SIEMENS

SIMATIC HMI

WinCC V12
Runtime Professional

Getting Started

05/2013
A5E32311980-AA

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

⚠ DANGER

indicates that death or severe personal injury **will** result if proper precautions are not taken.

⚠ WARNING

indicates that death or severe personal injury **may** result if proper precautions are not taken.

⚠ CAUTION

indicates that minor personal injury can result if proper precautions are not taken.

NOTICE

indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

⚠ WARNING

Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Siemens AG
Industry Sector
Postfach 48 48
90026 NÜRNBERG
GERMANY

A5E32311980-AA
℗ 05/2013 Technical data subject to change

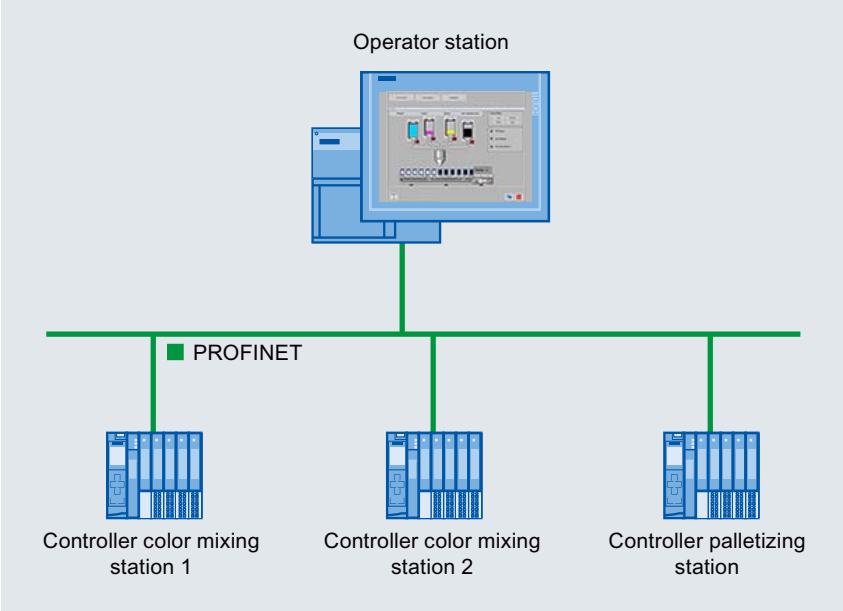
Table of contents

Welcome

1

1.1 Welcome

Welcome to "Getting Started WinCC V12 Runtime Professional". Based on the example of a color mixing plant, we are going to show you how to use WinCC V12 Professional Edition to complete an existing configuration and thus obtain a complete, executable project.



You will also learn the following configuration steps in this Getting Started:

- Creating faceplates
- Configuration of a screen navigation with screen windows
- Creating recipes
- Logging alarms and process values
- Configuring reports
- Creating user-defined functions
- Configuring a user administration
- Configuring a multilingual user interface
- Testing and loading the project on the HMI device

Sample projects

Two sample projects are prepared for this Getting Started:

- "HMI_Color_Mixing"

  If you want to carry out the configuration steps shown in this Getting Started yourself, use the "HMI_Color_Mixing" project. Some areas are preconfigured in this project. In particular, the controller and the connection with the controller are already available.

  By completing all the steps of this Getting Started, you will obtain a project that can be run on the HMI device.

- "HMI_Color_Mixing_Complete"

  If you simply want to view the configuration steps and results shown in this Getting Started, use the "HMI_Color_Mixing_Complete" project. This project contains the complete, ready-to-run project in which all the configuration steps described in the Getting Started have already been completed.

Download

You can find example projects here:

RT Professional Example Project
(http://support.automation.siemens.com/WW/view/en/72927729)

1.2 Classification

This Getting Started is intended for advanced users proficient in WinCC V11/V12 and in the configuration of operator control and monitoring solutions. General concepts such as tags and screens are not discussed in this Getting Started.

Position in the information landscape

This Getting Started explains the configuration of the visualization. Basic information on topics such as the configuration of the controller, as well as the integration of the visualization and controller, are discussed in other Getting Starteds.

- In "Getting Started WinCC V12 Runtime Professional - Advanced Configuration Options", you will be shown additional configuration options, in particular for multi-user systems and for communication and remote control.

  Link to download (http://support.automation.siemens.com/WW/view/en/72928407)

If you are also interested in PLC programming and/or diagnostics along with the "Operator control and monitoring" topic, you can use the following Getting Started documents to familiarize yourself with the color mixing plant from the control viewpoint:

- Getting Started STEP 7 Professional / WinCC Advanced V11 for sample project Filling Station

  Link to download (http://support.automation.siemens.com/WW/view/en/54430386)

- Getting Started SIMATIC S7-1500 and TIA Portal V12

  Link to homepage (http://support.automation.siemens.com/WW/view/en/71704272)

1.3 The sample project "HMI_Color_Mixing"

Introduction

If you want to perform the steps shown in this Getting Started yourself, use the sample project "HMI_Color_Mixing". The following configurations are already contained in this sample project:

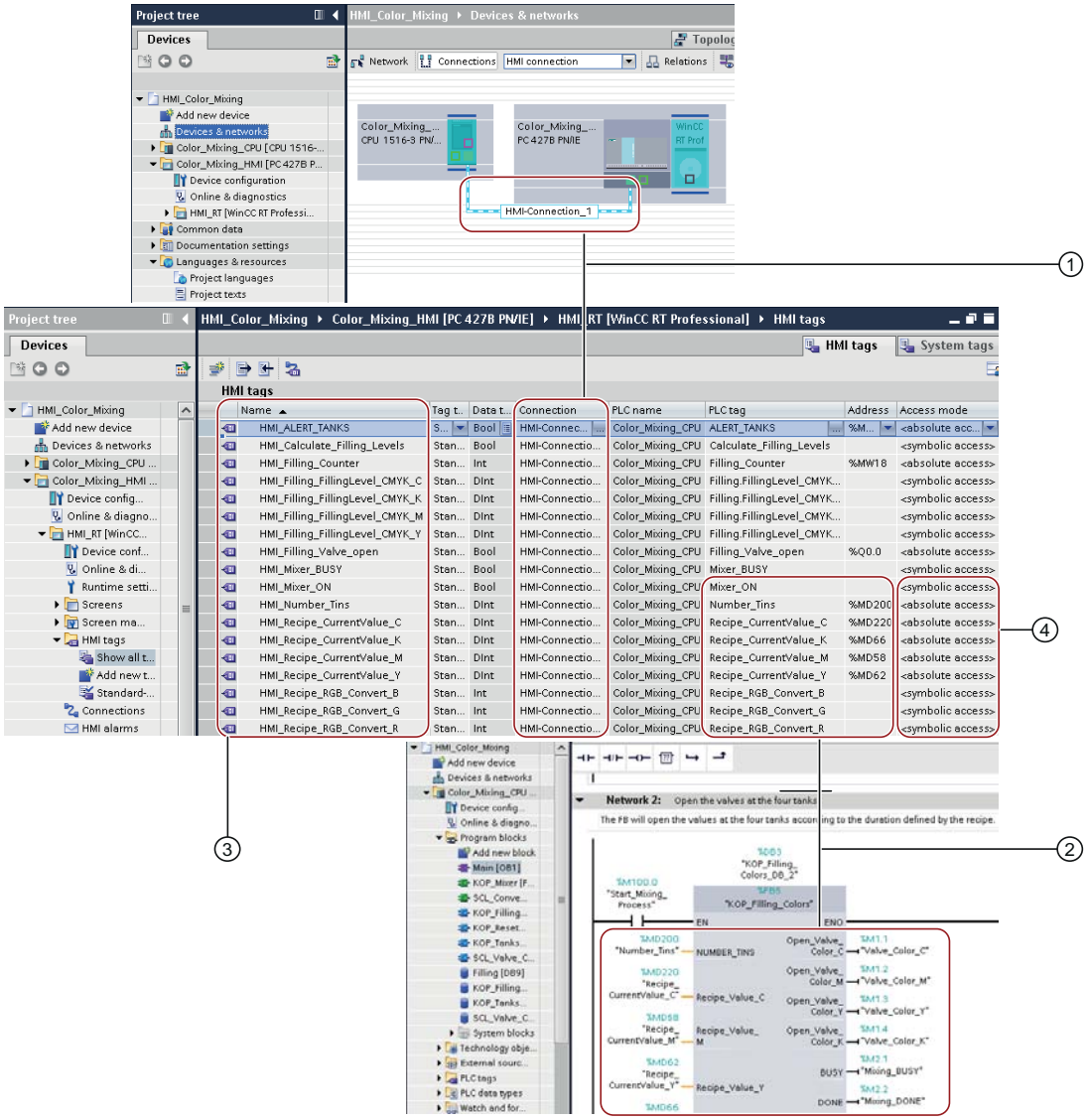
- Communication between the controller and the HMI device
- Screens and graphics
- Scripts

The object names are configured in English.

Communication between the controller and the HMI device

Communication between the HMI device and the controller is already fully configured in the sample project:

- HMI device and controller are connected to each other in the "Devices & Networks" editor
- The tags required for the communication are already created in the HMI device and in the controller



① The HMI device and the controller are networked and connected with each other:

- "Networking" defines the physical connection, such as the IP addresses of the associated interfaces.
- "Connection" defines the logical assignment of the two communication partners to the data exchange, for example, recipes.

② Tags for controlling the color mixing plant have been created in the controller. These tags are used in the control program to, for example, control the filling process.

③ For each PLC tag, an HMI tag is created on the HMI device. Each HMI tag has a unique name. You can use HMI tags for tasks such as the following on the HMI device:

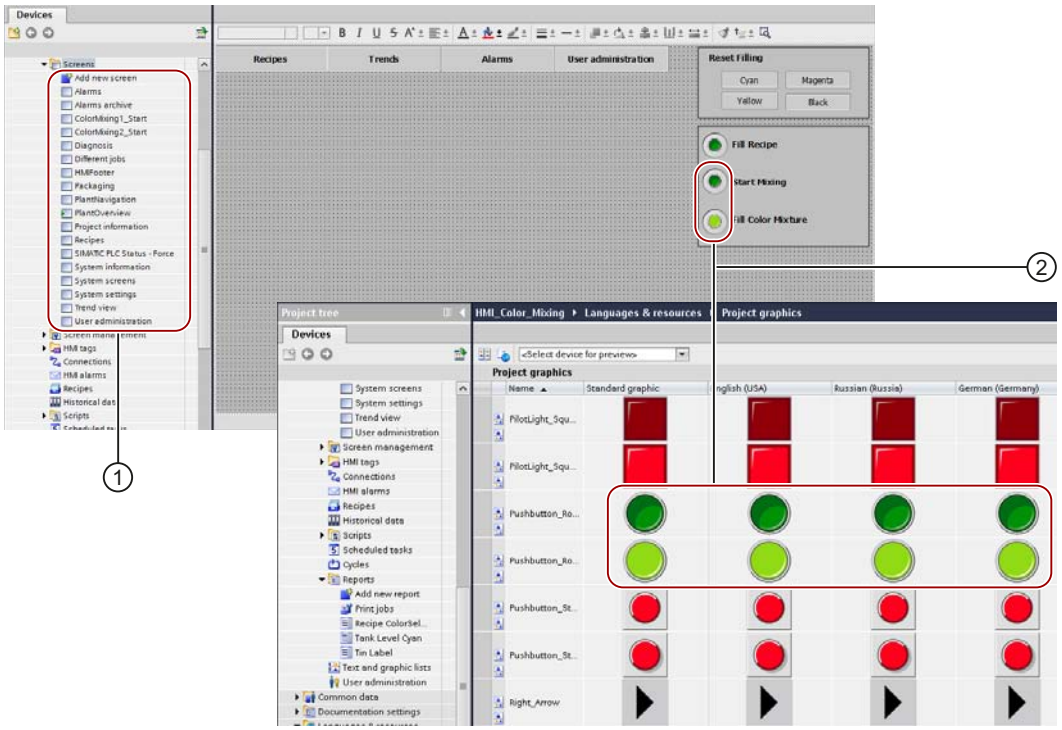
- Visualizing fill levels
- Specifying mixing ratios

④ For each PLC tag, an HMI tag is created on the HMI device. Each HMI tag has a unique name. You can use HMI tags for tasks such as the following on the HMI device:

- Visualizing fill levels
- Assign the mixing ratios. The "access type" defines whether the tags on the HMI device ③ address the memory area of the controller in symbolic or absolute terms. Advantage of symbolic addressing: The HMI tag is updated automatically when the addressing changes in the controller.

Screens and graphics

Screens and graphics are already configured in the sample project:



① The sample project includes all screens that contain operating objects for communication with the controller. Screen changes are also configured between the screens. In this Getting Started, you will mainly use the following screens:

- "ColorMixing1_Start": In this screen, you configure the visualization of the color mixing plant.
- "Recipes": In this screen, you configure the settings for the color components to be filled.
- "Plant_Overview": In this screen, you configure the screen window for navigation.

In the other screens you can, for example, view the diagnostic alarms.

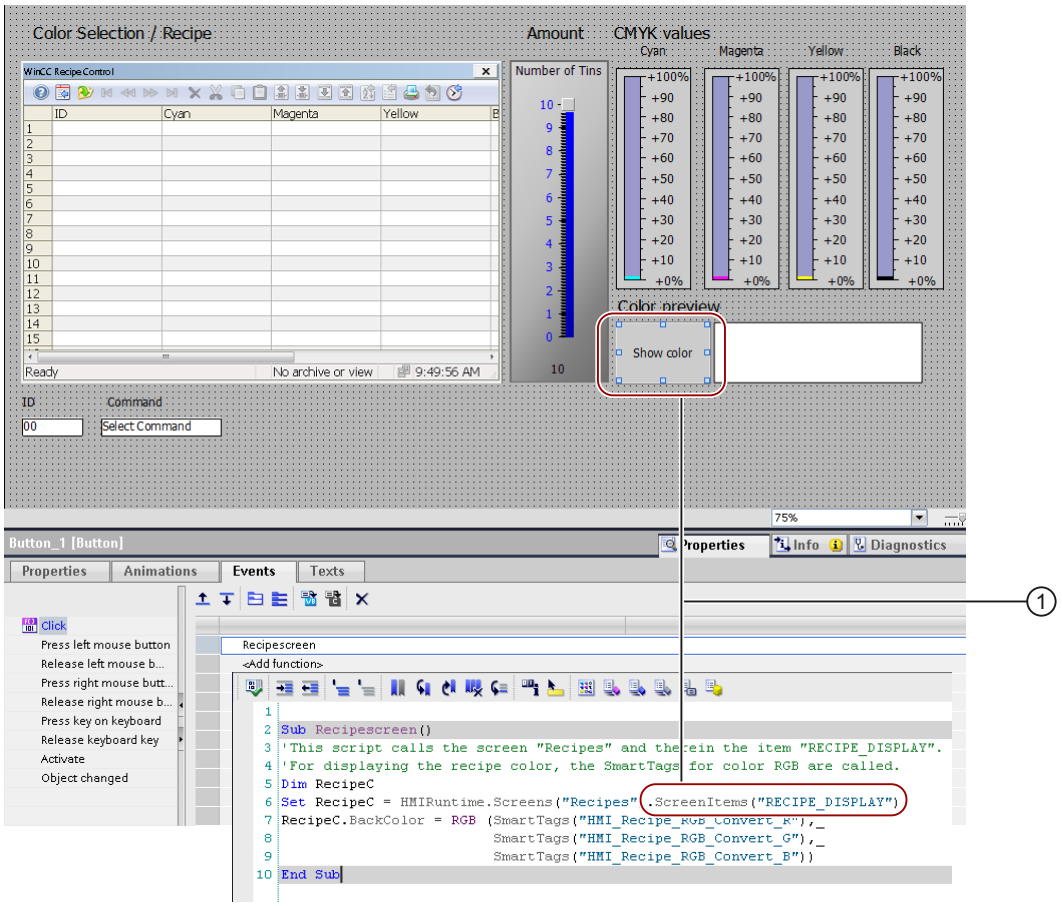
② If, for example, you insert a new graphic from the graphics library into the project, this graphic is automatically added to the "Graphics".

