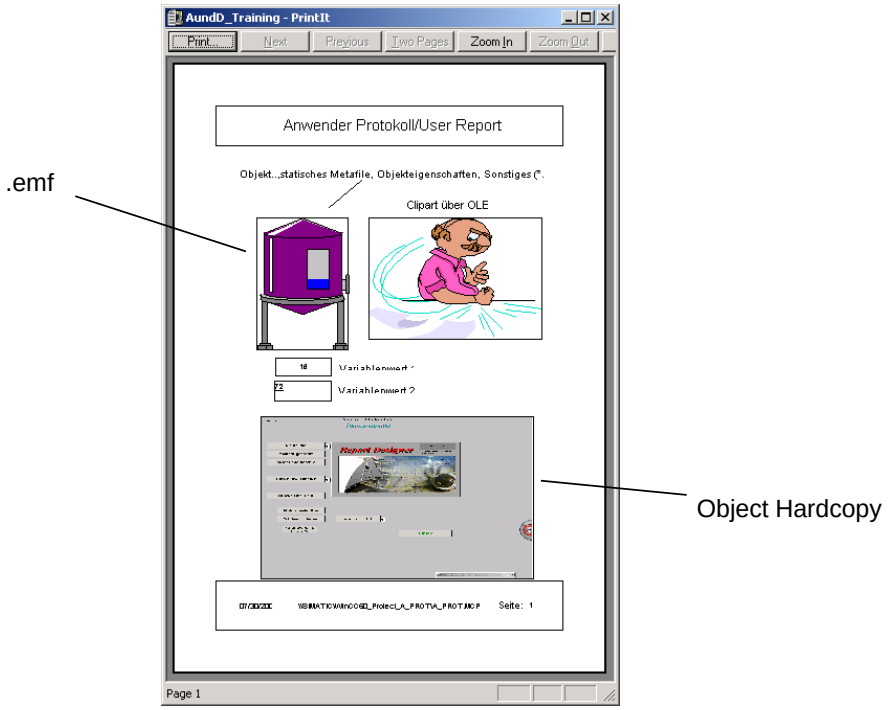
Feedback Documentation

1. You trigger the feedback documentation directly in the Editor that corresponds to the application.
2. You select the internal feedback documentation through the corresponding print job.

Note

You can also redirect the printout to be in a file or on the screen.

User Report - Print Preview

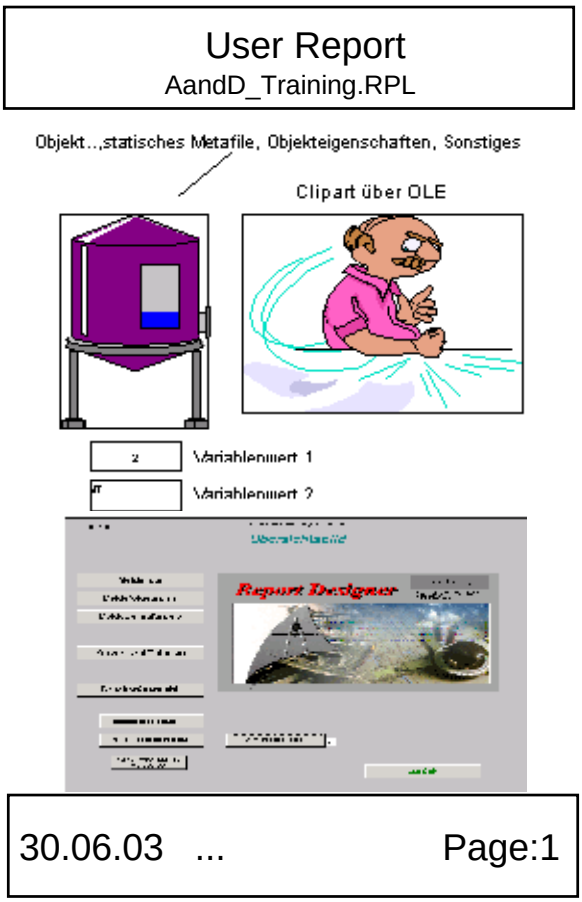


SIMATIC HMI

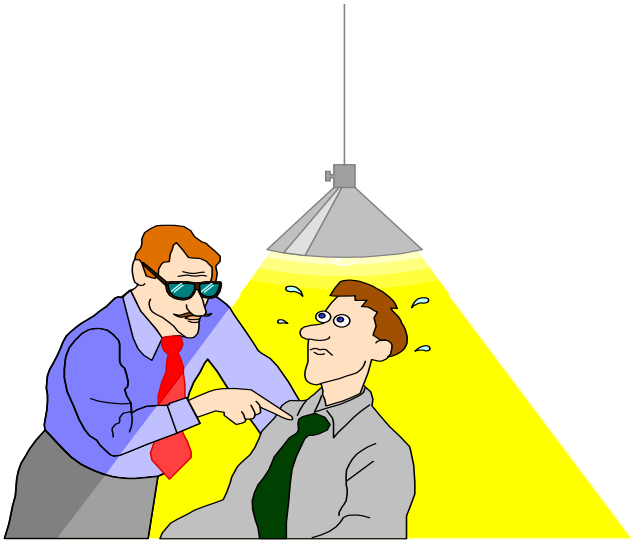
Date: 27.10.2003
File: SWINCC_08e.19

SITRAIN Training for Automation and Drives

Printout:



Background Processing (Global Script)



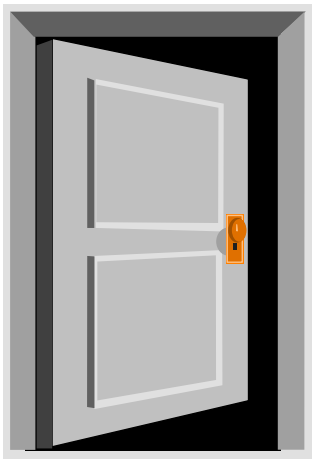
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_09e.1

SITRAIN Training for Automation and Drives

Contents Page

Objectives



- Project Functions
- Standard Functions
- Internal Functions
- Project-wide Actions

SIMATIC HMI

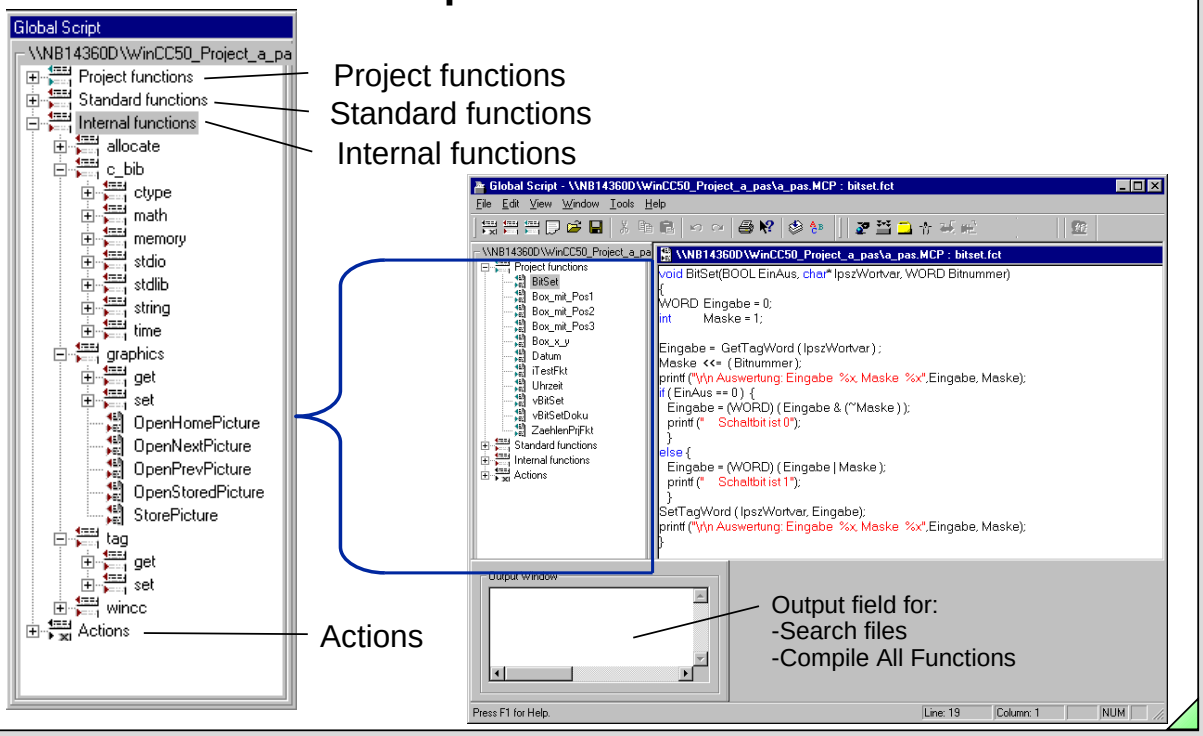
Date: 27.10.2003
File: SWINCC_09e.2

SITRAIN Training for Automation and Drives

⇨ **Upon completion of this chapter, the participant should be familiar with:**

- the advantages of the Global Script.
- "Project Functions", based on a C example.
- "Standard Functions", based on a C example.
- "Internal Functions", based on a C example.
- "Project-wide Actions", based on a C example.
- "Project Functions", based on a VBS example.

Global Script C Actions and Functions



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_09e.3

SITRAIN Training for Automation and Drives

General Information Global Script is the general term for C functions and actions. There is a series of C functions available for programming extensive actions in WinCC. These C functions are divided into groups and can be used both in the Graphics Designer as well as to a certain extent in Global Script.

Project Functions All project-specific functions are combined under this header. The functions are applicable within the project (function library).

Standard Functions This is where functions are located that can be used for more than one project. Pre-defined functions are present that can be expanded by your functions.

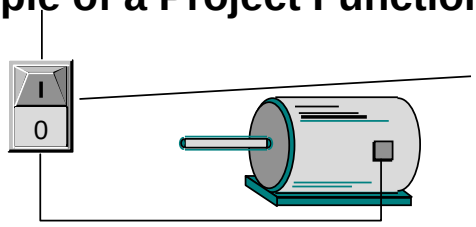
Internal Functions The system provides the functions listed here. The user can not change the internal functions. These functions contain all basic functions necessary for processing WinCC objects. These functions also include all the functions in the C standard library.

There is a subdivision into:

- alarm — Functions for receiving and processing individual messages.
- allocate — Functions for dynamic memory management (required if the return value is a character string, for example)
- c_bib — Standard C library functions
- graphics — Functions for reading and manipulating object properties of individual WinCC objects
- tag — Functions for reading and writing tags from the data manager
- wincc — Functions for influencing the individual WinCC Runtime components

Actions Actions are available only in the Global Script. This is where actions are defined that run in the background and are not connected directly with the selected WinCC object (screen, screen window, graphic object, etc.).

Example of a Project Function and/or Standard Function



Switching on a motor:
Events/ Mouse/ Left mouse button/ C function...
BitSet (1, "MotorStatus", 2);

/*Bit 2 is set in the “MotorStatus” tag, to reset the bit use: **BitSet** (0, “MotorStatus”, 2)*/

A function defined in the Global Script (Standard Functions:

```
void BitSet(BOOL OnOff, char* lpszWordtag, WORD wbitnumber)
{

WORD    winput = 0;                          // Create local tags
int     iscreen = 1;

winput =  GetTagWord ( lpszWordtag ) ;       // Read process tags
iscreen  <<=  ( wbitnumber );                // Create bit mask
if (OnOff == 0 )  {                          // Reset bit ?
  winput = (WORD) ( winput & (~iscreen) );   // Reset bit
  }
else {
  winput = (WORD) ( winput | iscreen );      // Set bit
  }
SetTagWord ( lpszWordtag, winput);           // Rewrite process tag
}
```

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_09e.4

SITRAIN Training for Automation and Drives

Task

A bit is to be reset in a data word. This function is to be stored centrally, because it is a function that is frequently used. This central storage can be either project-specific **(project functions)** or for more than one project **(standard functions)**.

Procedure

The function is created in Global Script after the method of functioning and the parameters to be transferred have been clarified. After the function has been saved in the project functions or standard functions, it can be used in the Graphics Designer at various locations and with various parameters. (Functions you write yourself can not be used in Global Script.)
The advantage of central storage is how easy it is to make changes and to maintain the function. Changes only have to be carried out at one location and are immediately available at all call-up locations (as long as the transfer parameters do not change).
Parameters are transferred in the C syntax.

Function Call

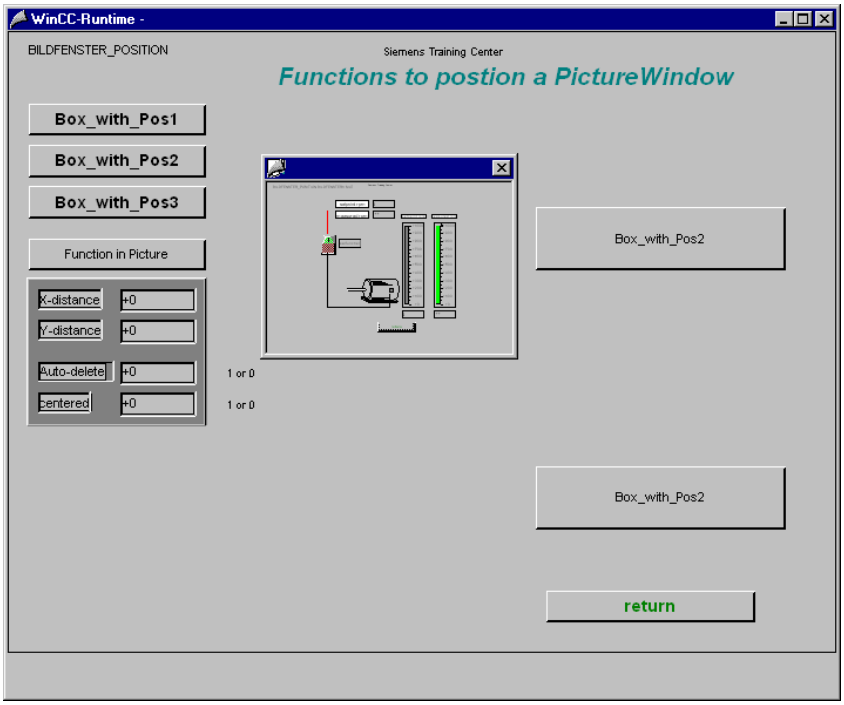
The call of the function named above has the following appearance in the Graphics Designer.
For example, Setting Bit 2 in a word:

Events/ mouse/ left mouse button/ C function...
BitSet (1, "MotorStatus", 2);

This sets Bit 2 in the "MotorStatus" tag. To reset the bit, you have to call the same function with the paramaters:

BitSet (0, "MotorStatus", 2);

Internal Functions: Example of Graphics 1/3



Date: 27.10.2003
File: SWINCC_09e.5

General Information

You can find functions for reading (Get) and manipulating (Set) object properties of individual WinCC objects under the internal **Graphics** functions. The functions are divided according to the object properties groups. Some functions can be used only with certain objects.

- Style — Styles (for example, line end and line weight)
- Flash — Flash (for example, background flashing color and flashing frequency)
- Fill — Fill (yes/no and fill level)
- Font — Font (for example, font, orientation, and writing style)
- Axes — Axes (only for bar graphs: axis labeling and axis division)
- Color — Colors (for example, foreground color and border color)
- I_O — Input/output (only for I/O fields: output values and input values)
- Geometry — Geometry (for example, polygon corners and X-Y coordinates)
- Limits — Limits (only for bar graphs and I/O fields: various limit values, etc.)
- Misc — Miscellaneous (for example, border adaptation, screen name, and visibility)

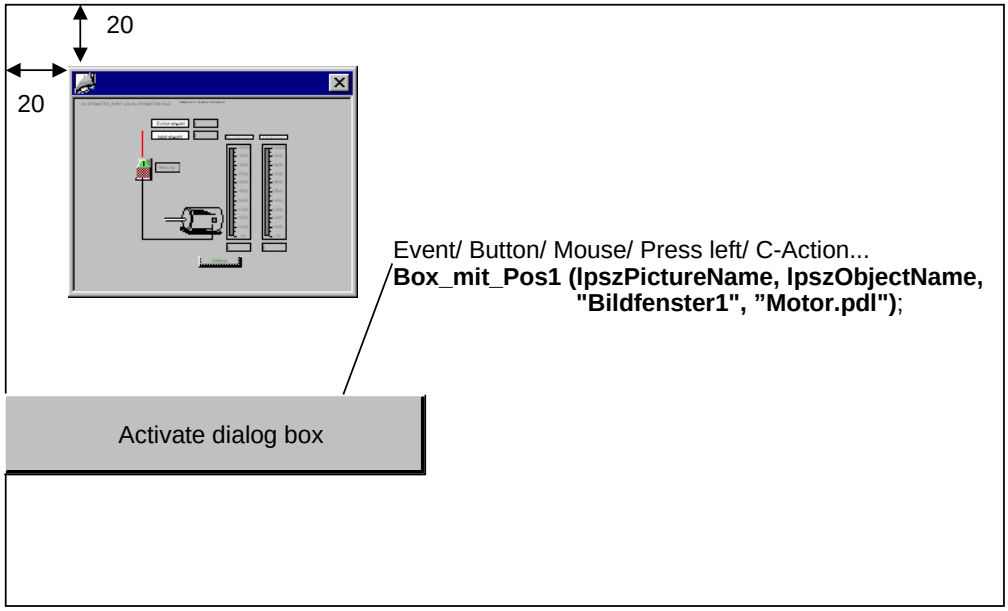
Example

A dialog box from which various buttons can be operated is to be called in one screen.
Example 1 shows the basic function for positioning and outputting the dialog box.
The function is expanded somewhat in Example 2, so that the box can always be output at the lower right corner of the button to permit easier assignment to the operated element.

Procedure

The geometry functions help to determine the button position. Then there is a check to determine whether the box fits to the right of and below the button. If this is not the case, then the box is output to the left of the button and above the lower screen edge. SetPictureName and SetVisible functions carry out the output.

Internal Functions: Example of Graphics 2/3



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_09e.6

SITRAIN Training for Automation and Drives

Example

Here you see the functions necessary for outputting the dialog box at a certain location on the screen. The function contains four transfer parameters.

- lpszPictureName Screen name for the current screen
- lpszObjectName Name of the operated object
- lpszBScreenName Name of the configured screen window
- lpszBoxName Name of the screen to be output in the specified picture window

The value for the left, upper corner of the dialog box must first be permanently specified in the present example.

Function Definition

```
void Box_with_Pos1(char* lpszPictureName, char* lpszObjectName,
                   char* lpszBScreenName, char* lpszBoxName)
{
```

Tag Declaration

```
int BScreenXPos, BScreenYPos;
BScreenXPos = 20;
BScreenYPos = 20;
```

Outputting the Box

```
SetLeft( lpszPictureName, lpszBScreenName, BScreenXPos);
SetTop( lpszPictureName, lpszBScreenName, BScreenYPos);
SetPictureName( lpszPictureName, lpszBScreenName, lpszBoxName);
SetVisible( lpszPictureName, lpszBScreenName, 1);
}
```

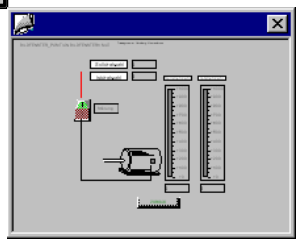
Note

There is a function that determines the position of the operated object and outputs the dialog box on the lower, right corner of the operated object on the next page

Internal Functions: Example of Graphics 3/3

Activate dialog box

Events/ Button/ Mouse/ Press left/ C action...
Box_mit_Pos2 (lpszPictureName, lpszObjectName, "Bildfenster1", "Motor.pdl");



You can place dialog screen windows where you want them on the screen. To establish a reference to the operated object, it is advantageous if the dialog box appears where it is used. You can attain this positioning with the following steps:

- Scanning the properties of the operated object and the entire screen
- Scanning the size and stretching the dialog window
- Specifying the appearance location of the dialog window (to the right or left next to the operated object)

These functions are combined in a standard function that can then be used from any location (each screen and operating object).

Function Definition

```
void Box_mit_Pos2(char* lpszPictureName, char* lpszObjectName,
                  char* lpszBScreenrName, char* lpszBoxName)
{
```

Tag Declaration

```
int ButtonXPos, ButtonYPos, ButtonWidth, ButtonHeight;
int ScreenWidth, ScreenHeight;
int BScreenXPos, BScreenYPos, BScreenWidth, BScreenHeight;
int XInterval=0, YInterval=0;
```

Fetching Current Data

```
ButtonXPos = GetLeft(lpszPictureName,lpszObjectName);
ButtonYPos = GetTop(lpszPictureName,lpszObjectName);
ButtonWidth = GetWidth(lpszPictureName,lpszObjectName);
ButtonHeight = GetHeight(lpszPictureName,lpszObjectName);
ScreenWidth = GetWidth(lpszPictureName,NULL);
ScreenHeight = GetHeight(lpszPictureName,NULL);
BScreenWidth = GetWidth(lpszPictureName,lpszBWindowName);
BWindowHeight = GetHeight(lpszPictureName,lpszBWindowName);
```

Output Data

```
BScreenXPos = ButtonXPos + ButtonWidth + XInterval;
BWindowYPos = ButtonYPos + ButtonHeight + YInterval;

if ((BScreenXPos + BScreenWidth) > ScreenWidth) {
    BScreenXPos = ButtonXPos - BScreenWidth - XInterval;
    }
if ((BScreenYPos + BScreenHeight) > ScreenHeight) {
    BScreenYPos = ScreenHeight - BScreenHeight;
    }
```

Outputting the Box

```
SetLeft(lpszPictureName,lpszBScreenName,BScreenXPos);
SetTop(lpszPictureName,lpszBScreenName,BScreenYPos);
SetPictureName(lpszPictureName,lpszBScreenName,lpszBoxName);
SetVisible(lpszPictureName,lpszBScreenName,1);
}
```

Background Processing (Actions) with Global Script

Operating hours.PAS

```
WORD  HOUR, MINUTE;
BYTE Z_ON;

Z_ON = GetTagByte ("Z_On");

if (Z_ON) {
   MINUTE = GetTagWord ("Minutes");
   MINUTE = MINUTE + 1;
   SetTagWord ("Minutes", MINUTE);

   HOUR = MINUTE / 60;
   SetTagWord ("operating hours", HOUR);
   }
return(HOUR);
```

Note: Global Script must be activated in the computer startup list

SIMATIC HMI

Date: 27.10.2003
File: SWINCC_09e.8

SITRAIN Training for Automation and Drives

General Information

You have a **P**roject-wide **Ac**tion**S**ystem (**PAS**) available in WinCC for actions independent of the screen. These actions run in the background and can be started cyclically (time trigger), acyclically (at a certain time), or even dependent on an event (tag). These actions serve to implement actions for an entire project. The individual actions are created in the Global Script Editor.

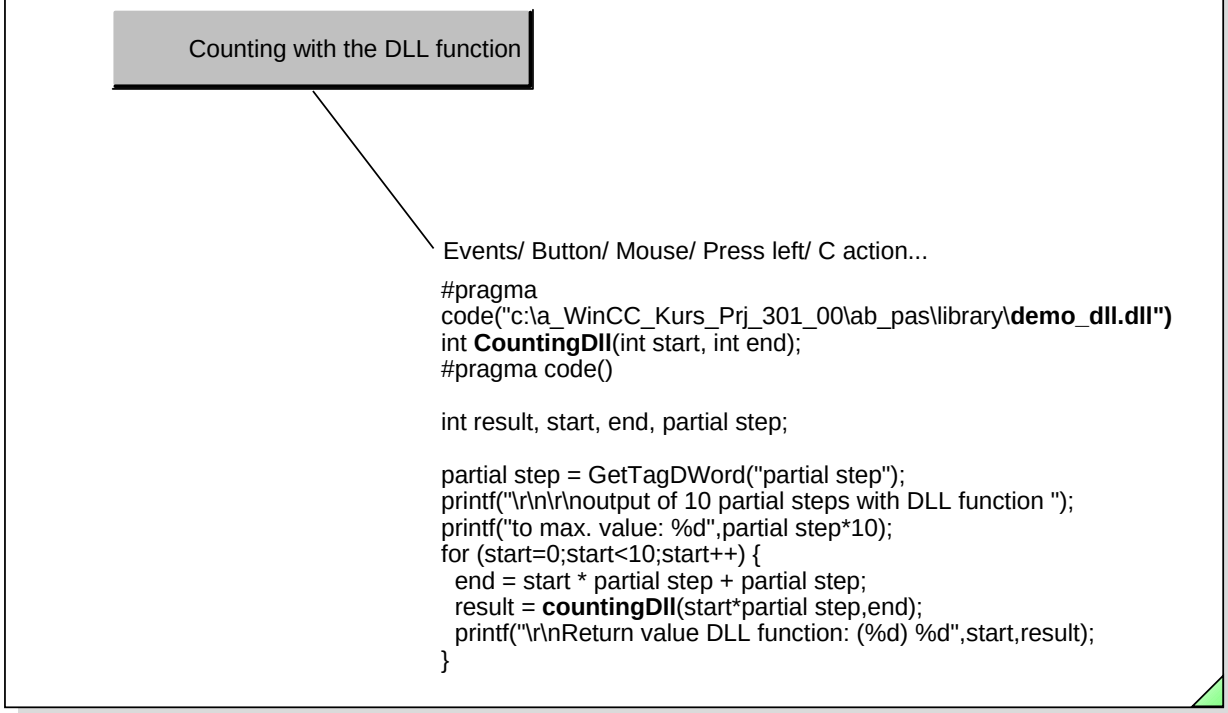
Example

An operating hours counter is to be implemented as an example. If a certain drive or even system component is working, an internal counter is incremented, which records the minutes and hours. A minute trigger starts the action. A start bit makes it possible to record the operating hours separately from the WinCC Runtime and store it in assigned tags.

Procedure

1. Create internal tags in the data manager.
2. Create an action in the Global Script Editor (for example, operating hours.PAS).
3. Select Global Script Runtime in the startup list.
4. Activate the Runtime.

Example with Functions from DLLs



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_09e.9

SITRAIN Training for Automation and Drives

General Information

Action programming from WinCC offers a multitude of generation possibilities. It is also possible to access external functions stored in DLLs, for example, within the C action. These DLLs can be both standard DLLs as well as function libraries you create yourself. Using a function from a DLL is especially recommended where extensive calculations must be carried out. Because the code is already stored in an executable way in the DLL and does not have to be interpreted at runtime, there is a significantly faster processing speed for extensive calculations.

Example

In the example above, nothing more is done for the DLL function than processing a loop from a start value to an end value. The intermediate results are output in a diagnostics window. In a direct comparison, you do not see a significant time advantage in using DLL functions until there are very many operations.

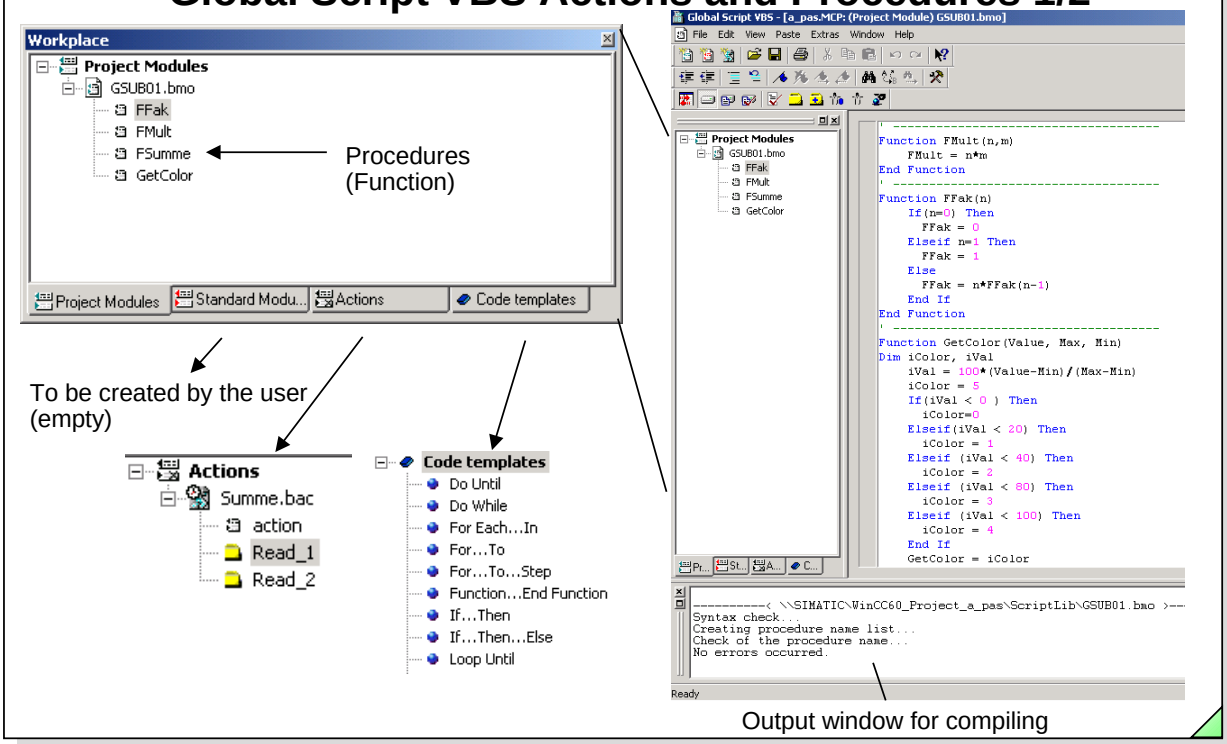
Note

To create the DLL, you can use the Application Wizard from Visual C++, for example (MFC-AppWizard (dll)). All functions to be used are entered in this DLL and are declared as external "C." In addition, each function must also be entered in the export table.

FAQ solution

FAQ 830 1801 Question: How can I implement a selfmade DLL in WinCC in the correct way?

Global Script VBS Actions and Procedures 1/2



Date: 27.10.2003
File: SWINCC_09e.10

General Information

Global Script is the general term for functions and actions with ANSI-C and VBS. There is a series of C functions available for programming extensive actions in WinCC. These C functions are divided into groups and can be used both in the Graphics Designer as well as to a certain extent in Global Script.

Project Modules

All project-specific procedures are grouped together under this tab. The procedures are only applicable within the project (ScriptLib Catalog).

Procedure Characteristics

+ You create and modify them.
+ They can be protected against change and viewing through a password.
+ They don't have a trigger.
+ They are saved in a module.

Standard Modules

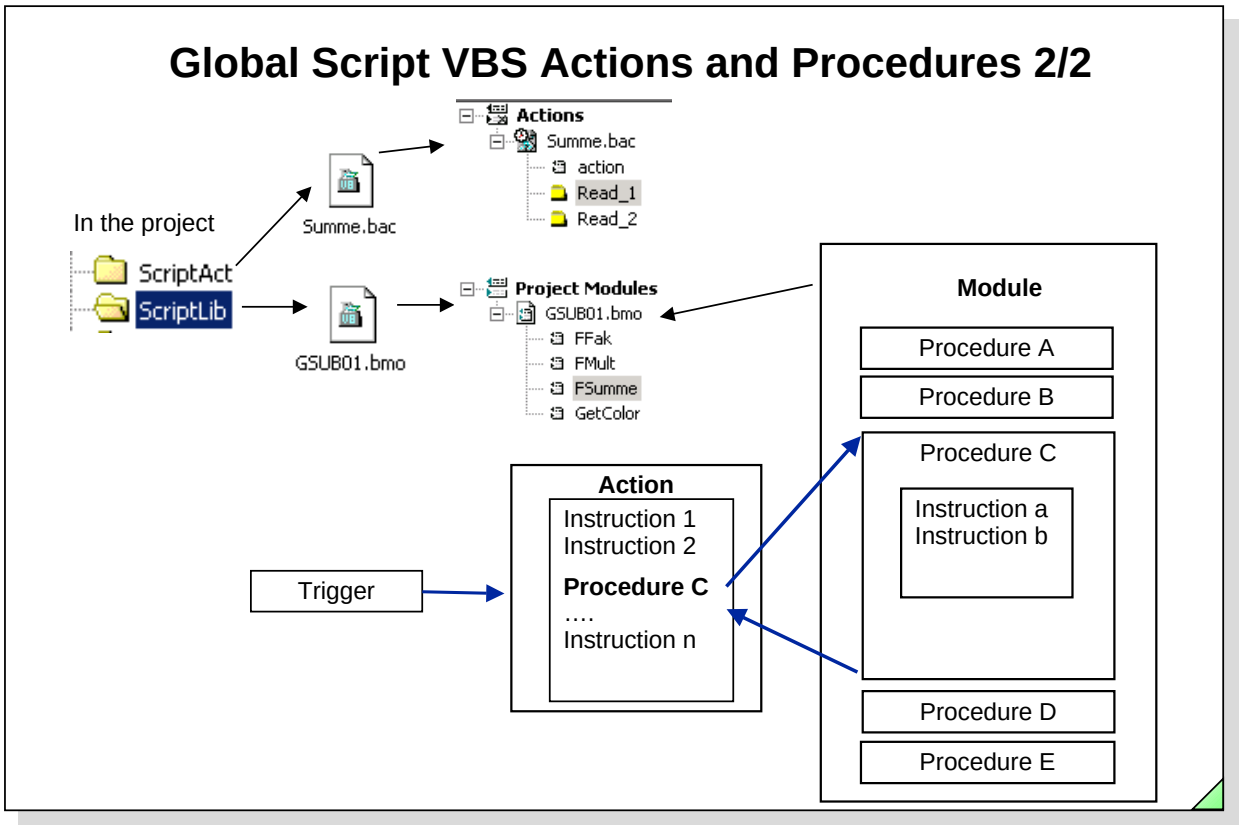
This is where functions are located that can be used for more than one project. (ScriptLibStd Catalog).

Actions

Actions are available only in the Global Script. This is where actions are defined that run in the background and are not connected directly with the selected WinCC object (screen, screen window, graphic object, etc.). For example, the daily transmission of values into an Excel spreadsheet.

Code Templates

Code templates provide the programming person with frequently used instructions, such as for loops and conditional instructions. You can insert these templates into your procedure code using drag & drop.



VBS in WinCC enables you to use procedures, modules and actions for making your runtime environment dynamic.

Procedures

A procedure corresponds to a function in C. You store the code in procedures that you would like to use in several locations in your configuration. You call the code in an action or in another procedure by calling the procedure name. In WinCC you can create procedures with or without a return value. Procedures don't have a trigger; the call is always made through an action.

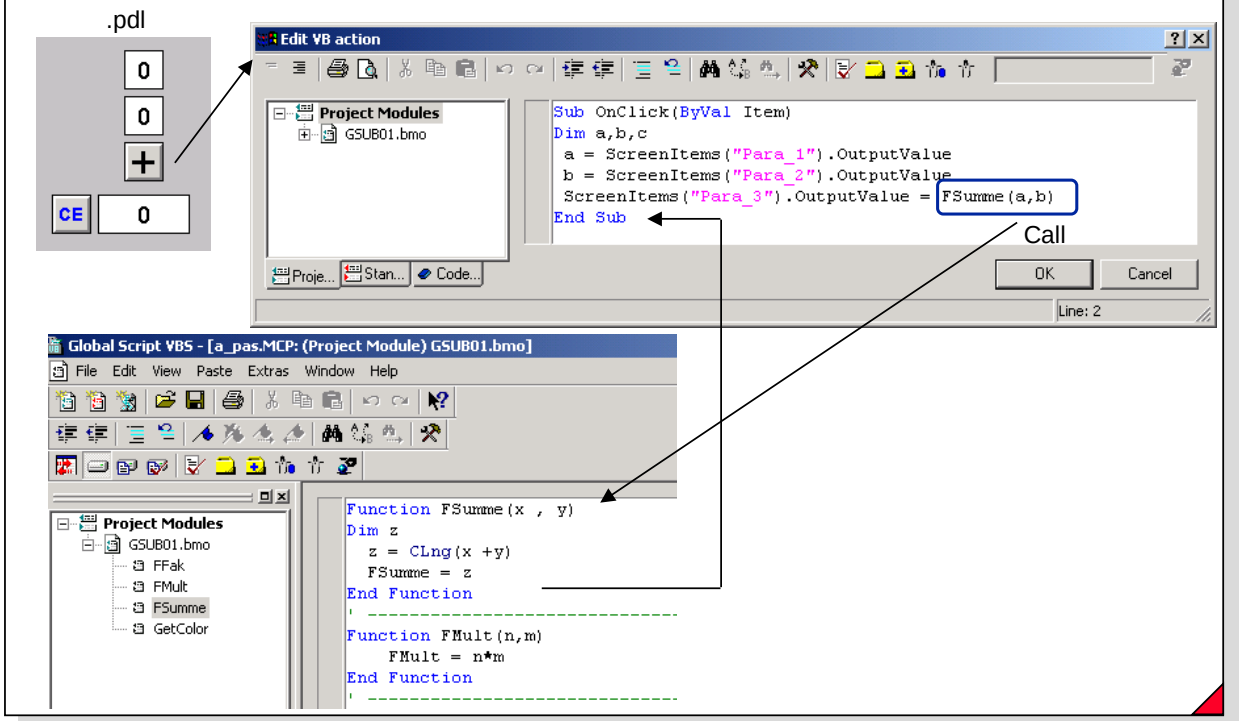
Modules

In modules, you group procedures into meaningful units. You create, for example, modules for procedures that are used in a specific picture or that belong to a specific topic, such as mathematical auxiliary functions or database access functions.

Actions

Actions are always brought to a start through a trigger; in other words through a triggering event. You configure actions for properties of graphic objects, for events that occur in a graphic object, or globally in the project. In actions, you can call frequently used code in the form of procedures.

Example of a VBS Project Function



General Information For frequently occurring applications, you can create a Global Script function. Then the function is called in a picture object and supplied with operands.

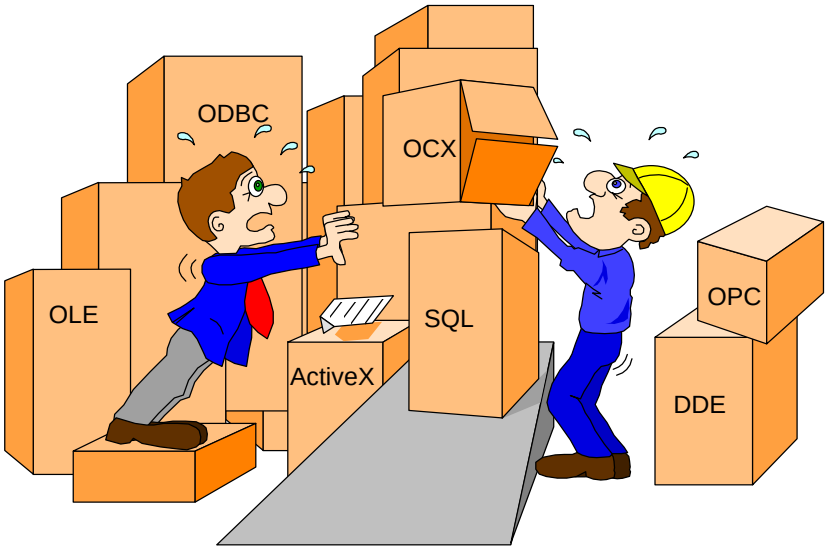
Script Operands

Para 1 = Object name IO Field1
Para 2 = Object name IO Field2
Para 3 = Object name Output field
FSumme=z is the return value from the function.
CLng Function = returns a term that was converted into a value of the Variant type with the sub-type Long.

Execution Event

1. Cell: Tag declaration
2. Cell : Through the OutputValue property, read the value of the 1. IO Field.
3. Cell : Through the OutputValue property, read the value of the 2. IO Field.
4. Cell : Call the function Summe and supply the output field with the return value.

Using the Windows Standard Interfaces



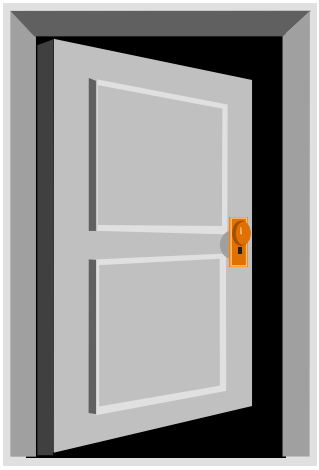
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_10e.1

SITRAIN Training for Automation and Drives

Contents Page

Objectives



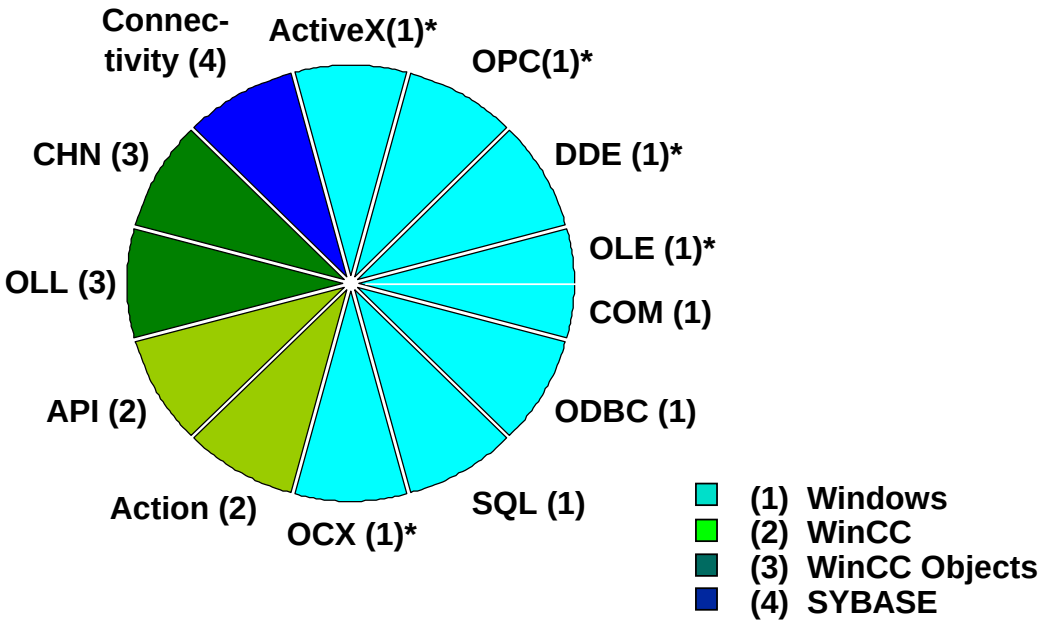
- Open Architecture
- OLE Automation
- ODBC
- DDE
- OCX
- ActiveX
- OPC
- VBS

Date: 27.10.2003
File: SWINCC_10e.2

⇨ **Upon completion of this chapter, the participant should be familiar with:**

- the advantages of open architecture.
- the integration of an OLE automation object, based on an example.
- the ODBC connection to a database, based on an example.
- a connection using DDE, based on an example.
- the advantages of "documents" in WinCC.
- the advantages of "standard market objects (OCX and ActiveX)".

An Overview of the WinCC Open Architecture



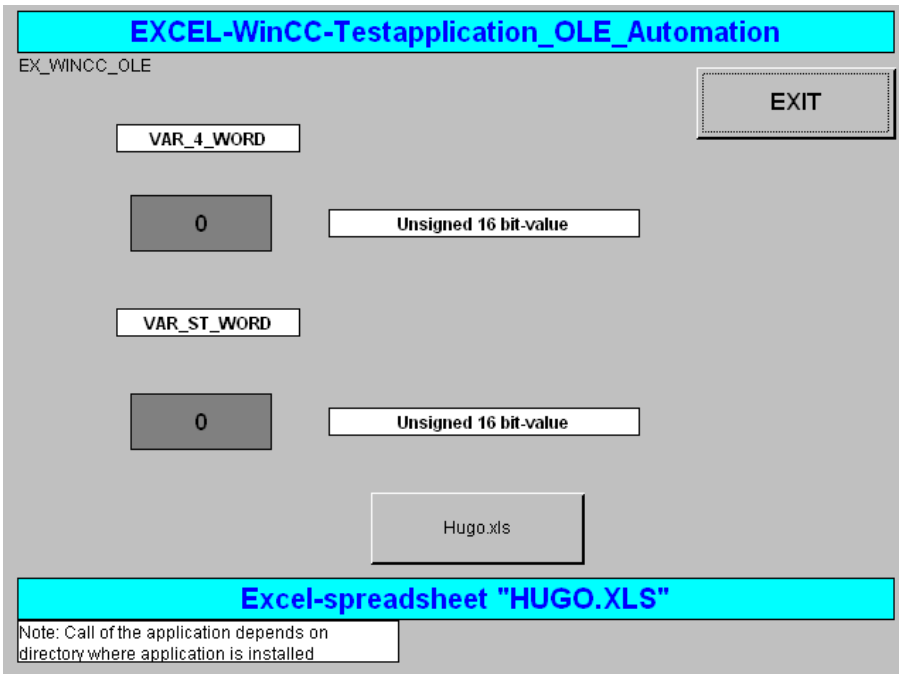
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_10e.3

SITRAIN Training for Automation and Drives

Open Architecture	This slide shows all system modules with opened interfaces. The user has the opportunity here to integrate his own programs in WinCC. Here are five examples identified with an asterisk (*). Refer to the CD-ROM for additional examples.	
Acronyms:	**Description:**	
ActiveX	**(1)***	Software module for applications and Internet (see OCX,OLE)
OPC	**(1)***	Communication standard for automation components and office areas
DDE	**(1)***	Dynamic Data Exchange (data exchange between applications)
OLE	**(1)***	OLE 2.0 (using the functions of another program)
ODBC	**(1)**	OpenDataBase Connectivity (standard for access to databases)
SQL	**(1)**	Database query language (mostly with SQL commands)
OCX	**(1)***	Objects VB 4.0 (standard objects, 32 bit, OLE Custom Control)
Action	**(2)**	Action configuring (for example, messages, measured values, and time-referenced tags)
API	**(2)**	WinCC Application Programming Interfaces (functions, for example, changing languages and module call-up)
OLL	**(3)**	WinCC objects (for example, OCX+ changing languages)
CHN	**(3)**	Channel (communication driver to WinCC)
Connectivity	**(4)**	Databases (for example, ORACLE, INGRES, DB2, IMS, ADABAS, etc.)
ODK	**(3,2)**	Open Development Kit (own objects OLL, own message frames DLL)
CDK	**(3,2)**	Channel Development Kit (development of own DLLs for the data manager)
COM/DCOM	**(1)**	Component Object Model and Distributed COM is a specification for finding and connecting objects and the execution of methods (client-server).

OLE Automation (1/2)
(Access to WinCC Tags through Excel, for Example)



Date: 27.10.2003
File: SWINCC_10e.4

Objective

This example clarifies how links are made between Microsoft Excel and WinCC through OLE Automation 2.0.

Configuring WinCC

Two input/output fields, the output format (999999), and mouse operation, each with a "VAR_4_WORD", "VAR_ST_WORD" internal tag (unsigned 16-bit value) are configured in the screen.

Dialog

When you operate "GetValue()" D-6 switch in the Excel table, the value of WinCC is read into the table. When you operate the SetValue()" E-6 switch in the Excel table, the value is written from the table into WinCC.

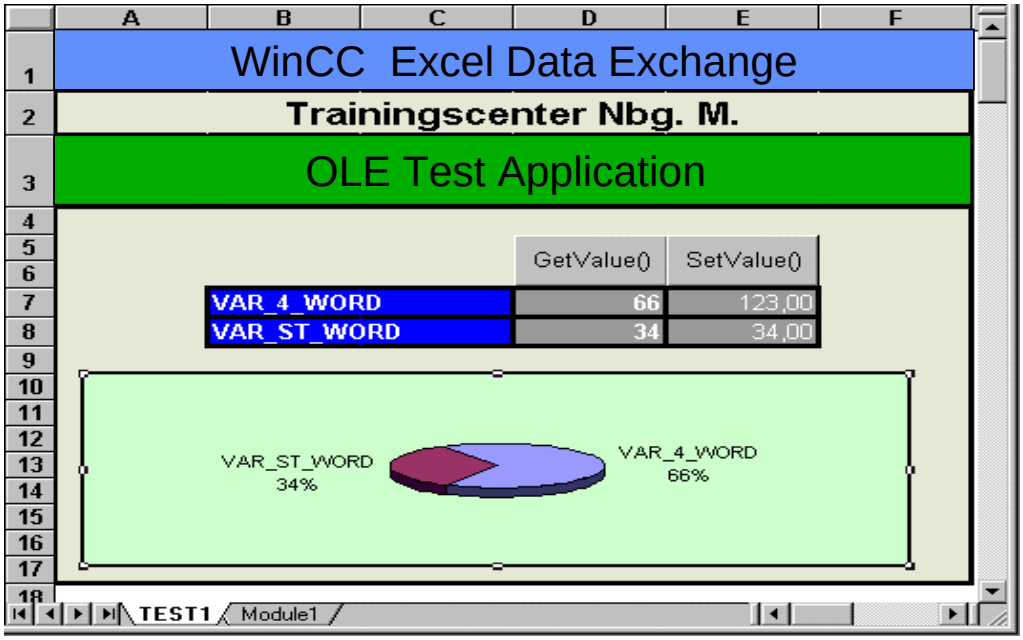
Configuring Excel

The Excel table is used to write or read the "VAR_4_WORD", "VAR_ST_WORD" tag values to or from WinCC.

Excel Macro

In Excel, you create the "GetValue()," "SetValue()" macro in the menu Tools, in the drop-down menu Macro.
(See the program list on the next page.)