The "Graphics" already contain all graphics that are used in this Getting Started, such as icons for displaying a valve status. Alternatively to the steps shown in the Getting Started, you can also drag the graphics from the graphics collection into a screen.

Scripts

In the sample project, the "Scripts" editor contains two scripts that control how the HMI device displays the color to be filled:

- Startscreen: Controls the color display on the color tins
- Recipescreen: Controls the color display in the "Recipes" screen:



① The script references the rectangle and converts the color values in the CMYK format into an RGB value. The RGB value is assigned to the rectangle as background color.

1.4 HMI device and monitor

To operate the color mixing system, the SIMATIC Box PC IPC627C HMI device is used together with a 15" (1280*1024) SIMATIC Industrial Flat Panel IFP1500.

The Box PC SIMATIC IPC627C is compact and high-performance. It is suitable for challenging measurement, open-loop and closed-loop control tasks in ambient temperatures of up to 55 °C and can be used in shipbuilding. The two slots make it scalable and extendable.

HMI device SIMATIC Box PC IPC627C

The SIMATIC Box PC IPC627C is characterized by the following:

- Highest performance and ultra fast system responses:
  - Intel Core processors (i7, i3)
  - Powerful onboard HD graphics integrated in the CPU
  - DRR3 memory technology
- Very high system availability and data security through RAID controller onboard, Solid State Drive (SSD) and ECC RAM
- Optional PROFIBUS or PROFINET interface with three ports for the connection of distributed field devices or for interfaces to SIMATIC S7
- Flexible control cabinet installation cabinet with high user friendliness
- Very robust in operation (protection against vibration, shock)

For more information, refer to the operating instructions of the SIMATIC Box PC IPC627C HMI device.

The SIMATIC Industrial Flat Panel IFP1500 15“ monitor

The SIMATIC Industrial Flat Panel IFP1500 is an LCD monitor for connection to industrial PCs.

The SIMATIC IFP industrial flat panels are ideally suited to use as industrial monitors with fast reaction times for delay-free image refresh, for example during jog mode or trend display. The devices are available with 15", 19" and 22" widescreen fronts, as display device only or display device with touch operation.

A 15" touch version with function keys, front USB interface and a keypad is also available.

The main features of the SIMATIC Industrial Flat Panels are:

- High-resolution widescreen displays for a well-structured layout of the display and control interface
- Very good readability from all directions
- Front panel design matching SIMATIC Panel PC and Comfort Panels

For more information, refer to the operating instructions of the SIMATIC Industrial Flat Panel IFP1500 15“ monitor.

1.5 The SIMATIC S7-1500 controller

The CPU 1511-1 PN controller

The S7-1500 automation system is used to control the color mixing station. The SIMATIC S7-1500 automation system represents the further development of the SIMATIC S7-300 and SIMATIC S7-400 controller systems.

Thanks to the integration of numerous new performance features, the innovative controllers of the S7-1500 product range provide excellent operating capabilities and maximum performance to users:

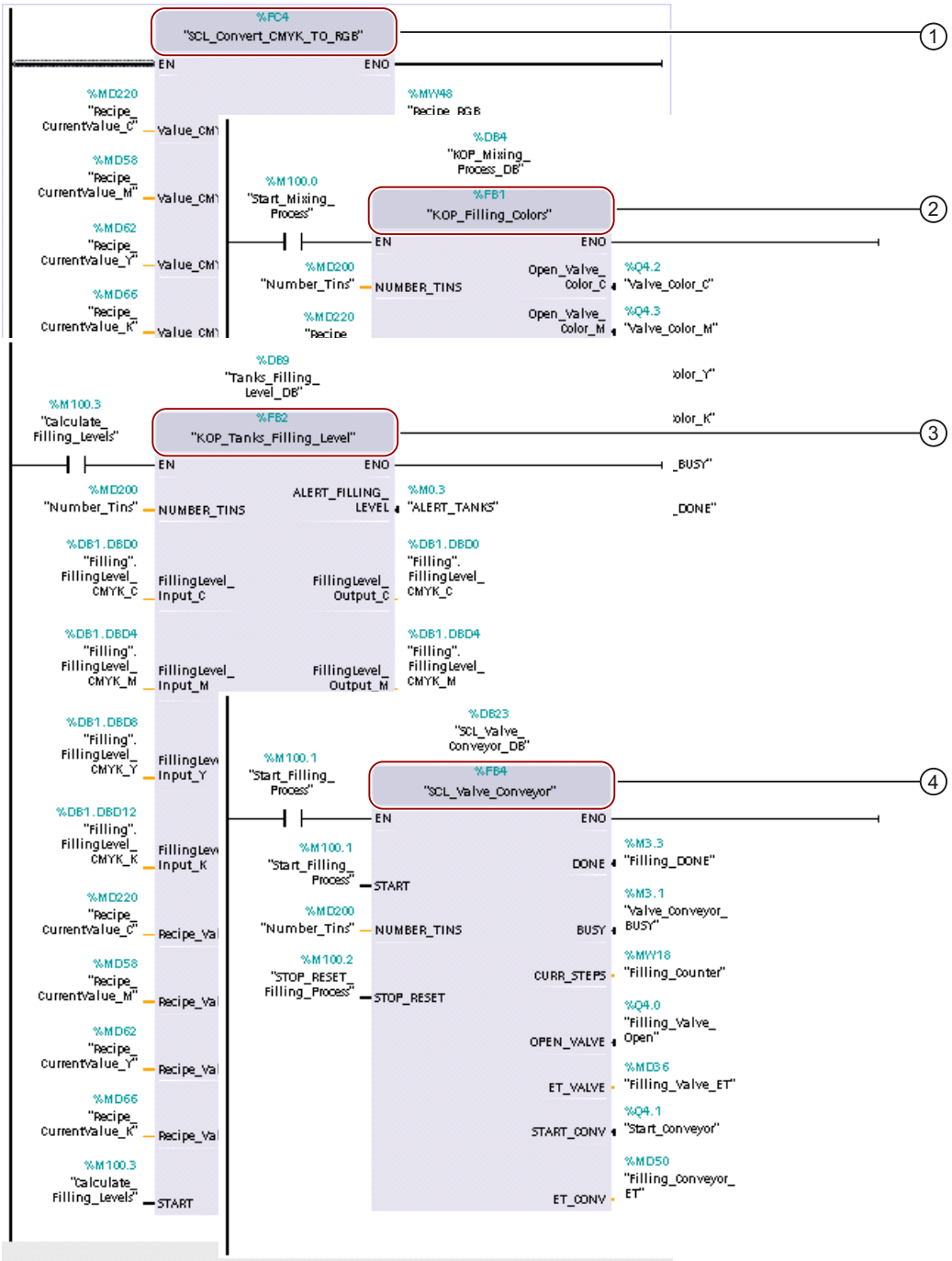
- High-performance command processing and communication
- Integrated Motion Control functions via PLCopen blocks
- Integrated Security - know-how protection, access protection, and copy protection
- Fast commissioning thanks to the simple configuration of the hardware, integrated display, and trace function
- System-supported diagnostics - no configuration required; even when PLC is in Stop mode

The CPU 1511-1 PN is introduced and configured in a separate Getting Started. For more information on this topic, refer to the following documents:

- Getting Started S7-1500
- "S7-1500 Automation System" system manual
- CPU 1511-1 PN Manual

Important elements of the control program

The control program for the color mixing station is already integrated in the controller. We primarily use the following program elements:



① Conversion of CMYK values for the recipe and calculating an approximate value for the RGB color space

② Conversion of mixing ratio portions into equivalent valve opening times at the reservoirs

③ Recalculation of reservoir fill levels

④ Opening the filling valve

These program parts can be found in the project tree under "Color_Mixing_CPU > Program blocks" in the "Main [OB1]" program block.

Configuring the HMI screen for the color mixing system

2

2.1 Introduction

Basics

In runtime, you operate and monitor machines and plants with the help of the screens loaded on the respective HMI device.

In WinCC, you manage the screens in the project navigation under "Screens". Predefined objects help you to map your plant, visualize process sequences, or display and specify process values.

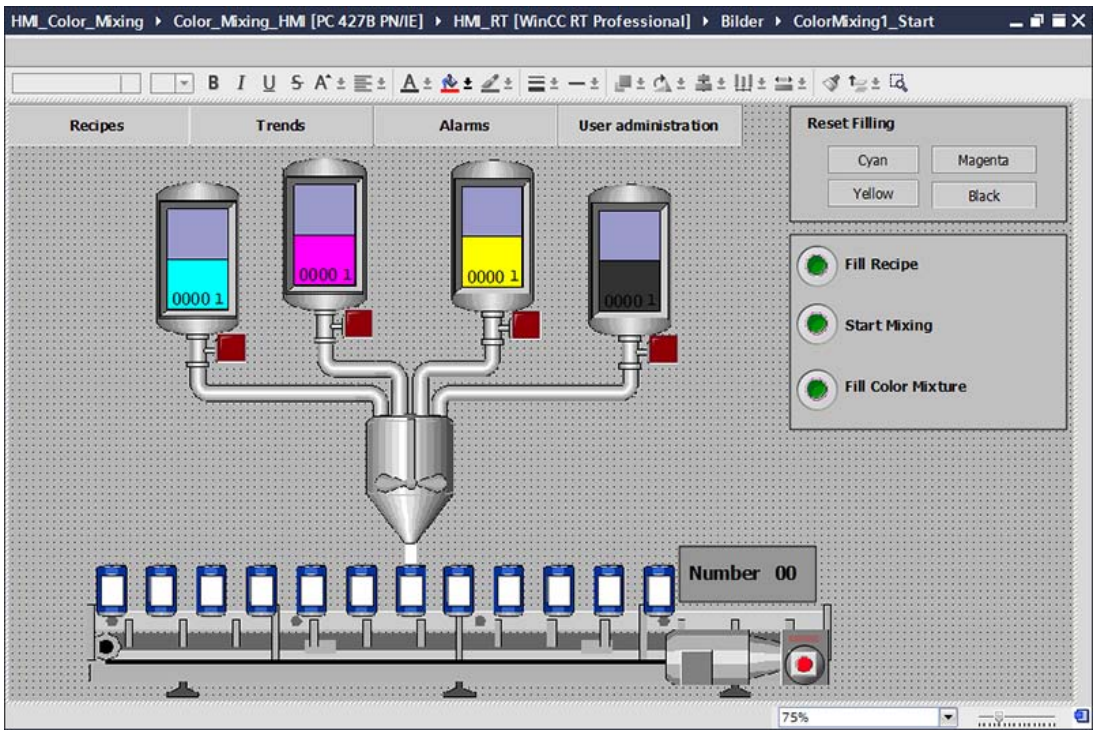
Tasks

We want to visualize the color mixing station, as well as the most important status information and parameters, on the start screen of the HMI device.