OLE Automation (2/2)
(Access to WinCC Tags through Excel, for Example)



Writing

Configuring the SetValue() macro:

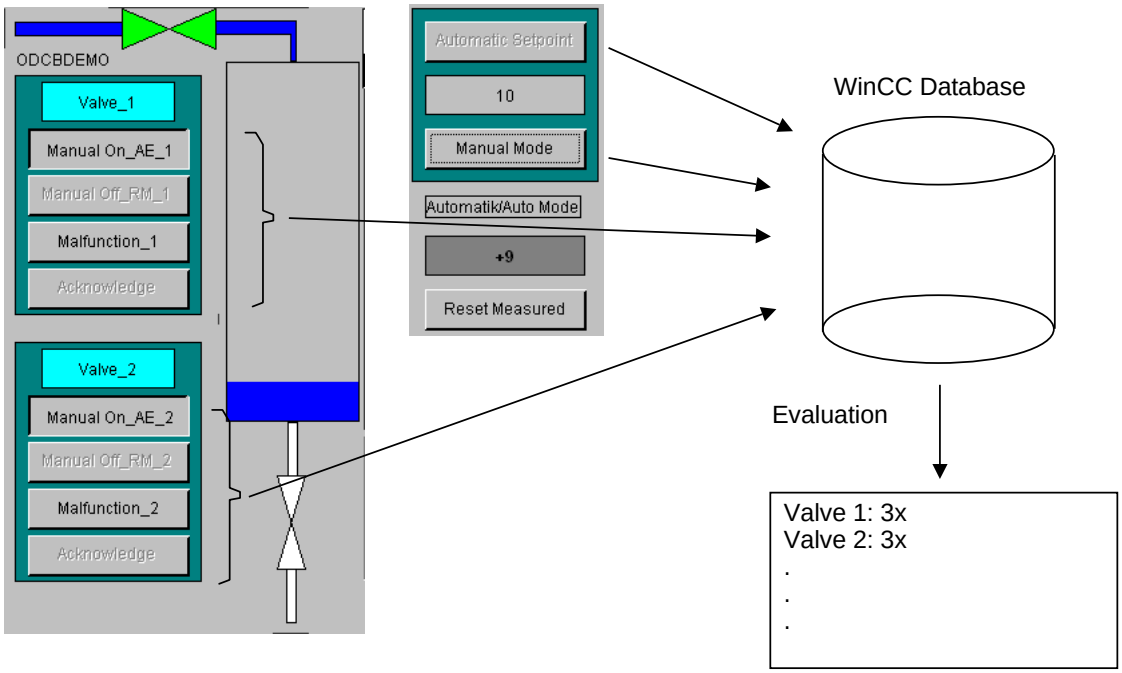
```
Sub SetValue()
Dim mcp      As Object
Dim value    As Variant
Dim vars     As Variant
Dim cell     As Variant
Dim i        As Integer
Dim bRet     As Integer
vars = Array("VAR_4_WORD", "VAR_ST_WORD")
Set mcp = CreateObject("WinCC-Runtime-Project")
For i = 0 To 1
    cell = "E" & (i + 7)
    value = Range(cell)
    bRet = mcp.SetValue(vars(i), value)
Next
End Sub
```

Reading

Configuring the GetValue() macro:

```
Sub GetValue()
Dim mcp      As Object
Dim value    As Variant
Dim vars As Variant
Dim cell     As Variant
Dim i        As Integer
vars = Array("VAR_4_WORD", "VAR_ST_WORD")
Set mcp = CreateObject("WinCC-Runtime-Project")
For i = 0 To 1
    cell = "D" & (i + 7)
    value = mcp.GetValue(vars(i))
    Range(cell).Select
    ActiveCell.FormulaR1C1 = Str(value)
Next
End Sub
```

ODBC Application with Data from WinCC 1/3



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_10e.6

SITRAIN Training for Automation and Drives

Objective

This example illustrates how an application with ODBC functions can access data from WinCC.

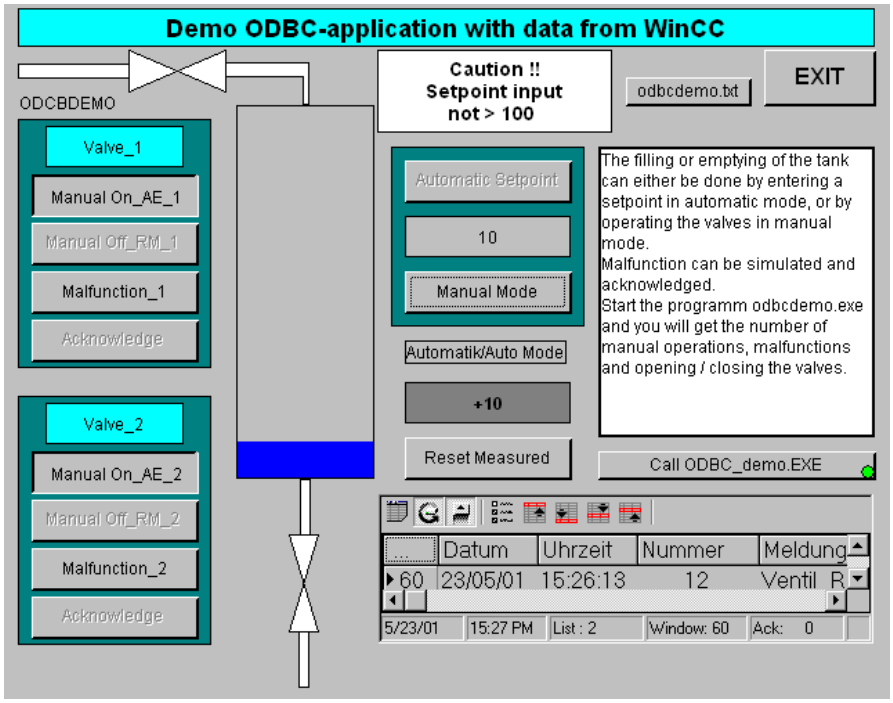
Configuring WinCC

The slide displays a container with an inlet valve (valve_1) and a outlet valve (valve_2). The container can be filled or emptied by manual operation or automatic operation (AUTO/MANUAL). During the process, both valves are opened and closed. Faults (FAUL_1/FAUL_2) can be caused and acknowledged by means of a mouse click. Management of both the switches and the faults leads to status messages that are saved in the message system and therefore in the WinCC database.

Evaluation

An evaluation is implemented by the odbcdemo.exe (Window-Box) program. This program was made with Visual Basic V5. A connection is set up to the WinCC database (with ODBC functions).
The number of events for inlet/ outlet valve open, manual/automatic switch and fault are then counted and the result is output to the screen and is written in the ASCII file ‘odbcdemo.txt.’ You can print out the ‘odbcdemo.txt’ file or output it to the screen or process it further with other programs.

ODBC Application with Data from WinCC 2/3

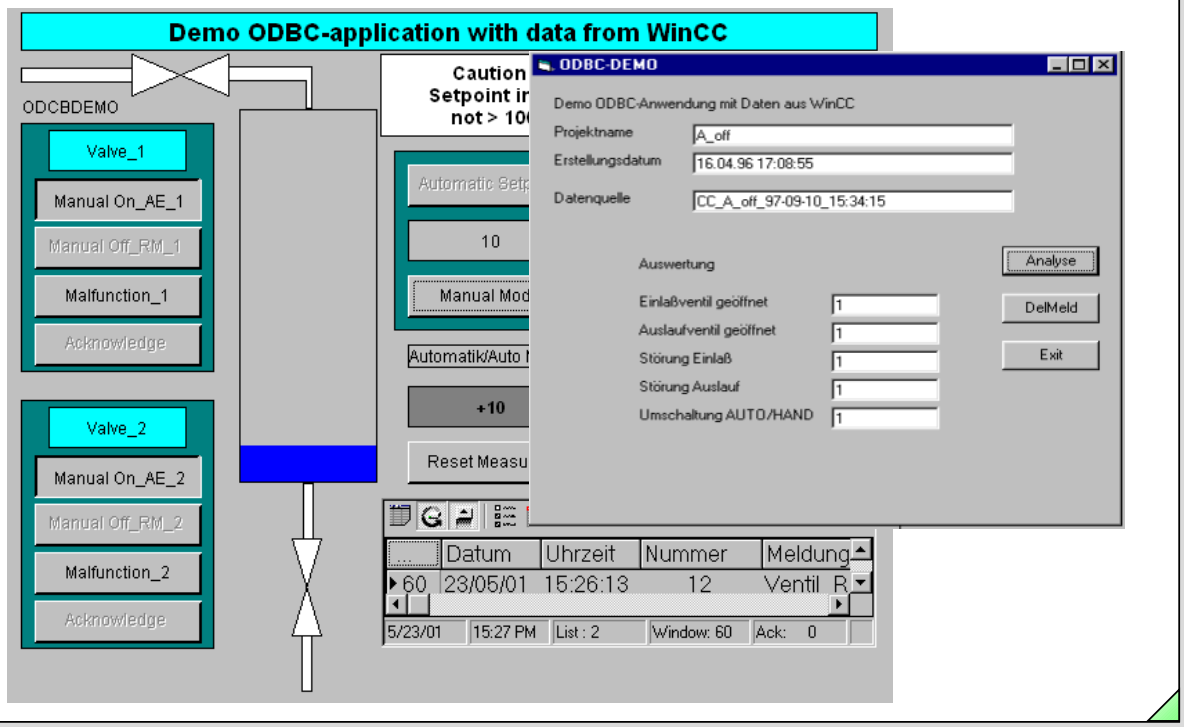


Date: 27.10.2003
File: SWINCC_10e.7

Configuring ODBC

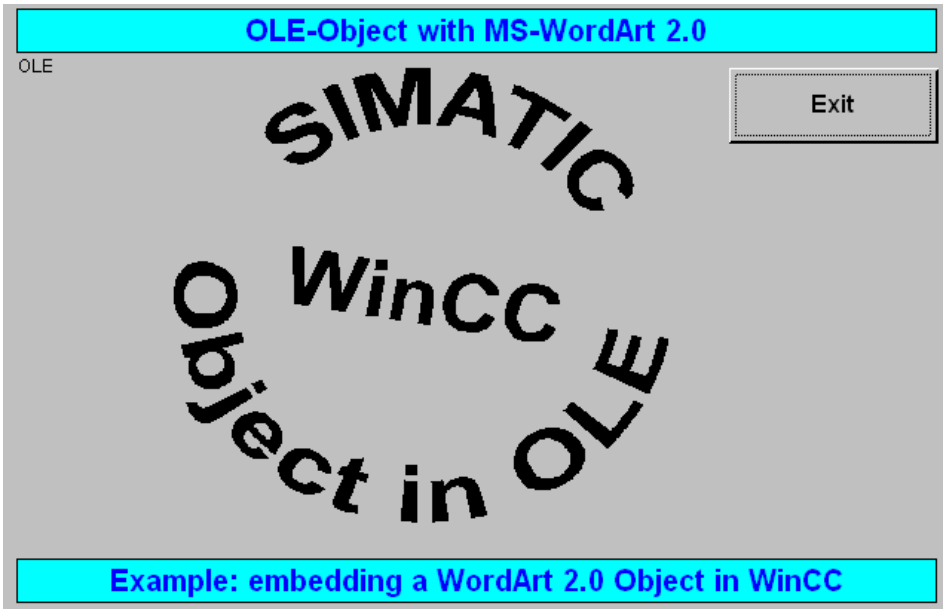
To implement this demonstration example, the "C" standard functions, from <math.h>, <stdio.h>, <stdlib.h>, <windows.h>, and <odbcinst.h> were used, as well as the following ODBC functions SQLAllocEnv, SQLAllocConnect, SQLSetConnectOption, SQLDriverConnect, SQLFreeConnect, SQLFrreEnv, etc. (Contained in <SQL.H> and <SQLEXT.H>) .

ODBC Application with Data from WinCC 3/3



Evaluation The screen shows the "odbcdemo.exe" with its evaluation in a Windows box.

Integrating OLE Objects in WinCC with MS WordArt 2.0, for Example



Date: 27.10.2003
File: SWINCC_10e.9

OLE Object

In WinCC, you can insert OLE objects under “Smart Objects.” You can:

- Create new
- Create from the file

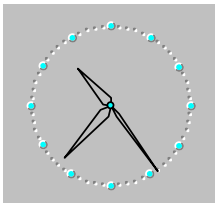
The selection depends on which object types are loaded on your computer. Some object types, such as bitmaps and MIDI sequences, already exist in the operating system by default.
The slide displays a Microsoft WordArt 2.0 object type.
In the Graphics Designer, you can double-click on the object to call up and change the object type.

Note

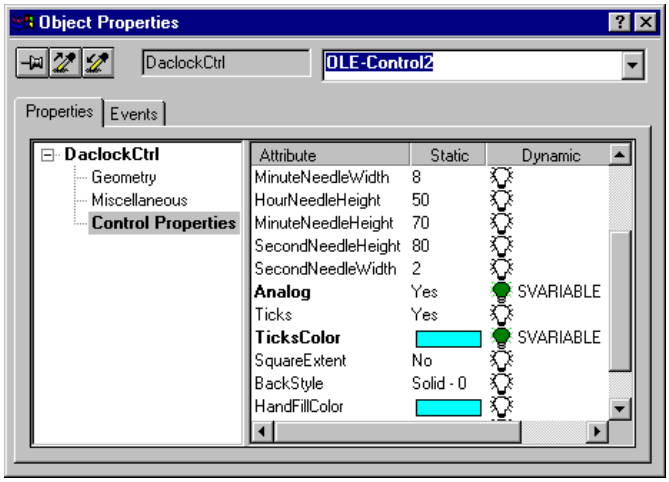
You must install Microsoft Word in order to use Microsoft WordArt 2.0.

Using OCX and ActiveX, Standard Market Objects 1/3

WinCC OCX



VB Object Palette with OCX Objects



OCX, ActiveX Object

OCX and ActiveX objects provide certain functionalities that are not covered as defaults by the WinCC objects. In these cases, WinCC makes it possible to embed OCX and ActiveX objects in pictures. The system can then use the OCX and ActiveX objects to carry out more complex functions. In the Object Palette, under Smart Objects, OLE Control, there are several OCX, ActiveX objects that you can integrate into your own project.

Some examples:

- WinCC Digital/Analog Clock Control
- WinCC Push Button Control
- WinCC Slider Control.

In addition, OCX and ActiveX objects from outside manufacturers or those created by you, can be embedded in WinCC pictures; however, WinCC does not provide any guarantees for their function. Such "external OCXs" should only be implemented in WinCC following extensive tests (see note in product information).

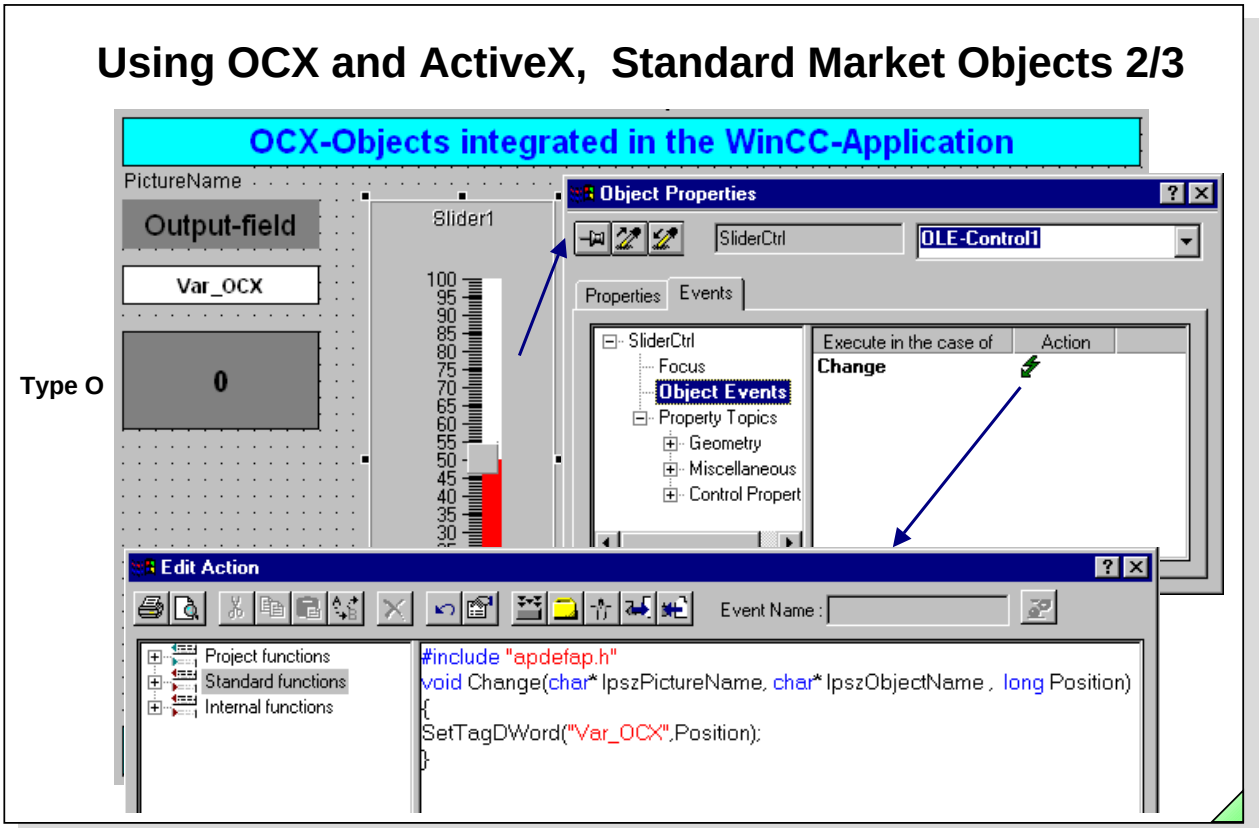
The WinCC OCX container provides the basic properties for the configuration. All of the OCX-specific properties are set up on their own property page.

The picture shown above contains the property page of the WinCC digital/analog clock control.

IndustrialX

IndustrialX is a WinCC option package for easy creation of ActiveX objects (using Visual Basic V5).

Using OCX and ActiveX, Standard Market Objects 2/3



Type O

Objective

This example illustrates how a link is made between an OCX slider and WinCC.

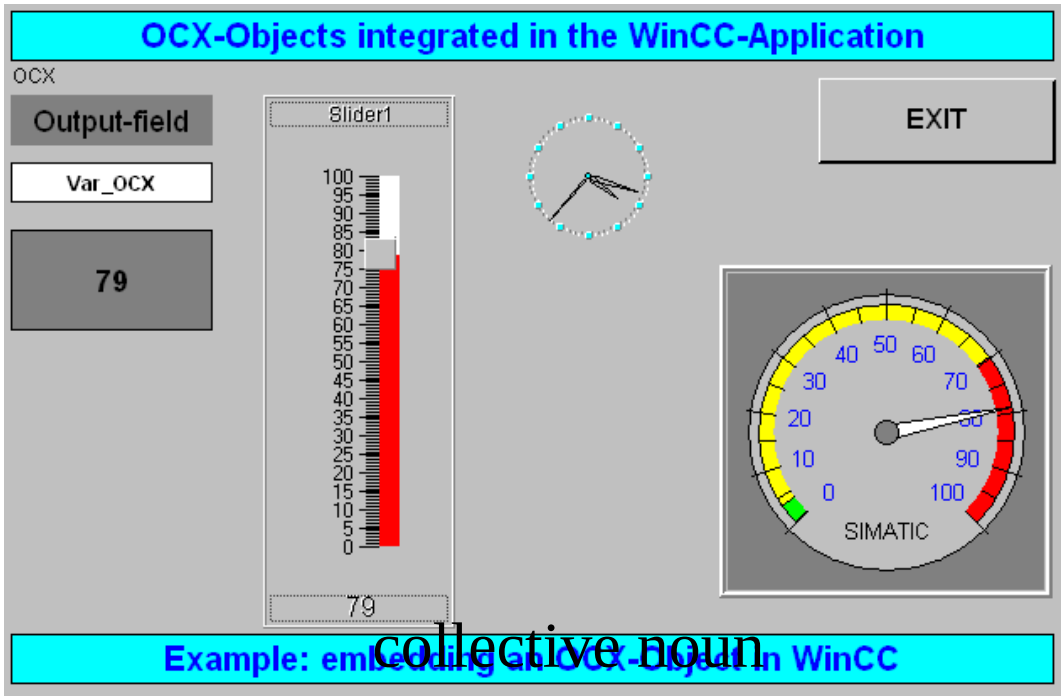
Configuring WinCC

The slide shows a slider that you can locate under Object Palette\Controls\ **WinCC Slider Control**.
You can change the position of the slider's linear regulator using the mouse (see configuration above), and corresponding to its position, transfer the value into an internal "Var_OCX" tag (unsigned 16-bit value).

Dialog

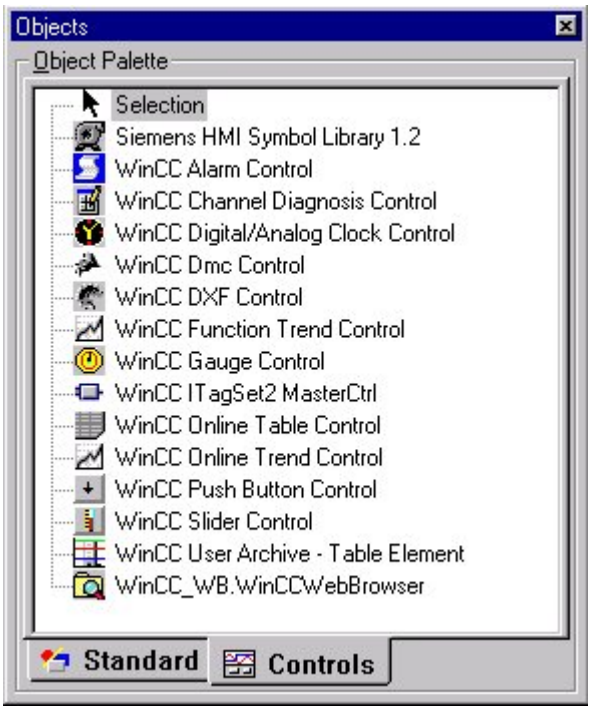
The output field of the **O**utput type is to be connected to the slider.

Using OCX and ActiveX, Standard Market Objects 3/3

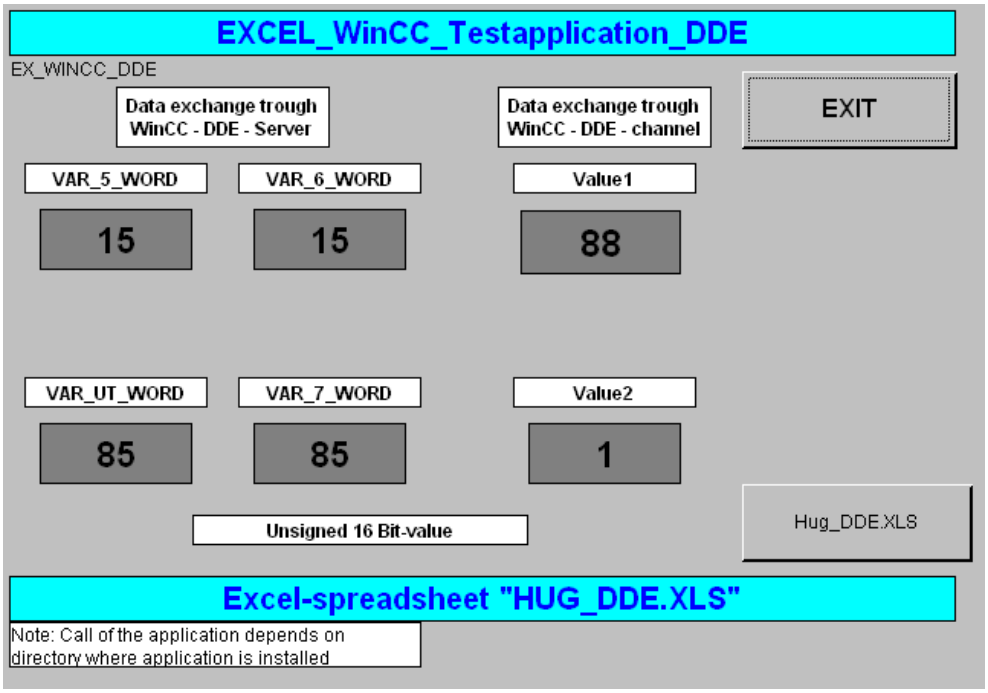


Controls

In the WinCC-Graphics Designer there is a large number of Controls in the Control Palette that can be used in WinCC. Controls is a collective noun for OCX and ActiveX.



DDE Linking 1/3
(Accessing WinCC Tags through Excel, for Example)

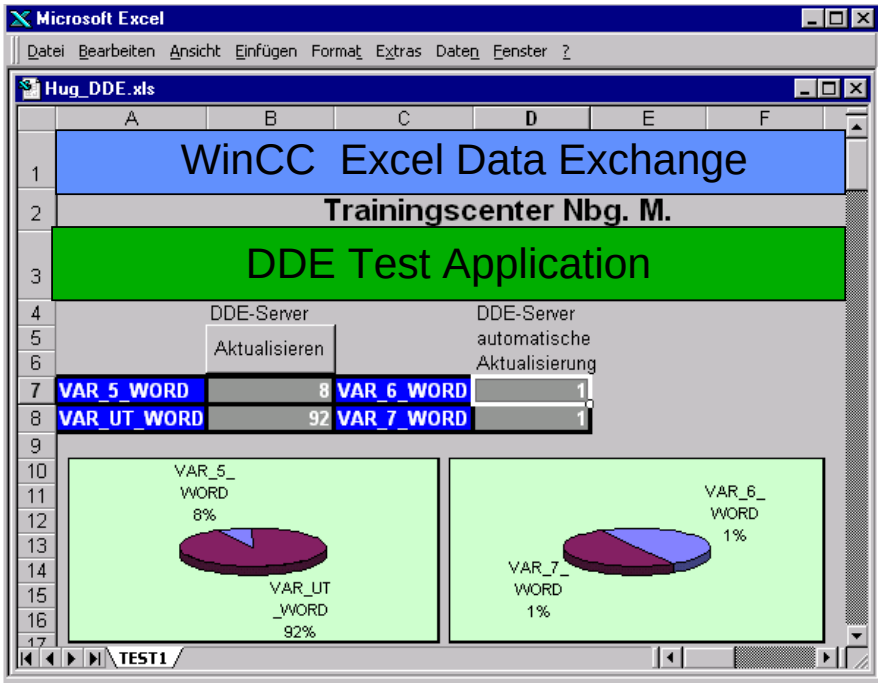


Date: 27.10.2003
File: SWINCC_10e.13

Objective This example illustrates how links are made between Microsoft Excel and WinCC using DDE.

WinCC Configuration In the slide, there are six Input Output fields configured each with one internal tag or DDE tag "VAR_5_WORD", "VAR_UT_WORD", "VAR_6_WORD", "VAR_7_WORD", "Wert1" and "Wert2" (unsigned 16-bit value).

Linking using DDE Server 2/3 (Accessing WinCC Tags through Excel, for Example)



Date: 27.10.2003
File: SWINCC_10e.14

WinCC-DDE Server

In order to be able to access WinCC data from Excel (for example, internal tags), the DDE Server has to be started in addition to Runtime. This is done in the Startup list.
You can update in Excel manually or automatically.

Dialog

Values are to be fetched from WinCC and updated in the Excel spreadsheet by operating the pushbutton "Aktualisieren" (Update) (in the Excel spreadsheet). For this, you have to edit the following VB module in Excel.

```
Sub Aktualisieren_1()
   ActiveWorkbook.UpdateLink
        Name:="WinCC|C:\...\ab_off\A_off.mcp!VAR_5_WORD", _
        Type:=xlOLELinks
   ActiveWorkbook.UpdateLink
        Name:="WinCC|C:\....\ab_off\A_off.mcp!VAR_UT_WORD" _,
         Type:=xlOLELinks
End Sub
```

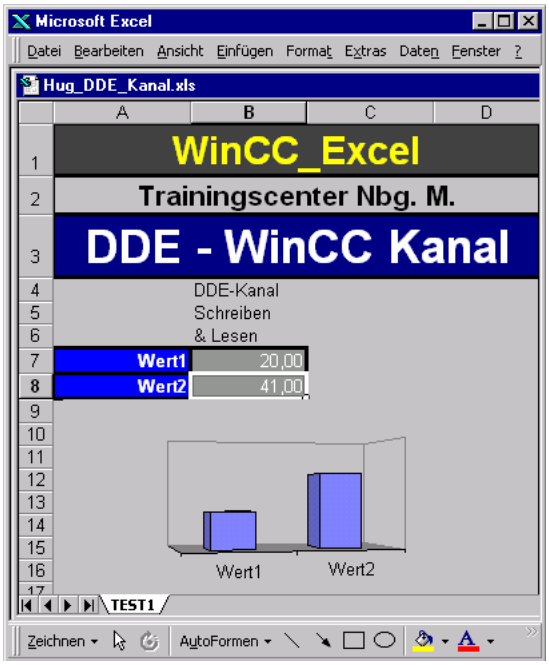
Excel Configuration

In the Excel spreadsheet, the WinCC tags are entered directly.
For the DDE channel, the cell is given the tag name so that the access from WinCC is unique.

The DDE Server must always be addressed as follows:

Application Name WinCC|	Topic and project!	Element Name Variable1
WinCC|	'C:\...\ab_off\A_off.mcp'!	VAR_5_WORD

Linking using WinCC DDE Channel 3/3 (Accessing WinCC Tags through Excel, for Example))



Date: 27.10.2003
File: SWINCC_10e.15

WinCC-DDE Channel

You can directly access Excel variables from WinCC using the DDE channel. In the DDE channel, tags are defined that refer directly to (are linked to) a specific Excel file.
In this Excel file, you can then address individual cells directly with WinCC tags.

DDE Link

The DDE link is configured with the following parameters:

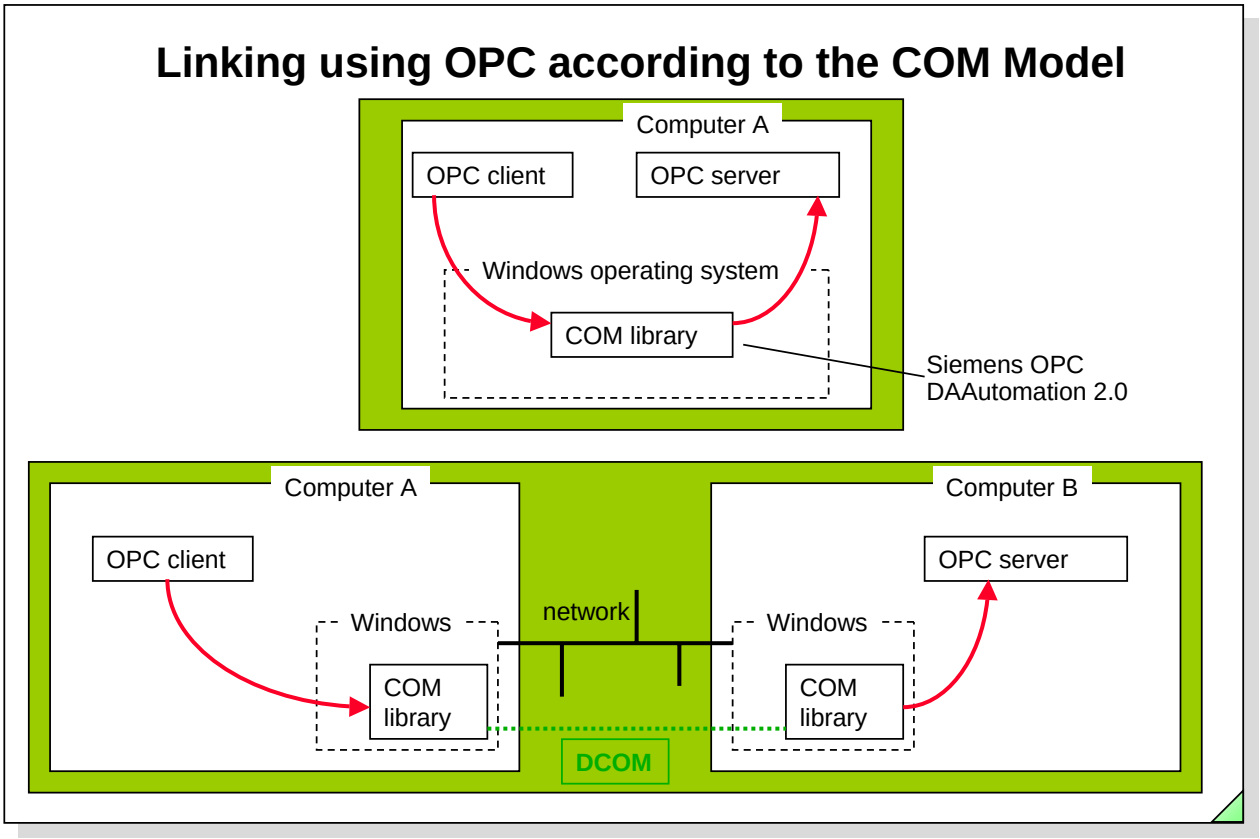
Computer Name:	(Only when accessing through Net-DDE)
Application:	Excel
Topic:	Hug_dde.xls
Optimization: (For Reading Tags)	Fetch Only After Change

DDE Variable

Name:	Wert1
Data Type:	Unsigned 16-bit value
Address:	RnCm (for Row n Column m) or the cellname: cellm
Data Type:	WORD

FAQ

An example is described in **FAQ 761 1667**.

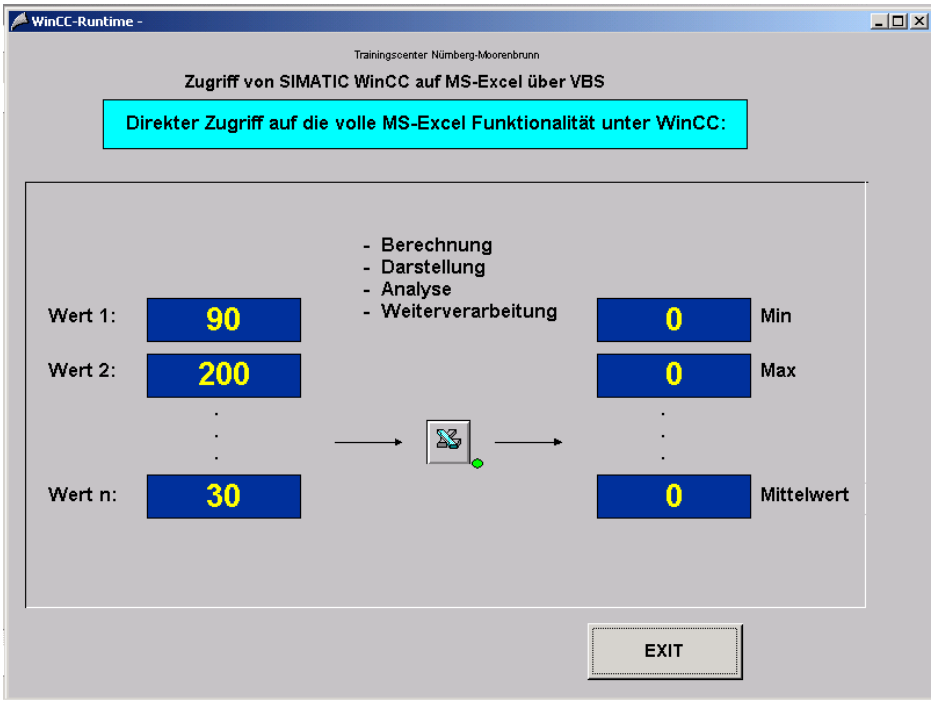


Component Object Model and Distributed COM is a specification for finding and connecting objects and the execution of methods (client-server).

Example

You will find and example in the WinCC Help->Communication->OPC.
This example describes the data exchange between WinCC and Excel.

Direct Access to the Complete MS Excel Functionality under WinCC

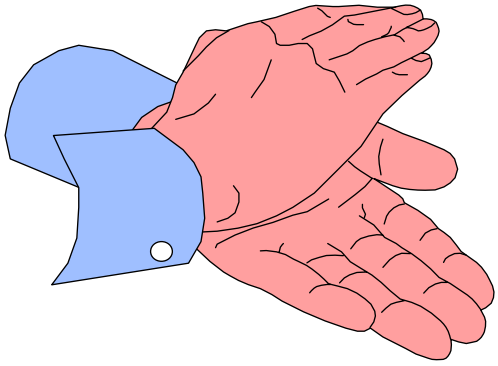



```
Sub StartX0000005FGD1_OnClick(ByVal Item)
Dim g_excelfilename
Dim excel

g_excelfilename = HMIRuntime.Tags("ProjectPath").Read & "\Excel1.xls"

Set excel = CreateObject("Excel.Application")
excel.Visible = True
excel.Workbooks.Open g_excelfilename
excel.Cells(4, 3).Value = ScreenItems("V6_ExcelWrite_1").OutputValue
excel.Cells(5, 3).Value = ScreenItems("V6_ExcelWrite_2").OutputValue
excel.Cells(6, 3).Value = ScreenItems("V6_ExcelWrite_3").OutputValue
excel.ActiveWorkbook.Save
MsgBox ("Continue ?")
ScreenItems("V6_ExcelRead_1").OutputValue = excel.Cells(4, 5).Value
ScreenItems("V6_ExcelRead_2").OutputValue = excel.Cells(5, 5).Value
ScreenItems("V6_ExcelRead_3").OutputValue = excel.Cells(6, 5).Value
excel.ActiveWorkbook.Save
excel.Workbooks.Close
excel.Quit
Set excel = Nothing

End Sub
```

Direct Access to the Complete MS Word Functionality under WinCC


```
Sub ObjectDefinedEvent(ByVal Item )
Dim k,m, aVarName(6), aVarValue(6), oWord, oApp, oTbl, oDoc
 m= 6                          ' Read Variable Names and Values
 For k = 1 To m
    aVarName(k)= ScreenItems("lap" + CStr(k)).Text
    aVarValue(k)= HMIRuntime.Tags("V6_Word_Value" + CStr(k)).Read
 Next
 ' Open Word document and get first table
  Set oWord = CreateObject("Word.Application")
  oWord.Visible = vbTrue
  Set oDoc = oWord.Documents.Open(HMIRuntime.Tags("ProjectPath").Read & "\MyWord.doc")
     Set oApp = GetObject(, "Word.Application")
  oApp.Visible = vbTrue
  Set oTbl = oApp.ActiveDocument.Tables(1)     ' Write values
 For k=1 To m
     oTbl.Cell(k + 1, 1).Range.Text = aVarName(k)
     oTbl.Cell(k + 1, 2).Range.Text = aVarValue(k)
 Next
' falls Druckeroption gewählt
' oDoc.PrintPreview, oDoc.PrintOut, MsgBox ("Continue ?")
  oApp.Quit
 Set oApp = Nothing
 Set oTbl = Nothing
 Set oApp = Nothing
 End Sub
```

User Openness API



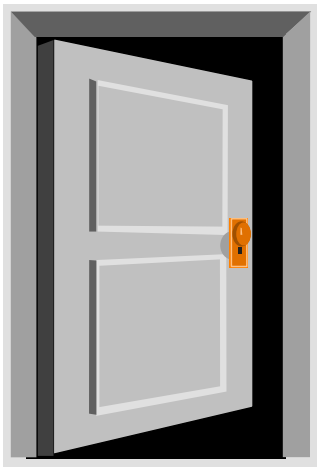
SIMATIC HMI

Date: 27.10.2003
File: SWINCC_11e.1

SITRAIN Training for Automation and Drives

Contents Page

Objectives

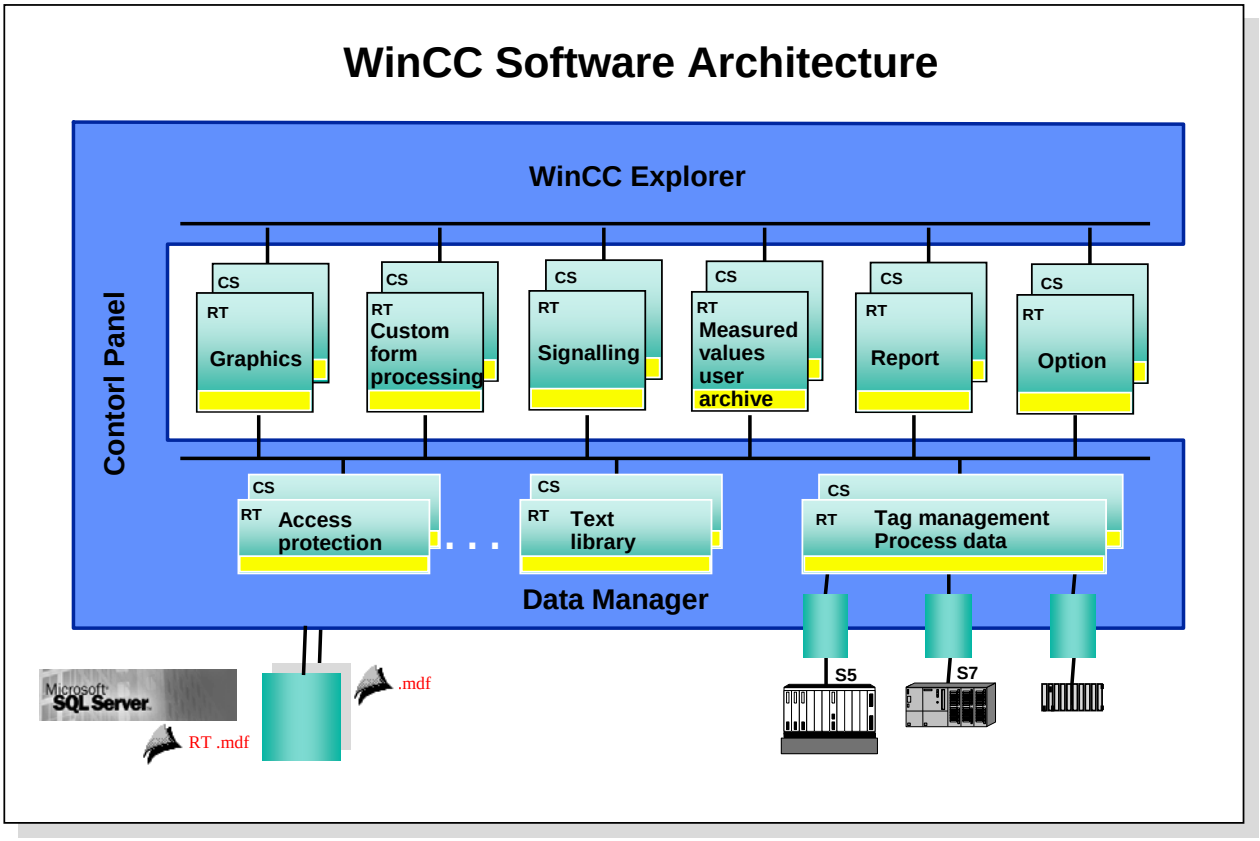


- API (Application Programming Interface)
- ODK (Open Development Kit)
- CDK (Channel Development Kit)

Date: 27.10.2003
File: SWINCC_11e.2

⇨ **Upon completion of this chapter, the participant should be able to:**

- identify the relationships between the API interfaces



SIMATIC HMI

Date: 27.10.2003
File: SWINCC_11e.3

SITRAIN Training for Automation and Drives

Software Architecture

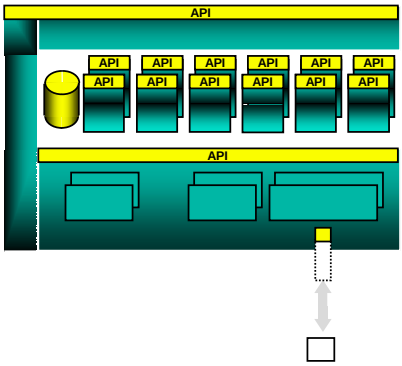
The WinCC software is modularly structured.
Each WinCC component has one API interface that helps to open WinCC and that other applications can use.

There are **API calls for Configuration and Runtime** components:

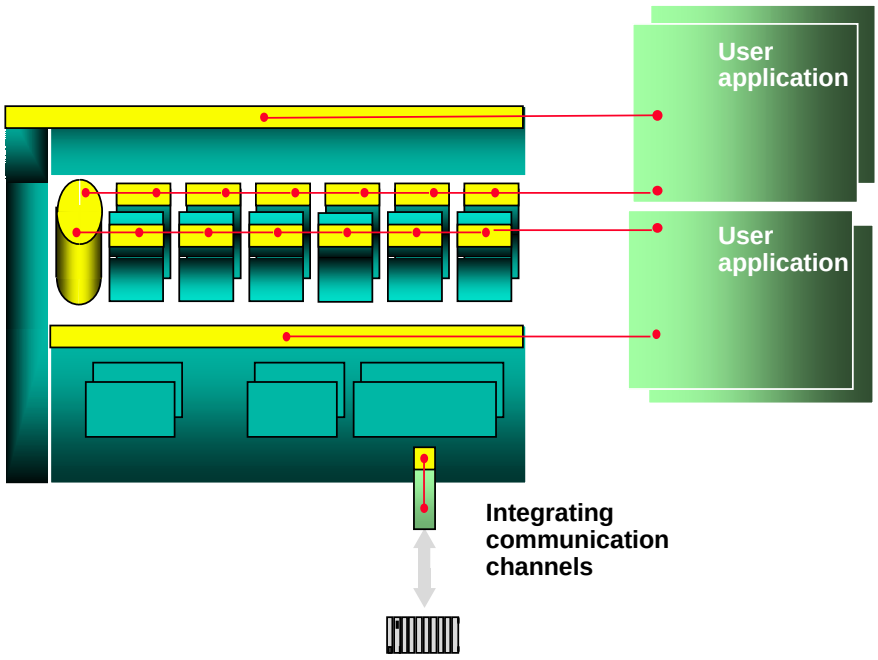
- WinCC Explorer with Data Manager
- Graphics Designer
- Global Scripts
- Alarm Logging
- Tag Logging
- Report Designer
- Diagnostics
- User Administrator
- Text Library

Application Programming Interface API (1/4)

- ANSI-C Functions
- Access to data
  - in the Tag Management
  - in the archives
  - configuration data
- Using system functions (ODK)
  - configuration functions
  - Runtime functions
- Integration of communication channels (CDK)
- Basis for
  - individual functions
  - technological-type options
  - industry-referenced function modules

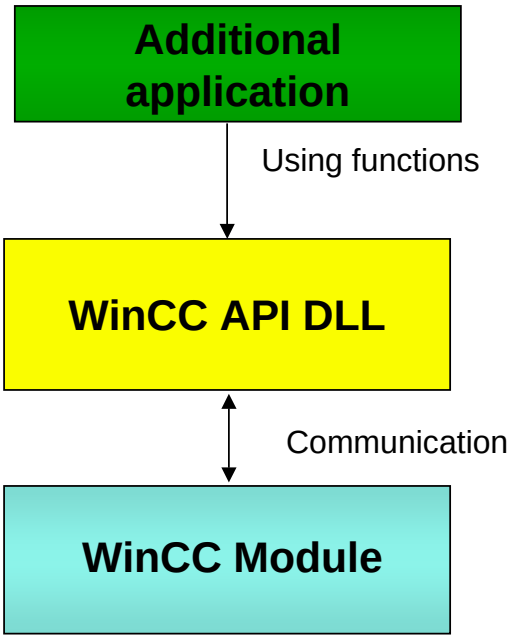


Application Programming Interface API (2/4)



Date: 27.10.2003
File: SWINCC_11e.5

Application Programming Interface API (3/4)



The following is needed for creating applications:

- **External C compiler (for example, Microsoft Visual C++ 5.0)**
- **Library file for the desired WinCC API (*.LIB)**
- **Header file for the desired WinCC API (*.H)**

Date: 27.10.2003
File: SWINCC_11e.6

General Information

To create external applications, you primarily need three parts. The ODK contains the **Library** and **Header** files for the WinCC-APIs. In addition, an external C compiler is necessary that can be used to create the external application.

Application Programming Interface API (4/4)

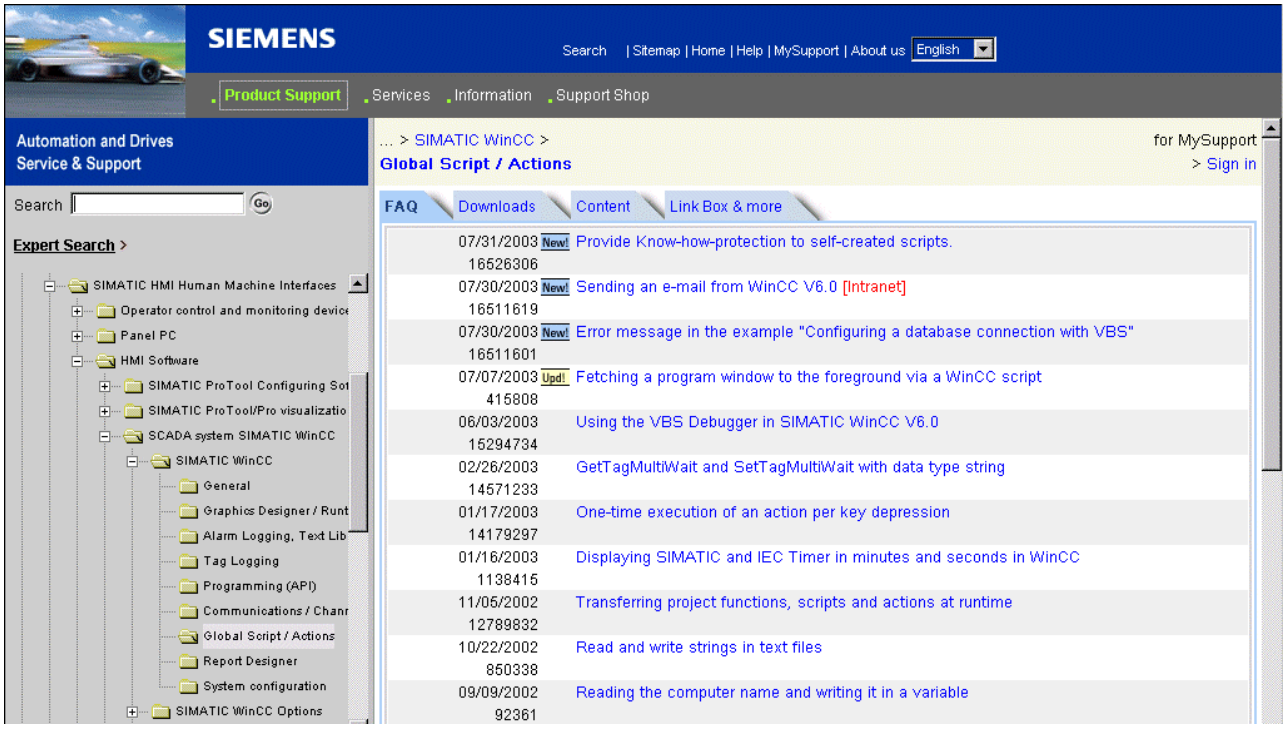
- **Application examples:**
  - **Access to values from the Tag Management**
  - **Access to archive data**
  - **Integrating communications channels**
  - **Direct display of processing results in Graphics**
  - **Generic configuration based on API functions**
  - **Integrating already existing processing programs**
  - **Generating messages from user routines**
  - **Controlling the measured value recording from user routines**
  - **Controlling reports from the user programs**

Processing Results	The processing results of the WinCC script control can be displayed directly on the graphics screen.
Generic Configuring	Generic configuring refers to generating WinCC elements with the help of an API interface. Generic configuring makes it possible for you to describe and create WinCC structures using texts.