The color mixing station comprises the following elements:

- One color reservoir with fill level display for each printing color
- Mixer
- Supply pipes to the mixer
- Conveyor with emergency-stop switch

We also want to show the "Mix color" and "Fill color" process steps as an animation with dynamic visualization objects.



In this chapter, you will learn the following:

- How to visualize the elements of your plant in a screen with the help of predefined graphics.
- How to use icons to visualize status information, e.g. the valve states at the reservoirs.
- How to use faceplates to configure objects of the same type, e.g. the color reservoirs.
- How to dynamize objects, for example, to animate process steps.

Notes

The start screen of the sample project contains additional operating objects whose configuration is not covered in this Getting Started. Information on the basics of configuration, in particular on screens, is available in "Getting Started STEP 7 Professional / WinCC Advanced V11 for sample project Filling Station".

These controls include:

- Screen change buttons
- Fill level reset buttons
- Switches for operating and monitoring the plant: Fill recipe, start mixing, fill color mixture

2.2 Using faceplates for color reservoirs

2.2.1 About faceplates

The current fill level in the color reservoirs is to be visualized in the start screen of the color mixing station: graphically using a color bar, and numerically with quantity specified in liters. An additional icon should indicate the open and closed states of the color reservoir valve.

Since you need a color reservoir for each color, it is expedient to visualize these color reservoirs in a faceplate.

What are faceplates?

Faceplates are objects that you create from basic display and operating objects, and for which you define configurable properties.

Faceplates reduce configuration requirements due to the following advantages:

- Reusability

  You can reuse faceplates. Faceplates are therefore particularly suitable for frequently used automation components and for a fast implementation of plant extensions.

- Can be modified centrally

  Changes to a faceplate type are transferred automatically to all of its instances - apart from the properties which are set individually at the instance.

Working with faceplates

You first create a **faceplate type** from certain display and operating objects and specify the properties that can be individually set when it is used. These may include properties such as the foreground color or the tag linking of an object. This faceplate type is stored in the project library.

To use it, copy the desired faceplate type from the project library to your screen. You can then configure the properties that can be individually set in this **instance** to suit your individual requirements.

Steps to complete

1. Visualization of the color reservoir, including a fill level display
2. Visualization of the valve state

3. Creating a faceplate type for reservoirs
4. Insertion of the reservoir instances in the screen and customization

2.2.2 Visualization of color reservoirs

Introduction

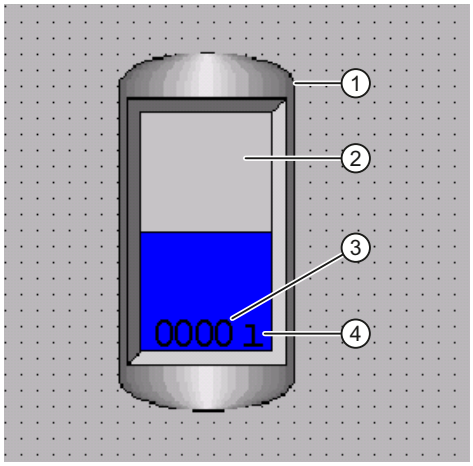
You use various objects to configure a reservoir with fill level display once. An icon in the screen visualizes whether the valve of the reservoir is open or closed. You can then use this reservoir later for the cyan, magenta, yellow and black printing colors.

Requirement

- The "ColorMixing1_Start" screen is open in the editor for editing.

Procedure

1. Create a color reservoir from the following objects:



① Tank with inspection window from "WinCC graphics folder > Automation equipment > Tanks > 256 colors"

② Bar graph to graphically display the fill level
Maximum value: 1000
Minimum value: 0
Start value: 0

③ I/O field for displaying the fill level in numerical format

④ Text box for the unit in which the fill level is being measured.