Table of Contents Utilities

Note: Several of the examples originate from the FAQs, the complete selection can be found on the Internet.

FAQs on the Internet: http://www.siemens.de/hmi



Applications and Configuration of a Direct Connection.

In addition to Actions, it is also possible, with WinCC, to configure events with a direct connection. The **Performance** can be increased through the use of a direct connection instead of Actions. In the selected graphic object in the "Properties" menu in the "Event" tab, you can choose the direct connection by clicking the lightning symbol (for example, in Event "Mouse – Mouse-click") with the right mouse button (Menu item "Direct Connection").

The following possibilities are available:

Source: Constants or Property or Tag.
Target: Current Window or Object in Picture or Tag.

Application: Objects Visible Non-visible For example - An object in the picture is to become visible or non-visible with every mouse-click on an "On" button and "Off" button.

Prerequisite: In Property/Other, the Display attribute of the object to be made dynamic, for example, Static Text, is to be set to No.

Source: Constants 1 for button "On"
Target: Object in Picture >Object: for example, Static Text n >Property: Display

Source: Constants 0 for button "Off"

Target: Object in Picture >Object: for example, Static Text n >Property: Display

Application: The decimal value of an output field is to be graphically displayed in a Bar, for example.
Prerequisite: The output field is linked with a tag via the Output Value attribute, and a Bar has been created.

Source: Property >for Object: I/O Field n >Property: Output Value.
Target*: Object in Picture* >Object: Bar n >Property: Process Driver Connection.

Application: For example - The background color of a circle is to assume the background color of a rectangle per mouse-click.
Prerequisite: Button, Circle and Rectangle are created.

Source: Property >for Object: Circle n >Property: Background Color.
Target: Object in Picture >Object: Rectangle n >Property: Background Color.

Application: For example - With every mouse-click on a button, the value of a tag is to be used as the fill level display for a polygon.
You select the source object, for example, a button and choose a tag.
Then, a property can be chosen at the target object.

Souce: Tag for example, Actualvalue_Tank1
Target: Object in Picture >Object: Polygon n >Property: Fill Level.

Exporting, Changing and then Reimporting Message Texts.

The contents of the exported text file (*.txt) is separated by a comma. With the help of the Wizard in Excel, the correct display of the column representation is ensured.

If you change texts, it is important to set the index to "0" in the 10 columns titled "TextID 1" to "TextID 10" (pay attention to the column titles as of version 1.10 that were also handed out)!

Under no circumstances can these fields simply be left with the old index and they must be completely deleted!

The file edited in EXCEL must then be saved as "Text (Tab delimited) ".

The call of the Export/Import function takes place with the Alarm Logging Editor using the menu *Messages*.

An exact description is displayed in the WinCC Help system. Proceed as follows. In the Alarm Logging Editor start the WinCC Help system. In the first step of the *Index* tab, enter the word *single mess(age).* In the second step *double-click* on *export* or *import*.

Exporting/Importing Tags in a *.CSV File

Short Description
From an opened project, the program exports all connections and tags into an ASCII file. These files can then be imported into a second project. The ASCII format enables the files to be edited with a spreadsheet before they are reimported.

The program is installed from the WinCC CD under install WinCC, Components, Smart Tools. On the WinCC CD, in the directory *Documents-English-WinCCInformationSystem.chm*, you will find an exact description of the Smart Tool.

Handling

EXPORT

1. First of all, start WinCC and open the project for the tag export.
2. Using WinCC->Tools-> start the program "TAG Export Import".
3. Select the path and file name to which you want to export. Only the file name without extension is needed here to begin with.
4. Switch to the *Export* mode.
5. Press *Execute.* Confirm the statements in the message box.
6. Wait until the "Finish" display appears in the status field.
7. With the buttons "*VAR*" and "*CON*", take a look at the created files.

IMPORT

1. First of all, start WinCC and open the project for the tag import.
2. Using WinCC->Tools-> start the program "TAG Export Import".
3. Select the path and file name from which you want to import.
4. Switch to the *Import* mode.
5. Press *Execute.* Confirm the statements in the message box.
6. Wait until the "Finish" display appears in the status field (this takes a little longer than with Export!).

In the WinCC Tag Management, take a look at the created files.

Hardcopy Function in Runtime

In Computer Properties, in the "Startup" register, add the application "prtscr.exe" from the sub-directory "WinCC\BIN\" into "Additional Tasks and Applications" via the "Add" key!
In the Control Center under Project Properties, Register Hotkey, set up your desired Hotkey, for example, Ctrl+P.

The Hardcopy is thereby available for RT and Configuration after the project has been activated.

Saving Project Data, without Incorporating the Online Archives

The entire project path is to be saved when saving the data of a WinCC project.

Nevertheless, it must be taken into account that the project path can also contain online data in the database. It is not always necessary to save all online data.

Start the Alarm Logging or Tag Logging Editor to eliminate the online data from the RT database.
In Alarm Logging, choose the Archive Configuration->Long-term archive. Using the long-term archive‘s shortcut menu, you can reset (delete) the archived messages.
In Tag Logging, choose the Archive Configuration->TagLogging Fast or TagLogging Slow . Using the TagLogging Fast or TagLogging Slow archive‘s shortcut menu, you can reset (delete) the archived tags.

You can delete the *.sav files in the GraCS Catalog.

The files in the Prt Catalog can also be deleted, if the report layouts have not yet been modified. (The standard report layouts are made available by the installed system after the Report Designer has been restarted).

The project path with the project database, (Projectname.ldf, Projectname.mdf, ProjectnameRT.ldf, ProjectnameRT.mdf, and Projectname.mcp),the GraCS, Library, Pas (if not empty) and Prt (if not in the installation state) Catalogs are important.

In addition, you could have created standard functions that can be found in the ...\WinCC\ablib\ Catalog and that are also necessary for a complete backup.

Online Language Switch-over with the Action "SetLanguage(ID)".

The online language switch-over is independent of whether the desired foreign language was installed with the installation of WinCC. The configured texts are stored in the picture with the corresponding language ID. It is, however, absolutely necessary that a foreign language is planned for in the Text Library. For that, you can either install WinCC with a foreign language or add a foreign language to the Text Library.

The program is installed from the WinCC CD under Install WinCC, Components, Smart Tools.

For the picture-specific Text Export/Import, you can start the EasyLang.exe through Start-Simatic-WinCC-Tools.

You will find an exact description of the Smart Tools on the WinCC CD in the directory *Documents-English-WinCCInformationSystem.chm*.

Calling WinCC Message Box into the Foreground by Mouse-click.

```
{
HWND Handle;
Handle=
FindWindow("PDLRTisAliveAndWaitsForYou","WinCC-Runtime -");

MessageBox (Handle, "Emerg. Tel. 0118              ","Operator Help",
MB_OK | MB_ICONEXCLAMATION | MB_SYSTEMMODAL);
}
```

Implementing LOGIN/LOGOUT via Button

- Action for LOGIN (e.g. on Event Mouse-click)

```
#pragma code ("useadmin.dll")
#include "PWRT_api.h"
#pragma code()
PWRTLogin('c');
```

- Action for LOGOUT (e.g. on Event Mouse-click)

```
#pragma code ("useadmin.dll")
#include "PWRT_api.h"
#pragma code()
PWRTLogout();
```

Displaying User Name in Picture

Configuration:

1. Configure static text field.
2. In the Properties Font> Attribute> Text, connect the system tag *@CurrentUser.*
3. Perform Login.
4. The user name is displayed in the static text field.

Modal Windows Message Box

- Action for example on Event Mouse-click

```
HWND Handle;
Handle=FindWindow(NULL,"Uhr");
if(Handle)
SetForegroundWindow(Handle);
else
ProgramExecute("C:\\\WINNT\\system32\\Clock.exe");
```

The **FAQ 415808** describes how a program window is called into the foreground using a WinCC Script.

Diagnosis Information in the Various LOG Files as of V3.x

As of Version 3.x, the system sets up files with diagnosis information in the case of errors. The system diagnostics manages the system messages and operator messages from the WinCC applications and forwards these to the WinCC message system for further evaluation.

The system messages are stored locally on every computer under the path ...\WINCC\DIAGNOSE in the CSV file format(comma separated value).

Various logbooks exist for the system messages:

1. System messages during startup : WinCC_SStart_xx.log
2. System messages during operation : WinCC_Sys_xx.log
3. License checking: Licence.log contains existing licenses
4. License checking: Authorinformation.inf contains missing licenses

(xx == 01..99)

The individual files of the logbooks are overwritten according to the cyclic buffer principle. That is, when the limit for the number of files set up is reached, the oldest logbook file is overwritten.This guarantees that the logbook only requires a specific maximum memory on the hard drive.
This is calculated as follows : Number of Files multiplied by Size of Files
The size and number of files can be separately set up for every logbook.
The settings are stored in the configuration file WINCC_LOG.INI in the WINCC\BIN path.
The size of the logbook files is set to 64 kbyte by default.

The structure of an operator message is as follows :
Date, Time, Message Number, Computer, User, Application, Old Value, New Value, Tag Status, Operator Reason

Tag status : 0 == denied, 1 == accepted

The structure of a system message is as follows:
Date, Time, Message Number, Class, Computer, User, Application, Error Text

Class : 1 == Information, 2 == Warning, 3 == Error

Using Any Project Function in Conjunction with "Loop in Alarm"

Using another function other than "OpenPicture", for example a project function, for "Loop in Alarm".

Example of a function that is to switch a picture in a picture window to visible.

```
void LoopVisible(Picture Picturename)
{
SetVisible("messagepicture","Picturewindow1",1);
}
```

Parameter explanation:
messagepicture : Name of picture which contains the picture window "Picturewindow1".
Picturewindow1: Name of picture window which is to be visible.
1: visible, 0: non-visible

This function must be created and integrated as a project function with Global Script. The assignment with the parameter "Picturename" is only for completeness, since Alarm Logging expects a parameter assignment that corresponds to the action "OpenPicture". The selected picture in the action in "Loop in the Alarm Configuration" has, for all that, no relevance and is only for completeness in the parameter assignment.

Alarm Report On per Mouse-click

for WinCC >=V5.0:

```
CMN_ERROR pError;
memset(&pError, 0, sizeof(pError));
MSRTActivateMProt(TRUE,&pError);
```

Alarm Report Off per Mouse-click

for WinCC >=V5.0:

```
CMN_ERROR pError;
memset(&pError, 0, sizeof(pError));
MSRTActivateMProt(FALSE,&pError);
```

Ruler for Trend Window Switch-on/Switch-off per Mouse-click

TlgTrendWindowPressLinealButton("TrendWindowTemplate_Toolbar");

TrendWindowTemplate_Toolbar=Window name of Trend Window Template

Further Processing of Values for Ruler (for Trend Window)

Example: Output via an output field
{
double valueoff;
//0=Trend1
valueoff=TlgGetRulerValueTrend("TrendWindowTemplate_Toolbar",0);
return(valueoff);
}

Information on the SIEMENS Authorization Tool

Siemens supplies part of its software with a copy protection. The protected programs will only run if an authorization for the program or software package has been installed on the hard disk drive of the relevant PG/PC.

The AuthorsW program is installed along with the Setup of WinCC as of Version 5.0. It can be run in German, English, French, Spanish and Italian. You use AuthorsW to install and uninstall authorizations.

You can also use the new AuthorsW program to install and uninstall authorizations from previous versions.

You will find an exact description of the Authorization Tool on the WinCC CD in the directory *Documents-English-WinCCInformationSystem.chm*.

You start the tool through the Start Window.

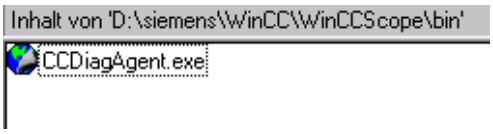
Diagnosis and System Information with WinCC Scope

Objective Diagnosis and system information using hardware and software.

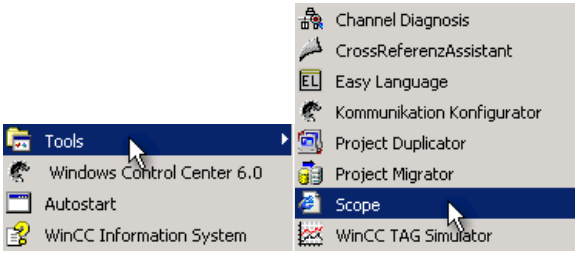
Access Access to the diagnosis and system information data is **local** and is **possible** via the **Intranet/Internet**.

Information
-System configuration of the computer.
-WinCC version information
-WinCC settings (project information, ...)
-Overview of installed licenses, ...

Start Server

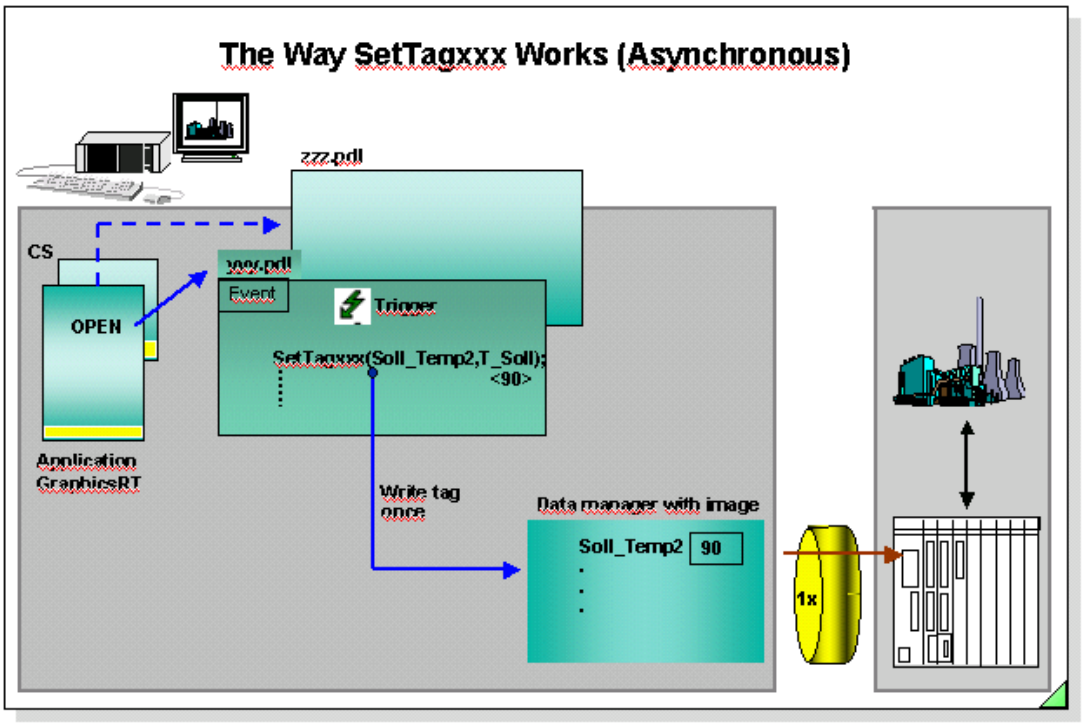


Start Scope



Note If you have a stand-alone computer and want to use this WinCC application, you must install the Microsoft Loopback Adapter. Please see FAQ Publication ID: 553872
The **execution** of SQL instructions is possible.
A backup copy should be made before every access to the database!

SIEMENS



Objective To understand the way different SetTagxxx functions work.

Task In the Global Script of WinCC there are various functions for writing tag values. This gives the user greater flexibility for solving tasks.

SetTagxxx The picture yyy.pdl is called using the Graphics Runtime application. In this picture, an event in a graphics object triggers the SetTagxxx function call, if the 'if' query is fulfilled.
Example:

```
if(var1>500)
SetTagxxx(Soll_Temp2,T_Soll);
```

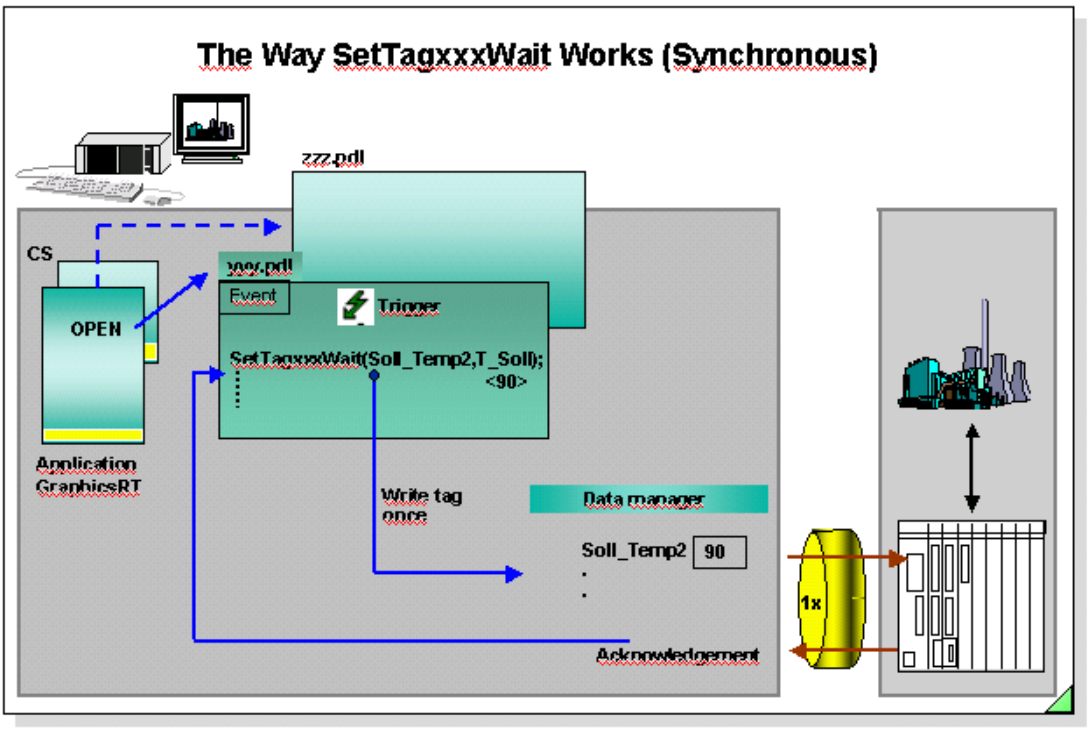
Note: xxx=data type

The SetTagxxx function then gives the data manager a job to write a value once. That way, the tag is known in the data manager for the entire execution time (as long as the picture yyy.pdl is selected) and can be used for further evaluations. The function passes the value to the data manager. The calling function does not wait until the value is actually written. When the picture yyy.pdl is replaced with the picture zzz.pdl, all tags are logged-off with the data manager.

Properties
-The call is fast.
-The calling function does not know when the value was actually written (asynchronous writing).
-The function does not provide any information on the status of the writing job.
-If the tag is also used for output objects, it is updated cyclically.

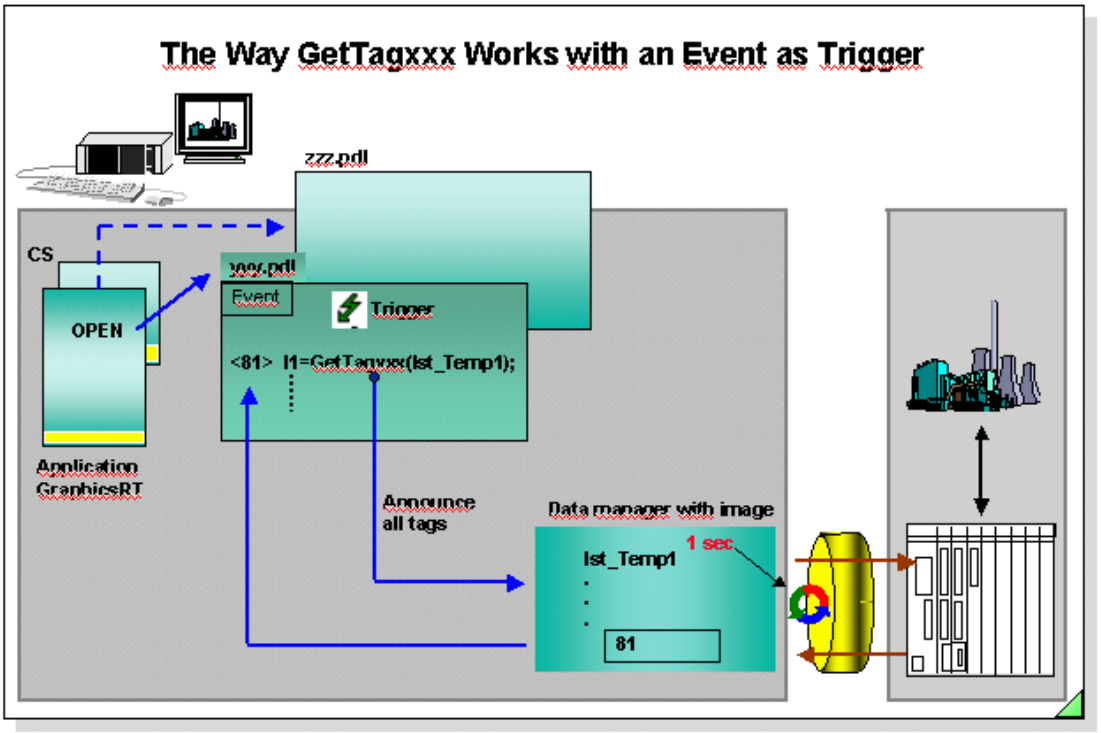
SetTagxxxState The function has the same characteristics as SetTagxxx. The SetTagxxxState function provides additional information about the status of the writing job. Since the status is always provided internally, there is no differentiation in performance to SetTagxxx.

SIEMENS



Objective	To understand the way different SetTagxxxWait functions work.
Task	In the Global Script of WinCC there are various functions for writing tag values. This gives the user greater flexibility for solving tasks.
SetTagxxxWait	The picture yyy.pdl is called using the Graphics Runtime application. In this picture, an event in a graphics object triggers the SetTagxxxWait function call, if the 'if' query is fulfilled. Example: if(var1>500) **SetTagxxxWait(Soll_Temp2,T_Soll);** Note: xxx=data type The SetTagxxxWait function then gives the data manager a job to write a value once. The function passes the value to the PLC and waits until the value is actually written before returning to the calling function. Afterwards the tag is no longer known in the data manager.
Properties	-The call takes longer when you compare it to SetTagxxx. The duration also depends on the channel and the PLC, among other things. -The calling function waits until the value was actually written (synchronous writing). Time out 10 seconds. -The function does not provide any information on the status of the writing job. - When the picture yyy.pdl is replaced with the picture zzz.pdl, all tags are logged-off with the data manager.
SetTagxxxStateWait	The function has the same characteristics as SetTagxxxWait. The SetTagxxxStateWait function provides additional information about the status of the writing job. Since the status is always provided internally, there is no differentiation in performance to SetTagxxxWait.
SetTagMultiWait	The function has the same characteristics as SetTagxxxWait. It does, however, offer the opportunity to be able to write several tags in one function call.

SIEMENS



Objective

To understand the way the GetTagxxx function works with an event trigger.

Task

In the Global Script of WinCC there are various functions for reading tag values. This gives the user greater flexibility for solving tasks.

GetTagxxx

The picture yyy.pdl is called using the Graphics Runtime application. In this picture, the GetTagxxx function is called using an event.

Example: **I1 =GetTagxxx(lst_Temp1);**
Note: xxx=data type

if(I1 >100)

The GetTagxxx function is introduced with the passing of the parameter “1 second“ to the data manager during the first call and from then on is cyclically fetched by the PLC. This makes sure that the value is available with every further call of the function. The tag is known in the data manager for the entire execution time (as long as the picture yyy.pdl is selected) and can be used for further evaluations. When the picture yyy.pdl is replaced with the picture zzz.pdl, all tags are logged-off with the data manager.

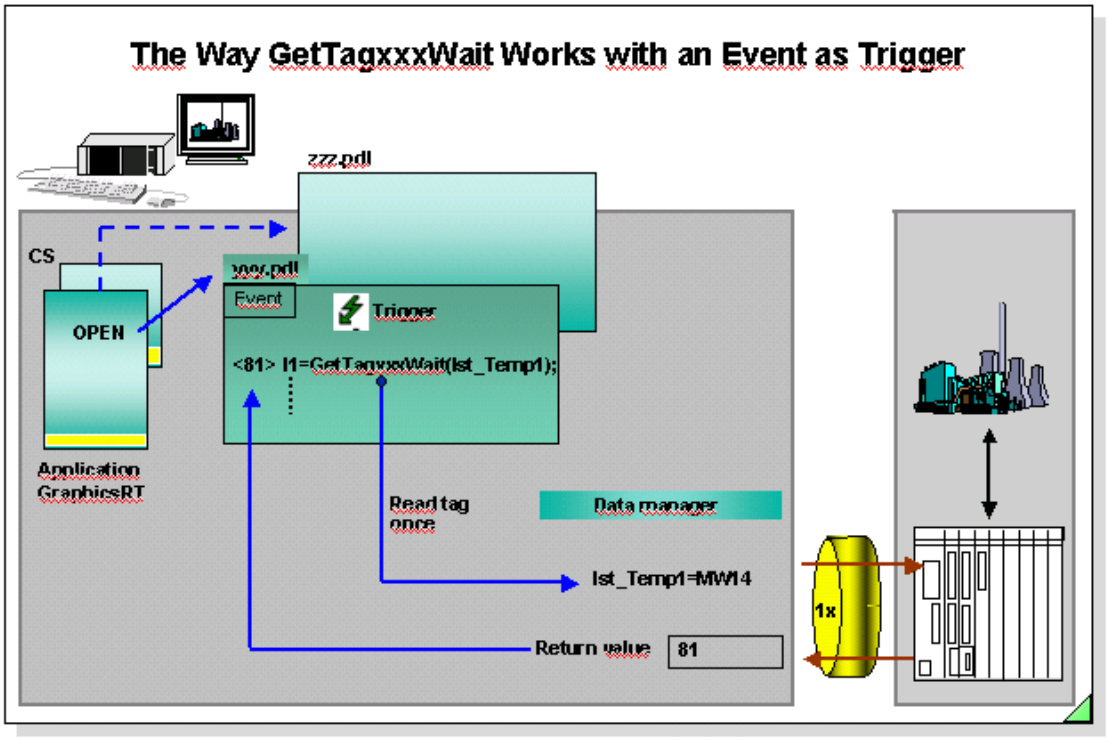
Properties

-The same as GetTagxxx.
-The calling function is processed with every trigger.

Note

If the event ‚mouse click‘ is only seldom triggered, then it is recommended that the GetTagxxxWait function be used.

SIEMENS



Objective To understand the way the GetTagxxxWait function works with an event trigger.

Task In the Global Script of WinCC there are various functions for reading tag values. This gives the user greater flexibility for solving tasks.

GetTagxxxWait The picture yyy.pdl is called using the Graphics Runtime application. In this picture, the GetTagxxxWait function is called using an event.

Example: **I1 =GetTagxxxWait(lst_Temp1);**
Note: xxx=data type

if(I1 >100)

The GetTagxxxWait function requests the data manager once to read a value. The data manager fetches the function value from the PLC. The function waits until the value was read. The tag is no longer known in the data manager after the function is processed.

Properties

-The call takes longer when you compare it to GetTagxxx. The duration also depends on the channel and the PLC, among other things.
-The function waits until the value was actually read (**synchronous reading**). Time out 10 seconds.
-The function does not provide any information on the status of the reading job.

GetTagxxxStateWait The function has the same characteristics as GetTagxxxWait. The GetTagxxxStateWait function provides additional information about the status of the tag that is read. Since the status is always provided internally, there is no differentiation in performance to GetTagxxxWait.

GetTagMultiWait The function has the same characteristics as GetTagxxxWait. It does however offer the opportunity to be able to read several tags in one function call.

C Operators

Operators:	Name:	Example:
&	Binary AND Operator	C=A&B In the variable name C, only bits that are set in A and in B are set - A=01001110 B=10100111 ---------- C=00000110
|	Binary OR Operator	C=A|B In the variable name C, all bits that are set in A or in B are set - A=01001110 B=10100111 ---------- C=11101111
^	Binary Exclusive OR Operator	C=A^B In der variable name C, all bits from A are inverted, that have the value 1 in B- A=01001110 B=10100111 ---------- C=11101001
~	Binary NOT Operator	C=~A In the variable name C, all bits from A are inverted- A=01001110 ---------- C=10110001
==	Compare for Equality Example: if(A==10){Then} else{otherwise}	C=A==B The variable name C is TRUE if the variables A and B have an identical value, otherwise C = FALSE (0) A=10 A = 10 B=22 B = 10 ---------- ---------- C=0 C = TRUE
&&	Logical AND Operator Example: if(A&&B){Then} else{otherwise}	C=A&&B The variable name C is TRUE if the variables A and B both have a value that is not 0, otherwise C = FALSE (0) A=10 A = 10 A = 0 B=22 B = 0 B =99 ---------- ---------- ---------- C=22 C = 0 C = 0
||	Logical OR Operator Example: if(A||B){Then} else{otherwise}	C=A||B The variable name C is TRUE if at least one variable A or B has a value that is not 0, otherwise C = FALSE (0) A=10 A = 0 A = 0 B=22 B = 33 B = 0 ---------- ---------- ---------- C=1 C = 33 C = 0
!	Logical NOT	C=A NOT, The neg. value of A is in the variable name C

VB Operators

Operators:	Name:	Example:
AND	Binary AND Operator	C=A AND B In the variable name C, only bits set in A and in B are set- A=01001110 B=10100111 ---------- C=00000110
OR	Binary OR Operator	C=A OR B In the variable name C, all bits set in A or in B are set- A=01001110 B=10100111 ---------- C=11101111
XOR	Binary Exclusive OR Operator	C=A XOR B In the variable name C, all bits from A are inverted, that have the value 1 in B A=01001110 B=10100111 ---------- C=11101001
NOT	Binary NOT Operator	C=~A In the variable name C, all bits from A are inverted- A=01001110 ---------- C=10110001

=	Compare for Equality
	Example:
	if(A=10) Then
	...
	Else
	...
	End If
		C=A=B The variable name C is TRUE if the variables A and B have an identical value, otherwise
		C = FALSE (0)
		A=10 A = 10
		B=22 B = 10
		---------- ----------
		C=0 C = TRUE

AND	Logical AND Operator
	
	Example:
	If(A AND B) Then
	...
	Else
	...
	End If
		C=A AND B The variable name C is TRUE if the variables A and B both have a value that is not 0, otherwise C = FALSE (0)
		
		A=10 A = 10 A = 0
		B=22 B = 0 B =99
		---------- ---------- ----------
		C=22 C = 0 C = 0

OR	Logical OR Operator
	
	Example:
	If(A OR B) Then
	...
	Else
	...
	End If
		C=A OR B The variable C is TRUE if at least one variable A or B has a value that is not 0, otherwise C = FALSE (0)
		
		A=10 A = 0 A = 0
		B=22 B = 33 B = 0
		---------- ---------- ----------
		C=1 C = 33 C = 0

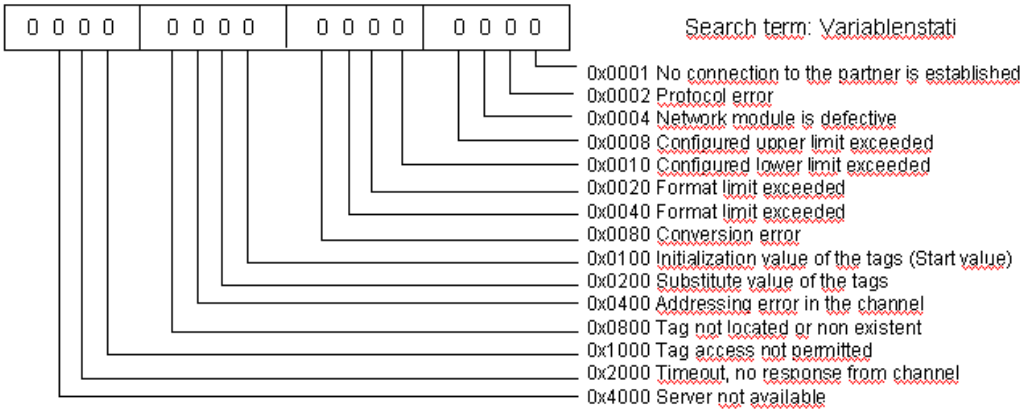
SIEMENS

Connection Status: Example of the Get/Set Tag State 1/2

BOOL SetTagDWordState (Tag Tag_Name, DWORD value, PDWORD lp_dwstate)
DWORD GetTagDWordState(Tag Tag_Name, PDWORD lp_dwstate);

Example: SetTagDWordState ("P_Variable1", 100, &Status);
GetTagDWordState ("P_Variable2", &Status1);

the returned bit register can be analyzed according to the following schematic.



General Information If process tags or even internal tags are to be used in actions, you must read the values via GetTag actions.
Permanent monitoring of the process connection from the PLC to WinCC can be implemented with the GetTagState action (for example, a warning box pops up in case of a fault).
Set Tag actions are made available for assigning the results of calculations.
The SetTagState action is available if the result of the SetTag action is to be checked.
The success of a value change can be queried with the SetTagState action (writing from WinCC to the PLC). This is especially important for process tags because the most varied coupling faults can occur there.

The status is transferred as a bit register and can be either evaluated by the Dynamic Dialog or by the status' own actions.

Evaluation

-available as system function by displaying a shading. No configuration required.
-definition of a substitute value that pops up when there is a connection error. Setting via Parameters.
-the status of a process tag can also be evaluated with the Dynamic Dialog (see the Chapter Graphics Screens). Configuration with limited possibilities for making dynamic.
-available as standard function in the option package Basic Process Control Package.
-configuration using C-Scripts, see example above.
-configuration with Toggle in the Simatic program, for example.

Connection Status: Example of the Get/Set Tag State 2/2

0000000000000000 Type O

Configuration:
-Properties/ I/O field/ output/input/
output value:**i_lp_dwatate**

no Connection

Configuration:
-Static text/ colors/ fill pattern color/
dynamic/ C-Action:

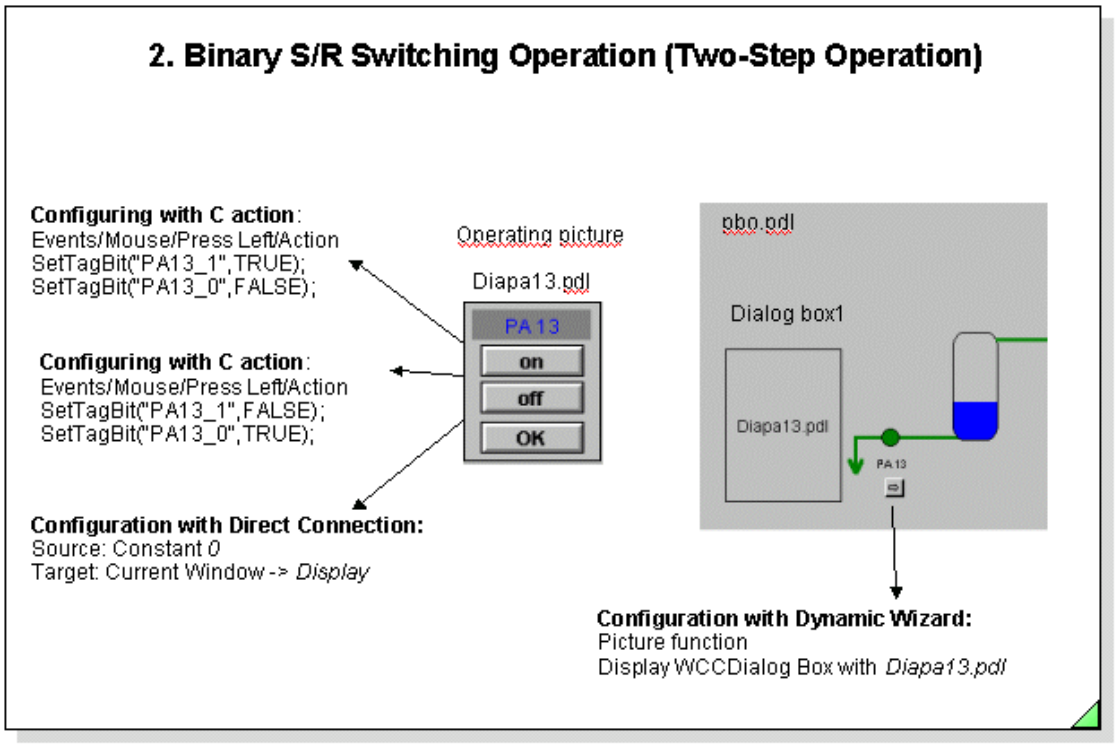
```
int var1,var2;
var1=GetTagWord("i_lp_dwatate");
var2=var1&0x0001;
if(var2==0) {
    SetBackColor(lpszPictureName,
                 lpszObjectName,CO_GREEN);
}
else {
    SetBackColor(lpszPictureName,
                 lpszObjectName,CO_RED);
}
return(var1);
```

0 Type O

Configuration:
-Properties/ I/O field/ output/input/
output value/ dynamic/ C-Action:

```
int var1,var2;
unsigned long var3;

var1 = GetTagWordState("wert1",&var3);//address

SetTagWord("i_lp_dwatate",(WORD)var3);//content

var2=GetTagWord("i_lp_dwatate");

if(var2!=0) {                    //means unequal
    SetVisible(lpszPictureName,"PictWindow1",1);
}
else {
    SetVisible(lpszPictureName,"PictWindow1",0);
}
return(var1);
```

SIEMENS



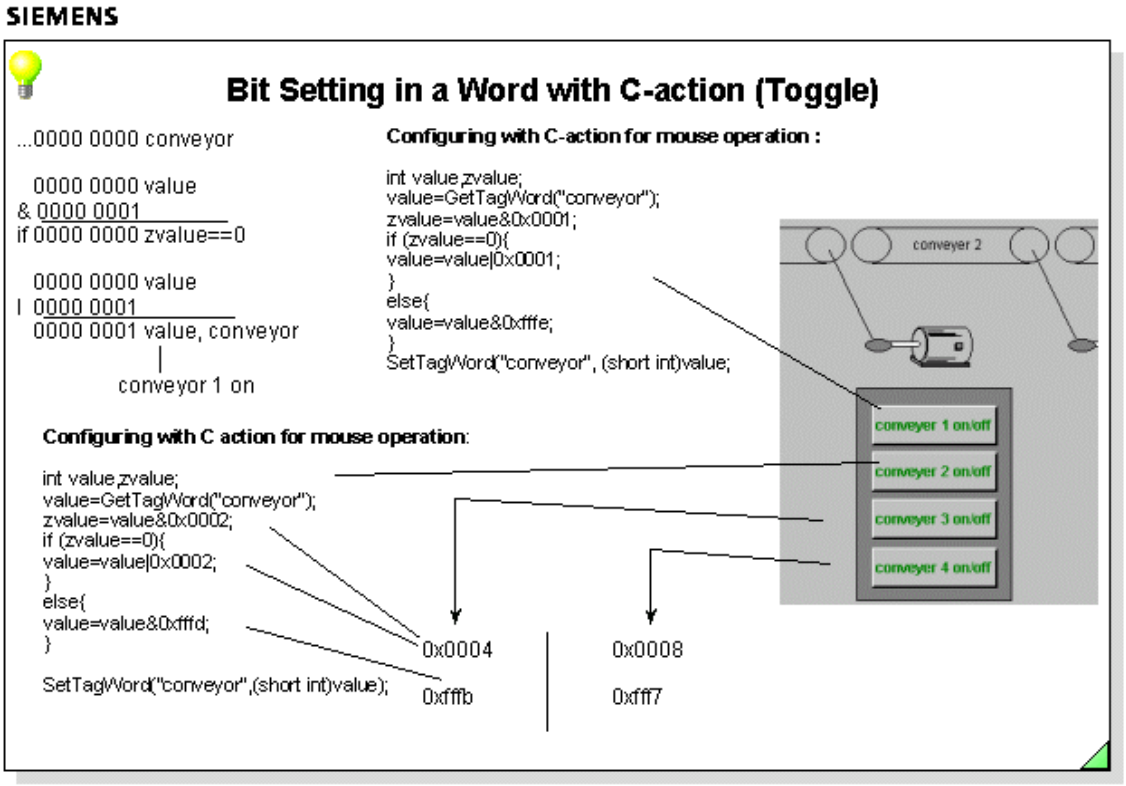
Task Switching a valve on and off is to take place in a two-step operation.

Example A two-step operation (1. Select Button PA13 and 2. "on" or "off" button) is to be carried out using a temporarily displayed picture. The binary state of a tag is set to 1 with the "on" button while at the same time the binary value of a second tag is set to 0.
This procedure takes place in a reverse sequence when the "off" button is operated. This switching operation is used in the programmable logic controller for setting and resetting flip-flops. The "OK" button is used to close the picture. The switching operation corresponds to the binary value of one of the tags.

Procedure 1.Configure the operating picture as shown in the slide above. The operating picture has three buttons. Two of the buttons are for the switching operation, and the third button is for closing the operating picture. When you create the Dialog box, the size of the operating picture (x- and y dimension) and the dialog box must be the same.
2.Configure a button in the picture for calling the dialog box. The size of the dialog box (x and y dimension) must be the same as the operating picture. You use the Dynamic Wizard to generate the dialog box.

Two binary type tags must be created for the value transfer.
If you want to have a title in the dialog box, the height of the dialog box must be 10 pixels more than the height of the .PDL file that is displayed in the dialog box.

Note The SetTagBit **cannot** be used for S5 switching operations since the handling blocks transfer the data word by word. S7 coupling is an exception. Remedy: Use bit setting in a word.



Task

Four assembly conveyors are to be switched with four buttons via a bank of switches. An assembly conveyor motor is assigned to each button. Each motor can be switched on or off. The same process tag (conveyor) is to be used for the switching operation. This ensures that the memory locations in the PLC are optimally used.

Example

Each switching operation for a motor leads to setting and resetting a bit in a word. Bit addresses 0 to 3 are assigned. Bit addresses 4 to 15 can be used in the controller program for additional operations.

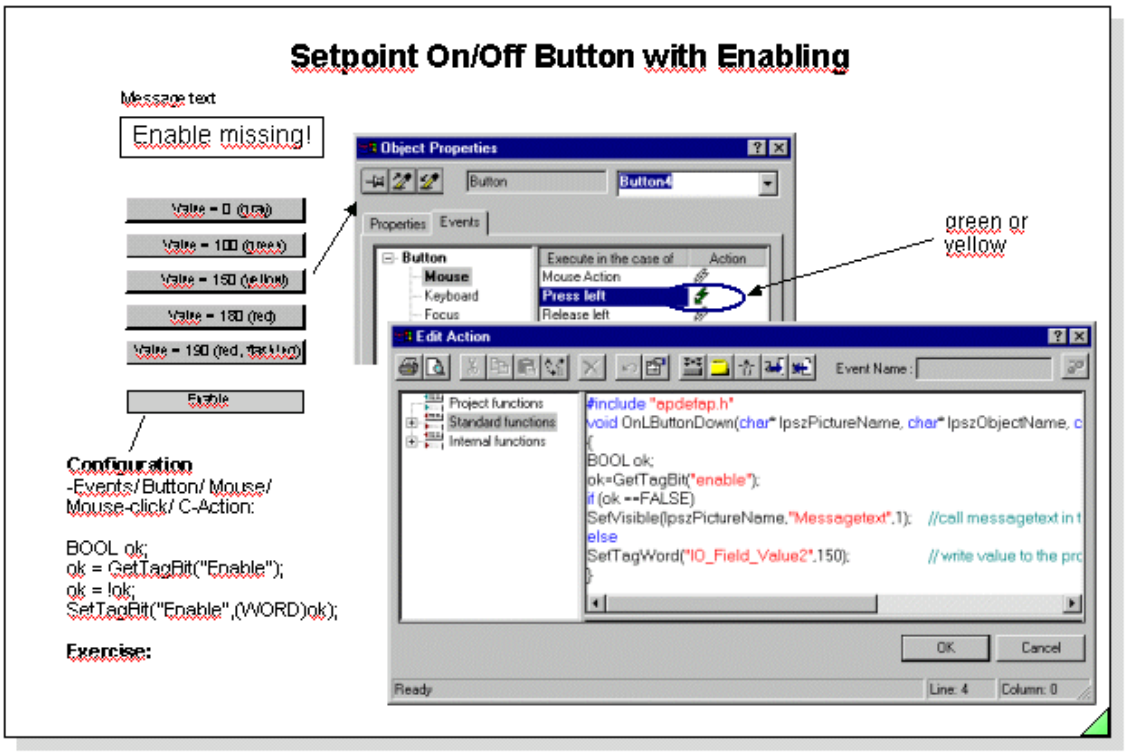
Procedure

The "conveyor" tag is defined as an unsigned 16-bit value in the data manager. The current status is loaded from the PLC into a "value" auxiliary storage with each processing of the action list.
After an AND masking, the respective bit address is scanned, and a decision is made as to whether the bit is to be set with an OR masking or whether the bit is to be reset with an AND masking. Create the static and dynamic screen components according to the screen template in the slide above.

Legend

|	OR masking
&	AND masking
0xfffe	Hexadecimal value
==	Equals

SIEMENS



C Functions

Very extensive actions, checks, and scans can also be processed with C functions.
In addition to the standard C functions (ANSI C), there are, of course, still the WinCC-specific functions available for reading and writing tags and for processing all possible objects.

Configuring

C functions can be used where the direct connection to individual tags is not sufficient, or even where several properties are to be changed simultaneously.
With the help of the C functions, you obtain an extensive comprehension of all the object properties available, and of all the WinCC tag contents.

Note

If a C action is configured for an event, a **green lightning bolt** symbol represents the action that is loaded.
If the C action has not yet been compiled, the **lightning bolt** symbol is displayed in **yellow**.
If the condition of a bit is used in an IF statement, the condition is either TRUE or FALSE (FALSE is equal to zero, TRUE is not equal to zero).

The type and number of properties of the object depends on the selected object.

The action is executed when the selected property changes, or by mouse click/keyboard.

SIEMENS

Incrementing/Decrementing in a Specified Value Range

I/O Field2 **1234**

Message text

Value range from 0 to 100!

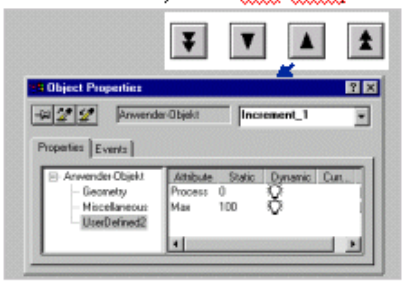
Button **L+/R-**

Making dynamic with Configuration Dialog

Configuring with C-action :
-Events/ Mouse/Press Left/ Action:
```
WORD Temp;
Temp = GetTagWord("value");
Temp = Temp + 1;
if (Temp>100)
{Temp=100;
SetTagWord("value",Temp);
SetVisible(lpszPictureName, "helptext",1); }
else {
SetTagWord("value",Temp);
SetVisible(lpszPictureName, "helptext",0);
}
```

Configuring with C-action :
-Events/ Mouse/Press Right/ Action:
```
WORD Temp;
Temp = GetTagWord("value");
 Temp = Temp - 1;
;
```

Alternative: Symbols from library



Task

Necessary precision settings for a motor's setpoint speed are to be made by mouse operation, whereby the speed's lower limit and upper limit are not to be exceeded. If this occurs, a message text automatically appears.
Use the left mouse button to increase the speed. Use the right mouse button to decrease the speed setting. The input/output field can be used for less precise settings.