2. Insert a valve and a graphic I/O box to display the valve status (1 + 2).

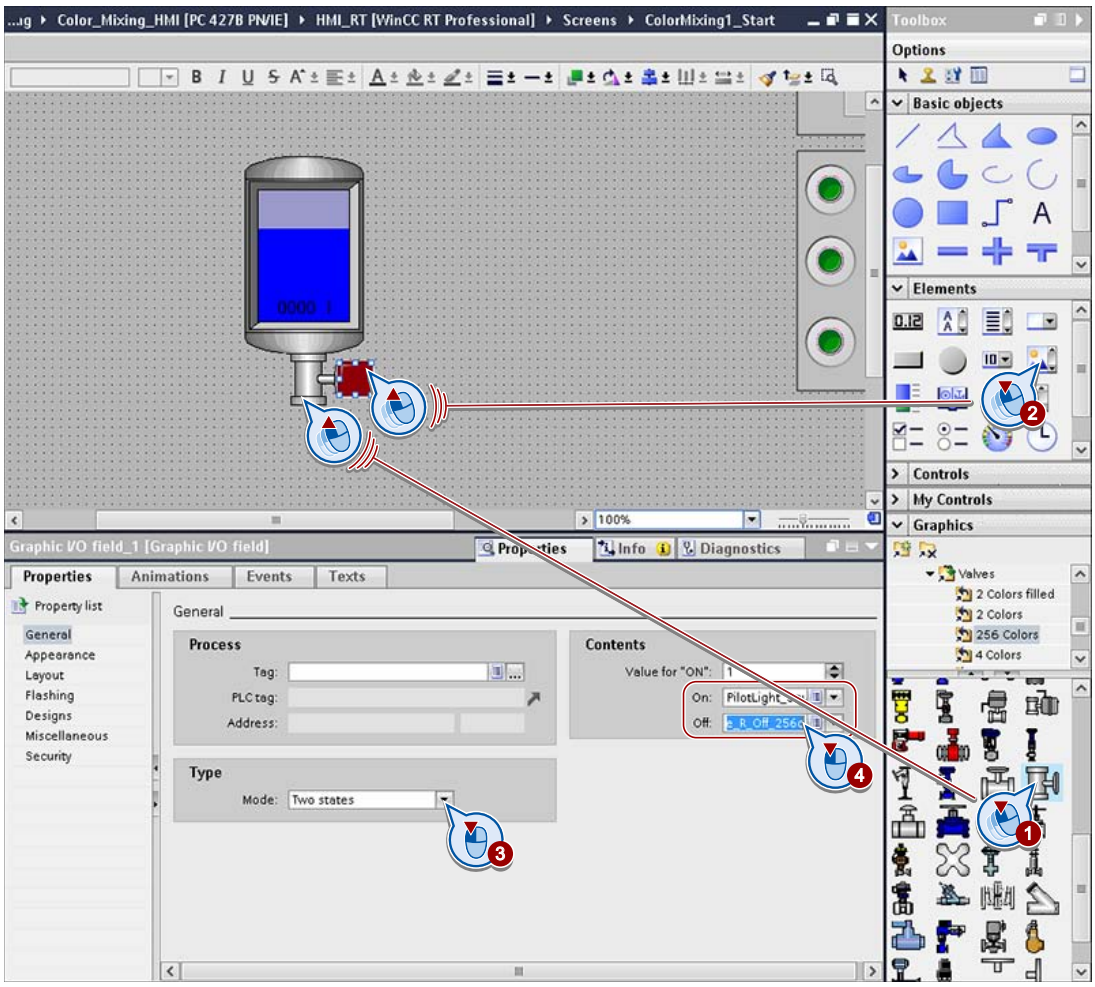
3. Configure the graphic I/O box in the inspector window (3 + 4):

   The graphics contain two icons that are suitable for displaying the valve status:

Meaning	Name
Valve open	PilotLight_Square_R_On_256c
Valve closed	PilotLight_Square_R_Off_256c

   – Select the "Two states" mode because the valve will be in either open or closed state.

   – Select the icons that display the valve status.



Note

Assign names to all objects to facilitate their identification in the future.

- Bar: "fill level bar"
- I/O field: "fill level num"
- Symbolic I/O field: "valve status symbol"

Result

A reservoir with display of the fill level and the valve status has been configured.

Since the visualization data is saved to a faceplate type, you should not configure the individual properties or link the tags until you initially use the faceplate.

2.2.3 Creating a faceplate type

Introduction

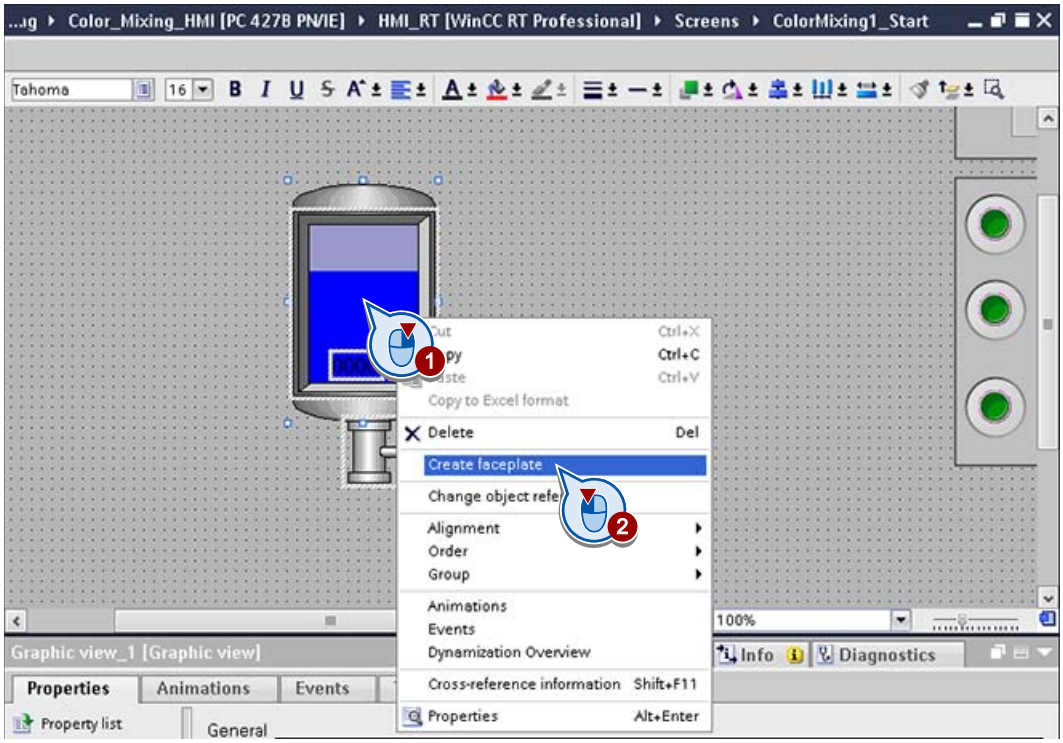
The configured color reservoir is to be used as faceplate type for the cyan, magenta, yellow and black printing colors.