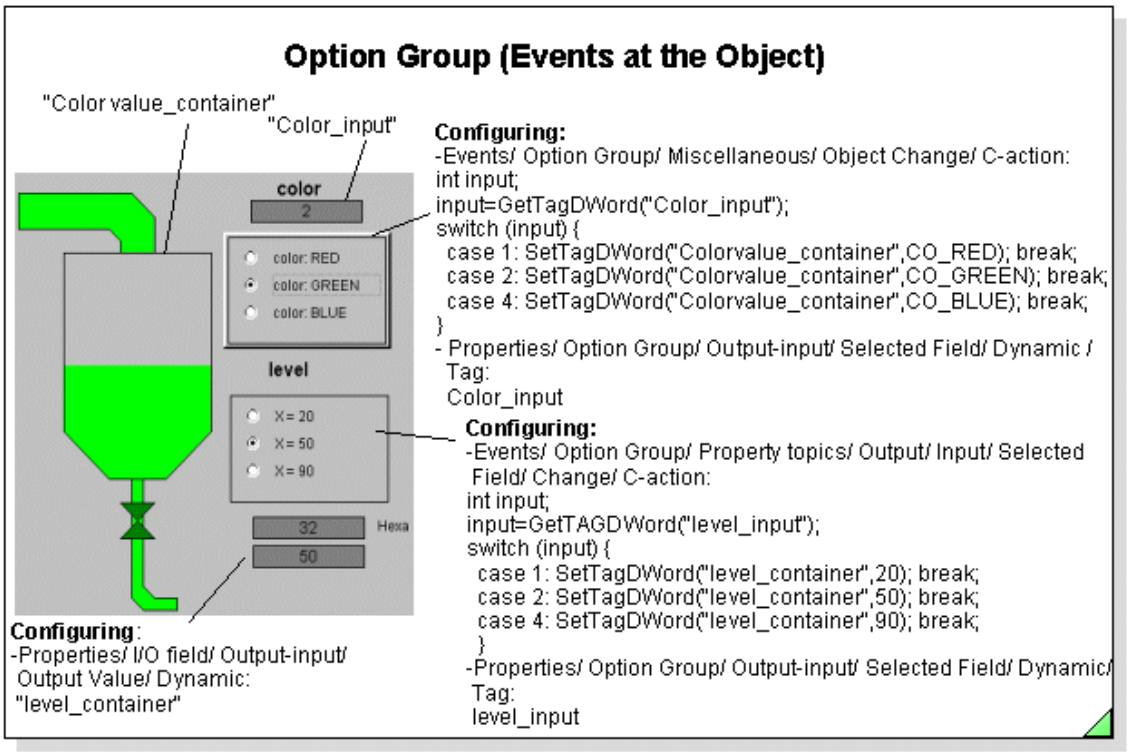
Example

The motor's setpoint speed is to be between 0 and 100.
The process connection takes place with the "value" tag.

Procedure

Configure a button and an I/O field in your picture according to the template in the slide above. Action configuring is displayed in the slide above.

SIEMENS



Task

The contents and the level of a container are to be specified via two option groups with three setting possibilities each. At the same time, the adjustment value is to be displayed via output fields, and the color attribute is to be displayed via the graphics.

Example

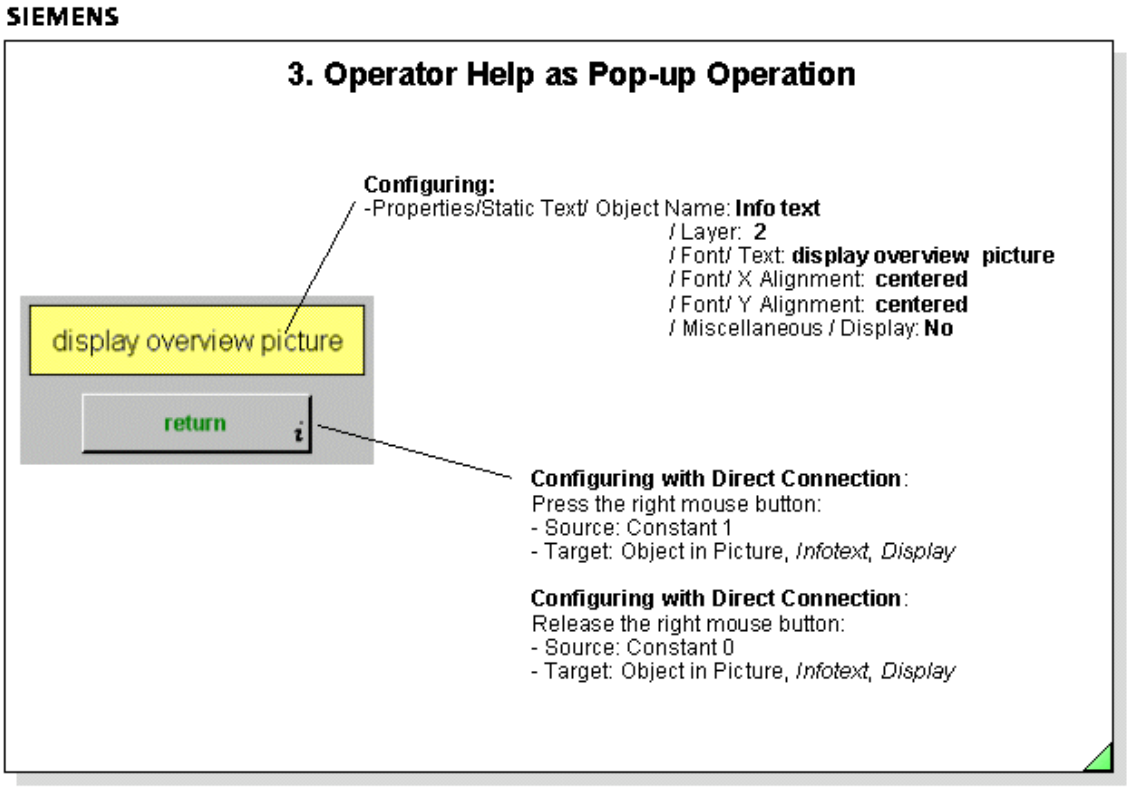
The display of a container with various contents can also be optically combined with the display of the fill level by using a color tag.
A polygon is to be used to display the container. The level is displayed in the polygon and is output as a value in an output field.
The background color depends on the contents. The binary value of the "Color_input" tag is scanned in the controller for measuring out the contents and is also displayed in an output field.
The value of the "level_container" tag is displayed in an output field and is used for filling the container.

Procedure

The "level_container", "level_input", "Colorvalue_container", and the "Color_input" tags are defined in the data manager. This tag must always be of the signed 32-bit value type for an option group. Use the screen template above for configuring the static and dynamic screen components.

Note

The tag used must be a signed 32-bit value data type and the tag occupies one doubleword. Each individual selection in the check box occupies one bit in the doubleword. In a multiple selection, the bit to the option button selected last is always set. All other bits are reset.

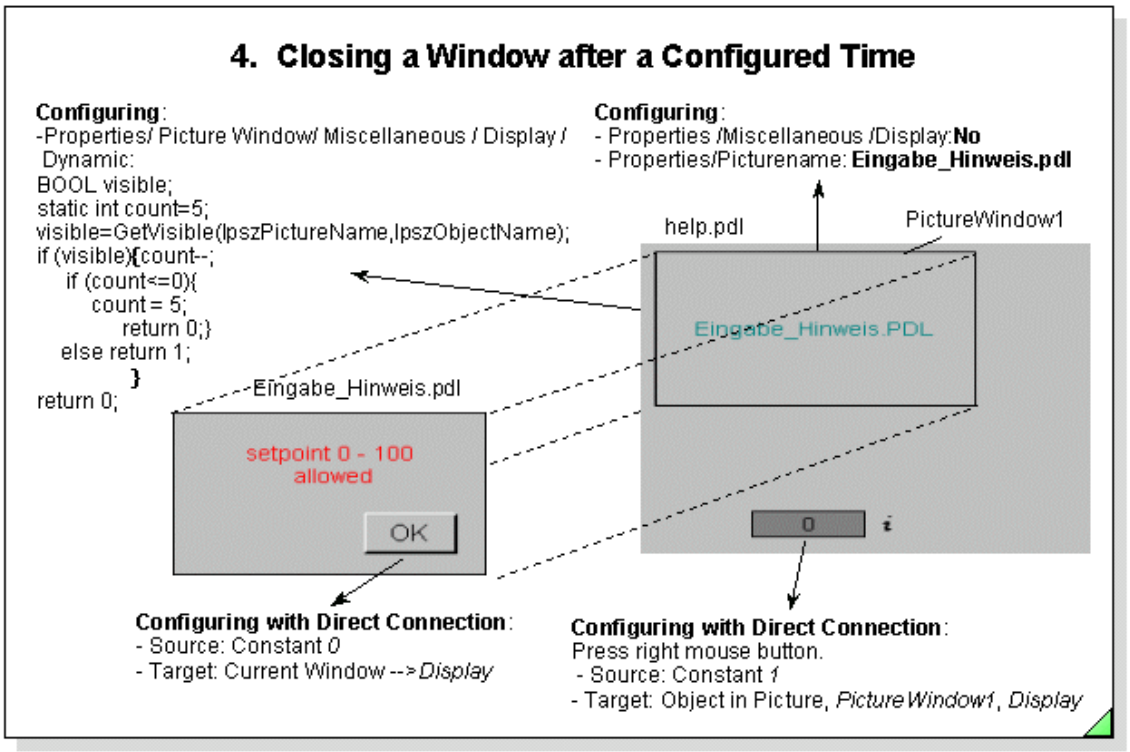


Task A help text is to be displayed with the right mouse button for the duration of the operation. This is also called pop-up help. Action configuring of the left mouse button for the screen deselection is not taken into consideration in this example.

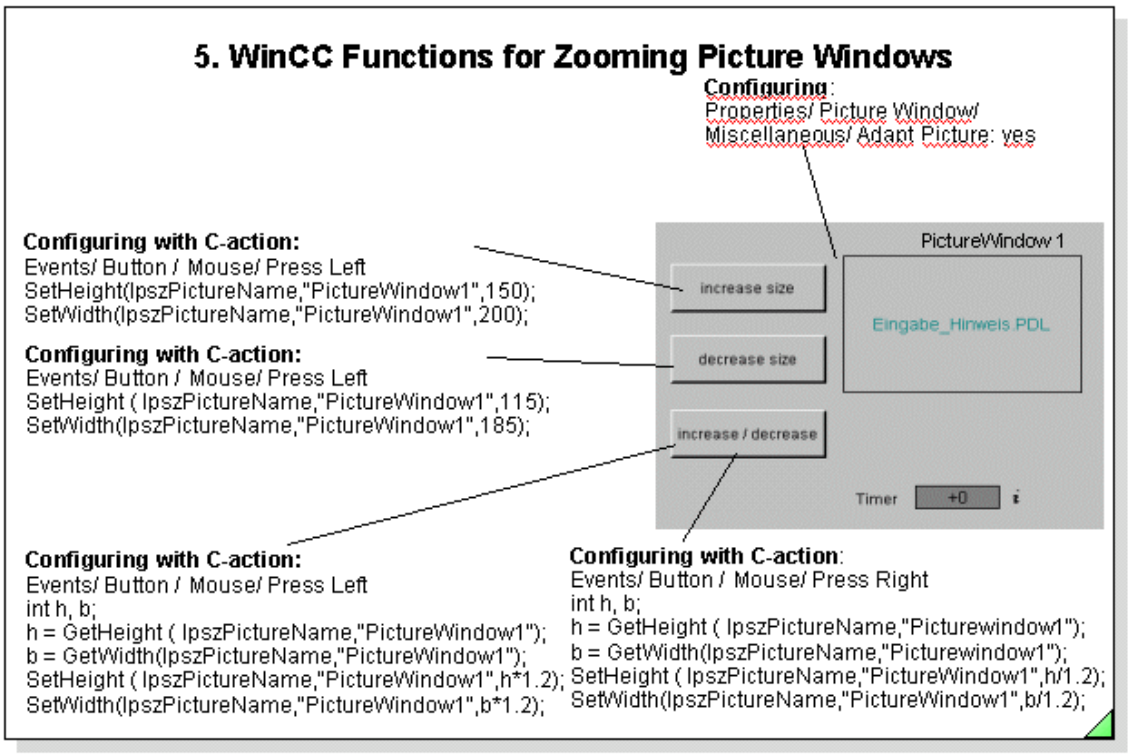
Example The help texts are determined by the technological application and give the operator information about steps that are executed after an operation, for example.

Procedure Create a text field and an operator button. Label the button according to the template in the slide above. An additional symbol must identify that the right mouse button (pop-up menu) can be used here.
Action configuring is displayed on the screen in the slide above.

SIEMENS



Task	An operator aid is to be closed manually or after 5 sec.
Example	The duration results from the number of increments and the update of the picture object.
Procedure	A C-action editing must be selected that provides a time trigger for the action processing. The C-action is displayed in the upper left part of the slide. A tag must be supplied with the addition of a "static" for saving the incremental value.
Static Tag	A local tag, supplied with the addition of a "static", retains its value. The tag is **not** deleted between two function calls in which the tag is defined.



Task

The picture window is to be zoomed with the "increase size" button. The picture window is reset to the configured size with the "decrease size" button.

Procedure

Configure two buttons corresponding to the screen template in the slide above. Then configure the event by operating the mouse. Use the WinCC functions in the action list for this purpose:
Configuring:
-Internal functions/ graphics/ set/ geometry: **SetHeight**

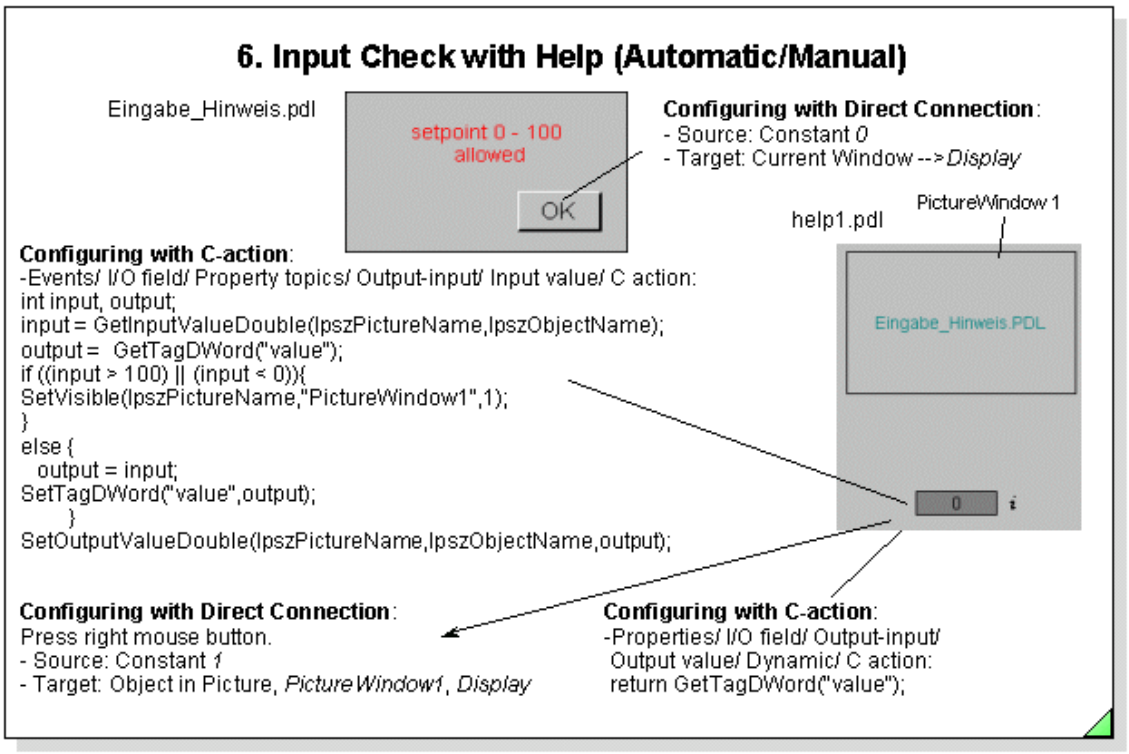
: **SetWidth**

The configured size of picture window1 can be determined via the geometry of picture window1.

Configuring:
-Internal functions/ graphics/ get/ geometry: **GetHeight**

: **GetWidth**
The thus determined width and height of the picture window can be enlarged or reduced by a factor and reset.

SIEMENS



Task The input of a setpoint is to be checked before it is made available to the controller.

Example The input of the setpoint for the container level is to be limited to the range of 0 to 100.
If the range is exceeded or underranged during the input, the value is not transferred.
In addition, help for the operator is to be displayed.

Procedure The input of the value is implemented using an I/O field. This I/O field is connected by means of several actions.
When an input is changed, the action for testing the input is called. The input value is checked for limits. If the limits are exceeded, a help text box is displayed. This help text box is displayed in a picture window. The help text box that is displayed for an incorrect input is stored in picture window 1.
You can also use the right mouse button to display this box.

Manual Help Event/ I/O field/ Mouse/ Press Right/ C-action:
SetVisible(lpszPictureName,"PictureWindow1",1);

Function Expansion WinCC V1.11

In addition to the function scope of V1.10, WinCC V1.11 contains the following expansions.

- System software in three languages (German, English, French)
- Online language switch-over
- Report system with project documentation (Picture Documentation and Info)
- Password protection for online parameter assignment of Tag Logging
- Serial communication via AS511 to S5 and SIMATIC 505
- DDE Server for data exchange with Windows programs
- Options: User archives, ODK (ß-Version), Basic Process Control (PCS7) and Channel DLLs for other manufacturer‘s controllers
- Powerpacks for the upgrading of the usable tag number

Function Expansion WinCC V3.0

In addition to the function scope of V1.11, WinCC V3.0 contains the following new functionality.

WinCC Multi-user system as Client /Server solution:

- Server under Windows NT 4.0
  (Windows NT with Workstation or Server licence (10 or 64 PCs on the network)
  (WinCC Server Option necessary)
  (At least 64 MByte main memory recommended)
  (Central data management on the Server, Online configuration possible)
- Clients under Windows 95 or Windows NT
  (Client has access to all tags of the Server)
  (Necessary licence for Client: RT 128 or RC 128)
- Server/Client Network (Terminal bus)
  (PC-LAN with TCP/IP or NetBUI protocol (e.g. 3COM cards)
- Multi-user configuration: Server with operator terminal (max. 4)
- Multi-user configuration: Server without operator terminal (max. 8)

Integration in PCS7:

- Process Control System7, the control system based on selected SIMATIC standard components
- Option packages PCS7 (for Windows 95 / NT)

Function exapansion in all Modules

Control Center:

-Central storage of all project settings for the project
(Simple change between different projects)
- Structured Tags
  (For easier handling of related process data)
- Referenced text object
  (Text output in I/O fields from the multi-lingual text library)
- User-friendly tag selection dialog
  (With filter and sorting according to connection, group and type)
- Tag Import/Export
  (Tool for importing/exporting tags and connections according to CSV)

Graphics Designer:

- Runtime button in the toolbar
  (Direct change into Runtime, to the process picture currently being processed)
- Identification of dynamic properties and objects
  (Simple finding of actions and tag connections)
- Multiple selection of different objects
  (Assign or inherit common properties (e.g.:Button with multi-line text)
- Bit selection in the dynamic dialog (reading the bit)

(Easier configuration without Script, faster processing)
- Bit selection via Dynamic Wizards (writing the bit)
(Bits for setting / resetting controller)
- Tab cursor control directly via the keyboard
(For navigation between I/O fields)
(User-friendly operation without the mouse via freely selectable keys)
- Set input focus
(For operation without the mouse in several part pictures)
- ActiveX support
(Integrating the "new“, performer Microsoft Controls)

Report Designer:

- Tags in the user protocol
(Integrating current process values or values from internal tags in the user protocol)
- Direct Picture Documentation and Info from the Editors
- Event-driven report start
(Start because of an operator operation)
(Start because of a value change in the process)
(Configurable using a function call)

Alarm Logging:

- Message window updating
(The message window from the short-term archive is updated)
- Actions to messages can be triggered
(Action to coming, going and acknowledge event can be configured)
(Configurable via a standard function in the Global Scripts)
- Acknowledgement of the Group Messages from the PLC
(Including the single messages contained therein)

Tag Logging:

- User-friendly trend window operation
(Zoom and Lineal (Ruler) can be operated in parallel)
(Interpolation of trend measured values)
(Continuous value display in Lineal (Ruler))
(Reload archive data while opening picture)
(Fill trend areas)
(Updating the compression archives)
- Simplified configuration for user archives
(Standard Script functions for user archives)

Increased Performance through Optimization:

- Faster picture selection through optimized data storage
(Only Delta info of dynamic objects is saved)
(Compiled action scripts arranged in tandem in the file)
- Faster graphic updating of objects
(Internal cache for referenced picture data (EMF, WMF, BMP))
- Faster trend output
(Reload archive data while opening picture)
(Optimization in opening / scrolling of measured values)

Additional coupling possibilities (SIMATIC S5 /S7 /TI)

Additional coupling possibilities (DDE, ... other driver)

Function Expansion WinCC V3.1

In addition to the function scope of V3.0, WinCC V3.1 contains the following expansions.

General Functions

– Performance improvement in system startup
- Executable under Office 97 (OLE,ODBC)
- Faster in the system startup

Totally Integrated Automation

- Synchronization with STEP 7 Level 3.1
  -- Tags, Messages
  -- Communication
- Synchronization with PCS7 Option Packages
  -- STEP 7, SCL, CFC, SFC, WinCC
- Mapper for tags and control system messages
- S7-PMC communication

User Object

-Grouping of graphic objects into a new object
-Definition of object interfaces for dynamics
-Simple process connection, only the defined object interfaces are visible on the user interface
-Configuration and online language switch-over
-User object with its own encapsulated processing

Text Lists Object

-For assignment of texts to values (binary or decimal)
-A text input (via a selection list) initiates a value output
-Because of a value input, a text output results

Direct Connection of Operating Elements to Tags

Expanded Configuration of the Status Display

-Assignment from 0 to 2^{32} -1
-Gaps between individual states is possible

Selection of Structured Tags in the Dynamic Dialog

-Editing of tag names that do not correspond to the C-Syntax

Tag Logging

-Functions for operator control of the trend and table windows via its own buttons
-Standard functions for triggering the buttons in the toolbar of the trend and table windows
-User archive Wizard for creating operator buttons and fields for any user archives

Function Expansion WinCC V4.0

In addition to the function scope of V3.1, WinCC V4.0 contains the following expansions.

Graphics Designer

Tooltiptext:
Object property Tooltiptext (ToolTipText) that can be made dynamic, for displaying a "Tooltips" at runtime.
Expanded Toolbar:
The library can now also be called via a button in the toolbar.
Direct Connection:
"This Object" in the list of exisiting objects in the picture. All dynamics are also copied during copying. If a source or target object is not referenced by name, rather by the entry "This Object", then the copy also has this direct connection.
Dynamic Wizard:
The Dynamic Wizard is implemented as overlayable and maskable toolbar and is thus always available.

Global Script

New Standard Functions:
New functions for trend and table windows
Additional key functions of the toolbar for trend and table windows are now also available as standard functions.
A new function for developing your own error handling.
New functions for the Report Designer
New Internal Functions:
Functions for reading object properties.
Functions for writing object properties
Function for reading the values and states of several tags (GetTagMultiWait)
Function for reading the values and states of several tags (GetTagMultiStateWait)
Functions for writing the values and states of several tags (SetTagMultiWait)
Funstions for writing the values and states of several tags (SetTagMultiStateWait)

Alarm Logging

Limit-value Monitoring:
A COM object can be used for the limit-value monitoring of analog tags.
Message Window as OCX:
The display of messages can now also take place in an OLE-Control object of the type "WinCC Alarm Control".

Tag Logging

WinCC under NT:
WinCC now works under NT not only in the Administrator mode.
Dialog Times Expanded:
In the establishment of archiving times, points in time can now also be specified when an archiving is to be performed for the first time.
Moreover, an archiving, independent of the specified timing code, can be triggered during startup of the system and during exiting of the runtime system.

Display Forms:
Ruler visible while opening picture,
Display trends staggered,
Recorder function,

Color Assignment with Limit-value Violations:
When there is an overrange or underrange of a specified limit value, the measured values that violate the limit value can be brought forward in color.
Identification of Time Jumps and Time Overlaps:
Time jumps or time overlaps that occur because of a switch from daylight-savings time to standard time, for example, can be marked in color.
Identification of Replacement Values:
The replacement values used instead of a measured value can be brought forward in color.
Buttons of the Toolbars:
In the "Toolbar" dialog, you can configure the buttons of application windows for displaying trends and tables at runtime.
New Buttons:
"Activate Original View"
"Open the Dialog Window for Selecting Archive and Tag"
"Open the Dialog Box for Selecting a Trend"
"Open the Dialog Box for Selecting a Column"
"Select Time Range"
"Previous Trend in the Front"
"Next Trend in the Front"
"Insert"
"Delete Data Records"
"Previous Column in the Front"
"Next Column in the Front"
Configuration Data:
Persistent in RT (Run Time)
Persistent in RT and CS (Configuration System)
Keyboard Shortcuts:
The buttons of the toolbars of application windows for displaying trends and tables at runtime can be linked with freely parameter-assignable keyboard shortcuts.
dBase:
The data in short-term archives can now also be stored in the dBase III format (an advantage with large amounts of data).

Report Designer

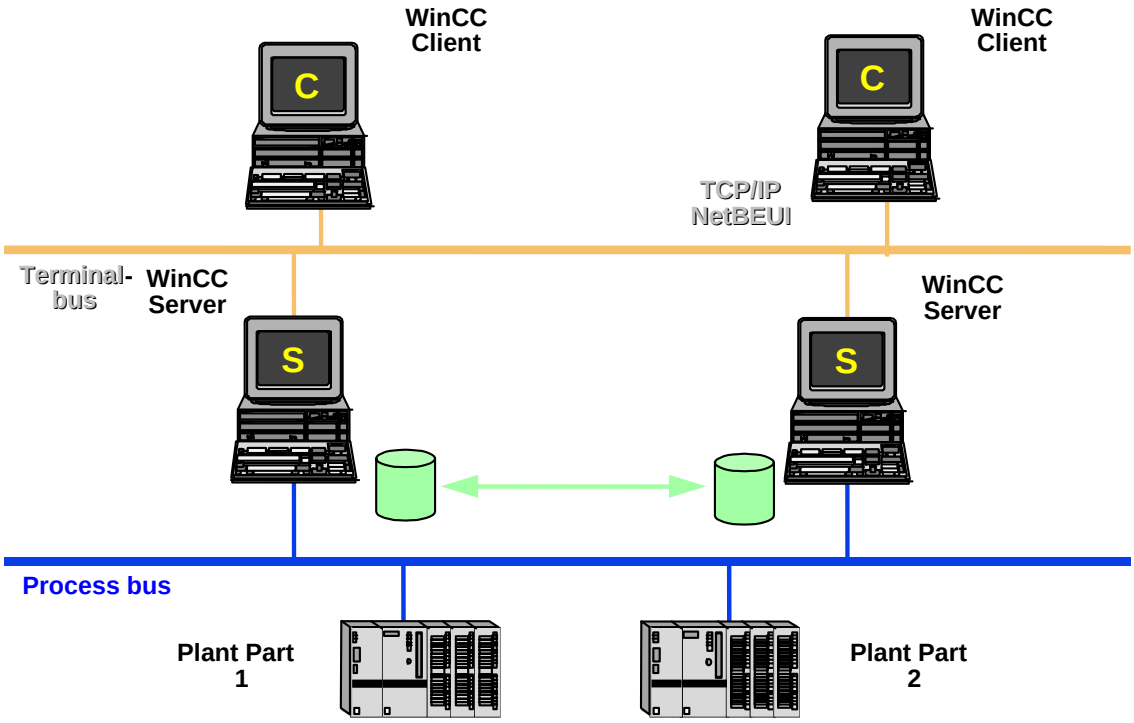
New Functionalities:
- The units of measure inch, centimeter and point can be set up for co-ordinates, text height and line wide.
- The "compact" format is available for the Project Documentation and Info of tags, whereby every tag is documented in one line.

User Archives

New Standard Function:
TlgDeleteRecord
New Standard Function:
TlgDisconnect

Redundancy

Two Servers running parallel in the same project guarantee the operation‘s maintenance when one of the Servers fails.



Storage

New Functionalities:

- Activating/Deactivating the automatic archiving.
- Lock-out data carrier
- Reconstruct logbook of data carrier
- Adjustable number of logbook entries
- Display of the archive server (ready or offline), after the Client was connected

Picture Tree Manager

New Functionalities:

- With every save, the hierarchy of the group displays is recalculated as an option. WinCC can, at the start of the runtime system, determine the topicality of the hierarchy of the group displays.

Alarm Logging Wizard

New Functionality:

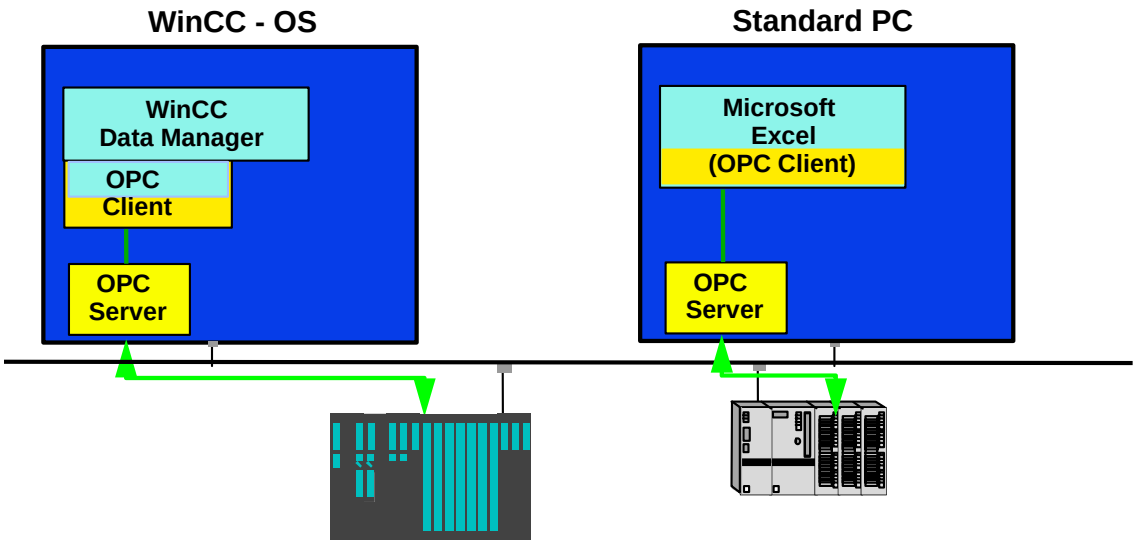
The user settings are saved. When the Wizard is later called, the previously saved settings are accepted (transferred).

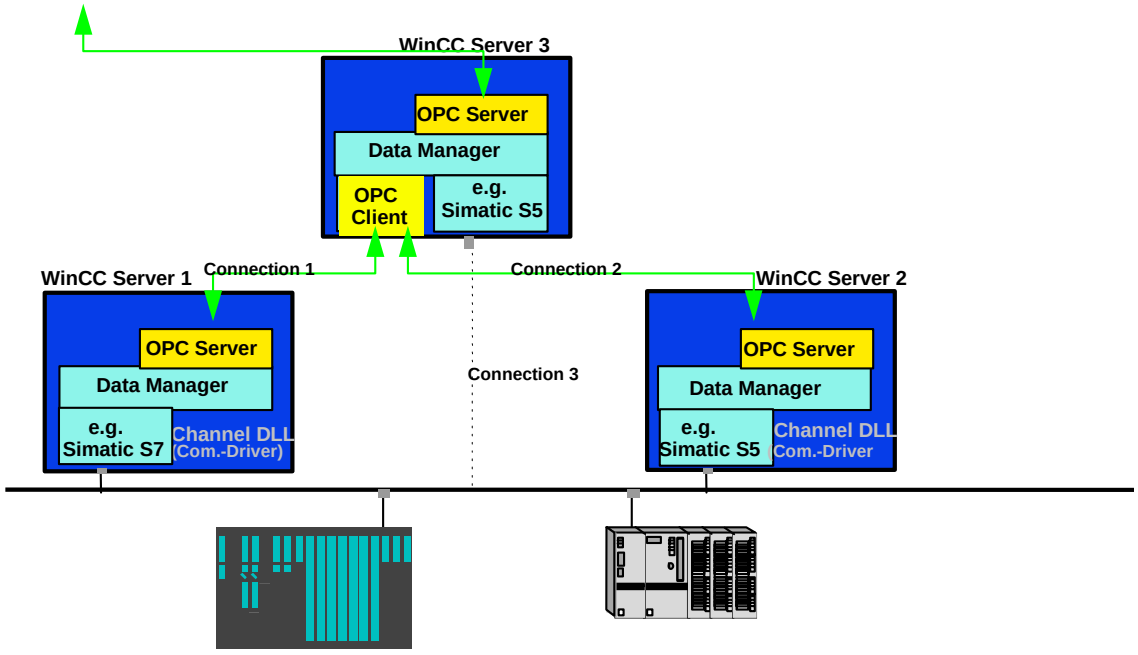
OPC (OLE for Process Control)

OPC is a new industry standard, that is being worked on and further developed in a Foundation. Members of this Foundation are, among others:
Microsoft, Fisher-Rosemount, Rockwell Software, Siemens Automation
WinCC becomes OPC Server, provides other applications with its data, for example, another WinCC System

WinCC maintains one OPC Channel DLL, for connection to other OPC Servers, for example, the OPC Manager of A&D PT2 (among others Profibus DB)

WinCC and Standard PC communicate with the process via OPC





cascadable Client/Server configuration

Changes in V4.0

Vis-à-vis earlier versions of WinCC, there are the following important changes:

GetLocalTime
The WinCC standard function "GetLocalTime" was renamed as „GetLocalTimeString", since a same-name function exists in Windows. Users of the standard function must either adapt their programs or rename the function.

S5-TRSP-Channel
Up to 30 WinCC connections per Channel-Unit can now be operated.

TagLogging
The trend display was heavily reworked and expanded.

User Object
User objects can be processed.
OLE-Controls can be integrated.

Serial Connection to S5
The Pseudowrite task is enabled.
The AS511 connector is now also enabled for Windows NT4.0.

Function Expansion WinCC V4.01

In additional to the function scope of V4.0, WinCC V4.01 contains the following expansions.

-Demo mode runs 1 hour.
-Access to process data using Internet/Intranet (Option),
Integration of Controls in HTML pages that display WinCC process data,
Remote display also using Internet/Intranet,
Economical long-distance (remote) diagnostics using Internet.
-WinCC Web Browser for direct surfing from WinCC,
modern technology at the work place,
Integration in process pictures,
Direct help in configuration through access to Customer Support,
Operator help / Work instruction as HTML page,
Data created once in a company can be displayed everywhere at the same time, from production manager to operator.

-Annotated e-mail system,
E-mails direct to/from operator,
Communicate system state by means of screenshots.

-Video Management System,
Optical identification of process changes,
by movement, for example, robot arm,
color change, for example, burner flame,
Saving, managing and reproducing video sequences.

-Integration in the new Microsoft world,
Microsoft Visual C++ Compilers V 5.0, the WinCC development tool,
Newest C++ - Compiler for application development available,
Increased performance through optimized code,
Low memory requirement,
Increased performance through Inproc technique: all configured actions in the Dynamic dialog run InProc, that is without task change between PDLRT and Script,
new technologies, for example, DCOM (Distributed COM),
Change-over of the data base interface to ODBC 3.0,
Compatibility with Microsoft Office 97.

Function Expansion WinCC V4.02

In addition to the function scope of V4.01, WinCC V4.02 contains the following expansion.

-System rounding out
-Basis for further WinCC options
-Stabilization Measures / Performance
-Expansion through further Diagnostic functionalities

-Default Trigger can be adjusted
Defaults of the updating trigger can be assigned parameters

-Dynamic Wizards for picture modules
Simple interconnection of picture modules (Picture-in-Picture Technic) (Use of picture windows, to integrate a template repeatedly in a ”main picture” and to let it work with its own instance data).

-Line intensity of dotted lines can be adjusted (only under NT)

-PLC / OS Software Redundancy
automatic switch-over of connection with the failure of a software redundant SIMATIC S7-300 / S7-400 (all types of connections)

-WinCC WebBrowser – Control
Integration of an Internet Browser in the user interface.
Password protection.
Configurable access to buttons for HTML pages.
Uniform display of information within a company.
Operates with Internet Explorer 4.

-Expansion through further Diagnostic functionalities
central authorization information with direct access from the Demo-Mode dialog.
Information about functionalities used and missing keys.
Information display via WinCC Scope.
Connection diagnostics for S5 Industrial Ethernet channels, AS511 and Serial 3964R (as with S7 channel) with entry in the logbook.
Performance monitor for early identification of bottlenecks in the system (hardware or software).
Performance monitor Monitors cyclically and when threshold values are exceeded.
Script Analysis, for example, for identifying endless loops.

-Measured value archiving
Optimized backup concept for dBase archives, limited system load during saving.
Reduction of load through the use of user-defined cycles in the WinCC Explorer for measured value archiving.
Optimized S7 connection, less telegrams through blocking.

- User Archive Option
Newly developed, completely innovative, user archive.
Ease of use with reference to configuration and operation.
Connections to the PLC is simply possible.
Old and new user archive can be used in parallel.
License for the user archive can be used for both.
Transfer of user data from the old user archive possible.

Use of Microsoft Standards such as Form Editor and Standard Data Base interface.

Stand-alone User Archive Editor in the WinCC Explorer
Setting up of archives
Definition of archive values
Changing, expanding of existing archives
Export / Import archive values.
Communication via raw data tags and process tags.
Display via ActiveX Control.

-WinCC Option Redundancy Level2
Online adjustment of innovative user archives.
Online adjustment of local server messages (operator messages, Batch Flexible).
Client switch-over when confirmed.
Configurable Client switch-over via operator or tag.
Monitoring of process connections with
Output of control system messages
Adjustment of values after restoration of process connection.

Function Expansion WinCC V5.0

In addition to the function scope of V4., WinCC V5. contains the following expansion.

-MultiServer
greater performance profile (6 Servers with up to 16 Clients), simplified expansion of the system, better performance.

-Standard Clients and Multi Clients can be mixed
-Distribution is triggered by the configuration (Trendserver, Alarmserver, Visualization-server)

-Configuration of distributed systems
Transferring / Conversion of V4 projects and pictures.
Independent configuration of the Server.
Generation of the ServerData. The ServerPackage contains a list of the picture names, tags and archives of the Server that are required by a Multi-Client for its configuration.

-A MultiClient offers the common view of

process data from distributed Servers in one picture.
all measured value archives, for example, in one trend window.
all message archives.
Pictures / Picture windows of various Servers.

-Web Navigator Server

Communication with the WinCC Server.
Management of the access rights.

-Web Navigator Client, system visualization and operation using the Internet / Intranet

Limited operation and/or purely a View-Only function depending on the access rights.
Minimum installation incl. the WinCC ActiveX Controls even via network.
Limitation Level 1, the outputting of reports to the WebClient is not supported,
no UserArchive, no project and global functions, that is,

Project functions and global functions must exist locally, which is not possible with a WebClient. These are therefore not executed at the WebClient.
Event triggered scripts (e.g. mouse click) in pictures and on objects are supported.

-Direct usage of STEP 7 - symbols in WinCC

-STEP 7 block call from WinCC Runtime

-Jump to the STEP 7 hardware diagnosis

-Cross Reference List

Tabular listing of tags, pictures, functions ...
Selective lists.
Central Linking.
Jump from the Cross Reference List to the Place of Use of the tags.

-Display of all dynamic properties of a picture

-Picture Preview

-Display of picture dynamics as an overview or selected according to the method of making dynamic

-Tag connecting in the GraphicsDesigner using drag & drop

-Quick linking of tags in pictures

-New toolbars for tags and AxtiveX

-Multi-line button labeling

-Conversion of the application windows to ActiveX Controls

Direct display and configuration of message and trend windows after inserting the controls in the Graphics Designer.
Storage of sample controls in the library.

-New Functions of the Controls

Trend window "Online Trend" without its own archive.
Open definition of the after decimal positions in the Table Control.

-Display of DXF data (E-Plan, AutoCAD)

Output via an ActiveX Control.
Zoom function in the view.
Support by Wizard.
Central or local data management of the DXF files is possible.

-General channel expansion

Channel-specific Help files.
Channel-specific Log files in the diagnostic directory.
Expanded channel diagnosis for the following channels
SIMATIC S5 Ethernet Layer 4 / TF.
SIMATIC S5 AS511 / Serial 3964R.
SIMATIC S7 Protocol Suite

-Expansion in the SIMATIC S7 channel

Expansion of the number of units for

Profibus (CP5412A2).
Industrial Ethernet (CP1413).
Two CPs can be slotted for this type of communication.

Write tasks can be processed with priority.

Support of the H-CPU of S7-400 for Profibus and Industrial Ethernet.

-New Documentation -->Communication Manual

Communication at a glance.
Decision making help.
Numerous communication examples (for installation and configuration).
Example projects for STEP 5 / 7 and WinCC.
Notes on diagnostics.

- Expansion of the OPC capability by functions of the current specification V2.0.
  Support of Automation Interface and Customer Interface.
  Direct access via VBA (Excel, for example) to the Automation Interface.

Function Expansion WinCC V5.0+SP1

In addition to the function scope of V5.0, WinCC V5.0+SP1 contains the following expansion.

-Script Converter (Conversion of scripts for the display of tags from scripts in the Cross Reference List).
-Display of the Quality Code in the WinCC Explorer even for OPC tags.
-WinCC Channel "System Info" for evaluating system information.
-WinCC Channel "SIMATIC S5 Ethernet Layer 4" with support by TCP/IP.
–Picture module technic with Tag Prefix.

With WinCC V5.0 SP1, even the control system options Basic Process Control and Storage, as well as the OPC Server function and the DP Channel DLL are enabled. The functions of the Advanced Process Control option have been included in the Basic Process Control option (time-of-day synchronization, for example). WinCC V5.0 SP1 can be operated with Windows NT V4.0/SP3/SP4/SP5/SP6 and Internet Explorer V4.01 SP1 / V5.0.

Function Expansion WinCC V5.0+ SP2

In addition to the function scope of V5.0+SP1, WinCC V5.0+SP2 contains the following expansion.

-WinCC V5.0 SP2 is released for the Windows NT 4.0 SP5/SP6 and Windows 2000 SP1 operating systems. Also released is the operation with Microsoft Internet Explorer V4.01 SP1, V5.0 and higher.
-WinCC with Italian and Spanish operating systems.
-Mixed operation - Client / Multiclient.
-Pictures can be run on Client and Multi-Client systems.
-Project download in the SIMATIC Manager of WinCC Stations, redundant Server and Multi-Client Stations can be set up and project data can be downloaded to the WinCC PCs.
-Permanent operability, that is, distribution of operator consoles amongst redundant Server.
-Central message sequence report for multi-user system.
-Central lock-out / release of messages using SIMATIC S7-400.
-Siemens HMI Symbol Library as ActiveX Control.
-Search and Replace of partial strings in the CrossReference.
-Search and Replace in the Graphics Designer after selecting several picture objects.
-Component selection during update installation.
-Sybase Version 7 for increasing the archiving performance.
-Ladder Rung Jump with operating authority in STEP7.
-Accelerated behavior in project handling.

Function Expansion WinCC V5.1

In addition to the function scope of V5.0+SP2, WinCC V5.01 contains the following expansion.

- WinCC Version 5.1 now also supports Italian and Spanish in full.
The interface, system messages and documentation are available completely in Italian and/or Spanish.
-The WinCC configuration tool enables simple and high-performance configuration of large quantities of data.
In MS Excel, new projects can be created and existing projects can be edited. WinCC tags as well as data of Alarm Logging, Tag Logging and the Text Library can be exported from projects, edited and imported.
Single messages, limit value monitoring and archive tags can be configured directly via the tags using the WinCC configuration tool.
The large number of editing options provided by MS Excel can be expanded by experienced users using VBA programs (macros).
- Support of SIMATIC system diagnosis messages. If a system error occurs, detailed information on the error is displayed automatically, e.g. type or location of the error.
This new function can be used to reduce system downtimes.
- WinCC provides the possibility to issue multi-line message texts in a tool tip.
This means that the user can be provided with detailed information on the system error, e.g. location of the error and remedial measures.
- The redundancy switchover also takes place when the process link to a server is disrupted.
Until now, multi-clients in a redundant system were only switched over if one of the two servers failed.
- With the F(x) Control, WinCC offers another possibility to document and analyze values.

F(x) Control can be used simply to display variables as functions of another variable, e.g. temperature as a function of pressure, in a single diagram.
F(x) Control also provides the possibility to compare these with desired trends.
- The number of possible user groups in the User Administrator has been increased from 10 to 128 user groups.
- WinCC Version 5.1 provides the possibility to change texts in pictures using Find and Replace. The function can be used for all or only for some of the objects in a picture.
- The channel "AS511 under Windows 2000" is also released for use under Windows 2000. The table headers are also copied when user archives are exported. Until now, only the data without column names was exported.
-Option Advanced User Administrator. Fulfil the FDA (Food and Drug Association) requirement 21 CFR Part 11 in the Pharma and Food Industrie.
-Touchkeyboard also with Login.
-Release Windows 2000 +SP2/SP3, IE 6.
-New OPC-Server V2.01 (better Performance).

Function Expansion WinCC V6.0

In addition to the function scope of WinCC V5.1, WinCC V6 contains the following expansion.

-Control and Monitoring under Microsoft Windows 2000 (Single-user systems, Clients, Server)
-Control and Monitoring under Microsoft Windows XP (Single-user systems and Clients)
-Historian based on Microsoft SQL Server 2000 with high data compression,
Swapping function and backup mechanism.
-12 WinCC servers and/or redundant server pairs.
-32 WinCC clients per server.
-Visual Basic for Applications (automation of the configuration).
-Visual Basic Scripting (Runtime scripts).
-Web Navigator Server on WinCC Clients as data concentrator for more security.
-Accepting configuration data for the Web without changes.
-OPC HDA, OPC A&E, OLE-DB.
-Functional adaptation of the Web Navigator to a WinCC-Client.
-New option: WinCC/Dat@Monitor WebEdition (Historian Tools).
-New option: WinCC/Connectivity Pack (OPC HDA, OPC A&E, Database access using
WinCC OLE-DB Provider).
-New option: WinCC/Industrial DataBridge (exchange of WinCC data using
standard interfaces).
-New option: WinCC/ SIMATIC Logon, WinCC/ Audit and WinCC/ Electronic Signature
(support in the implementation of FDA CFR21 Part 11 Compliant Solutions).
-Open database interfaces (ADO, OLE DB).
-The WinCC basic license (RT, RT/CS) contains 512 available archive tags and as many
messages as you like.
-The number of archive tags can be expanded using powerpacks.
-The license package palette was supplemented by the 8K Power Tags.
-By using dynamic report parameters, you can, for example, set the time range of a trend report
online.
-The scrope of the project documentation can, for example, be limited to certain objects
and thus be reduced.
-Database data can be integrated in a WinCC report as an individual value or table using an
ODBCProviderobject.
-Outside data in a CSV format can be integrated in a WinCC report as a table or a trend.
-You can develop your own report system provider for displaying data as a table or
graphic. The WinCC option "ODK" is required for this.
-The current screen view or only a screen segment can be integrated
directly into a WinCC report and can be printed out with the associated data.
-The current, online configured view of the WinCC Online Table Controls and
the WinCC Trend Controls can be printed out directly.
-Offline engineering and loading of project modifications in an active WinCC
Runtime environment.
-Zooming: Process pictures can be enlarged using the mouse wheel or by ‘pulling open‘ a
segment.
-Panning: Displayed segments of process pictures can be moved with the mouse.
-Decluttering: Picture objects can – depending on the layer or the
object size – be toggled invisible and then visible once more.
-Connector: Objects connected with the new line element can be shifted
within a process picture, whereby the connection remains intact.

Function Expansion WinCC V6.0

In addition to the function scope of WinCC V5.1, WinCC V6 contains the following expansion.

-Alias: Archive tags can be given an alias that simplifies handling in runtime.
-Screen keyboard: The screen keyboard can be automatically shown dependent on the selected objects.
-New message blocks: The message blocks "Current user" and "Priority".
-Redundancy toggling: Important WinCC applications are monitored using the cyclic life-sign monitoring.
-System monitoring in Runtime: Critical system states, such as llittle memory capacity on the hard drive, are automatically monitored and signaled.
-Option WinCC/Web Navigator V6.0.
-With the expanded printer selection dialog, you can, for example, switch between black/white and color printers.
-Ascending and descending, multi-level sorting of displays in the WinCC Message Control in Runtime (similar to Excel).
-Line object connector with connection with connection points (‘elastic band‘)
-Changeable mouse pointer.
-Expansion of the Graphic Object for displaying pictures in GIF format and JPG format.
-32 Layers per picture in RT can also be made dynamic.
-UTC (universal time coordinated) time format based on GMT.
-Migration of V5.1 projects.
-User Administrator: automatic logout after an inactive time period.
-VBA Programming for Engineering (Visual Basic for Application).
-VBS for RT (Visual Basic Scripting).

SIMATIC HMI **Operating and Monitoring** **ST-B WINCCS** Course Folder Version: A 6.0	**SIMATIC HMI** **Operating and Monitoring** **ST-B WINCCS** Course Folder Version: A 6.0	**SIMATIC HMI** **Operating and Monitoring** **ST-B WINCCS** Course Folder Version: 6.0	**SIMATIC HMI** **Operating and Monitoring** **ST-B WINCCS** Course Folder Version: A 6.0	**SIMATIC HMI** **Operating and Monitoring** **ST-B WINCCS** Course Folder Version: 6.0
SIMATIC HMI **Operating and Monitoring** **ST-B WINCCS** Course Folder Version: A 6.0	**SIMATIC HMI** **Operating and Monitoring** **ST-B WINCCS** Course Folder Version: 6.0	**SIMATIC HMI** **Operating and Monitoring** **ST-B WINCCS** Course Folder Version: 6.0	**SIMATIC HMI** **Operating and Monitoring** **ST-B WINCCS** Course Folder Version: 6.0	**SIMATIC HMI** **Operating and Monitoring** **ST-B WINCCS** Course Folder Version: 6.0
SIMATIC HMI **Operating and Monitoring** **ST-B WINCCS** Course Folder Version: 6.0	**SIMATIC HMI** **Operating and Monitoring** **ST-B WINCCS** Course Folder Version: 6.0	**SIMATIC HMI** **Operating and Monitoring** **ST-B WINCCS** Course Folder Version: 6.0	**SIMATIC HMI** **Operating and Monitoring** **ST-B WINCCS** Course Folder Version: 6.0	**SIMATIC HMI** **Operating and Monitoring** **ST-B WINCCS** Course Folder Version: 6.0