It must be possible to configure the following features individually for each color tank:

- The foreground color of the bar graph
- The tag connection of the graphic and numerical fill level views
- The tag connection of the valve status

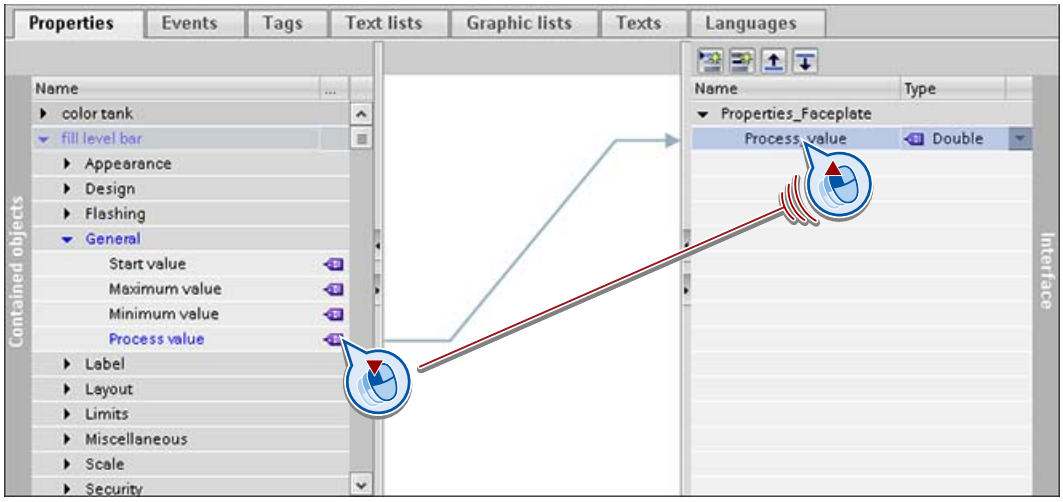
Creating a faceplate type

1. Drag a frame around the desired objects with the mouse.
2. Select the "Create faceplate type" command from the shortcut menu of the multiple selection.



The faceplate editor opens.

3. In the configuration area of the faceplate editor, define which properties of the faceplate can be individually set:
   - Look for the desired property in the "Contained objects" list, for example, the foreground color of the bar graph, and drag this to the title bar of the "Interface" list.



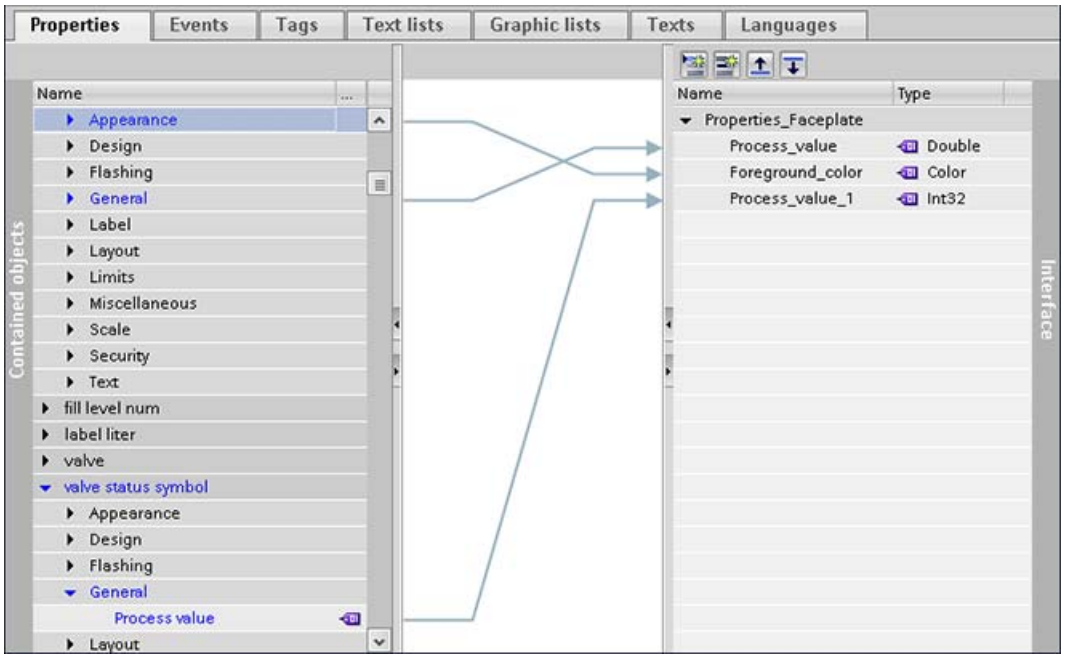
   A new property is created and linked to the property of the object contained.

   - Create a connection to a property in the "Interface" list for the following properties of the contained objects:

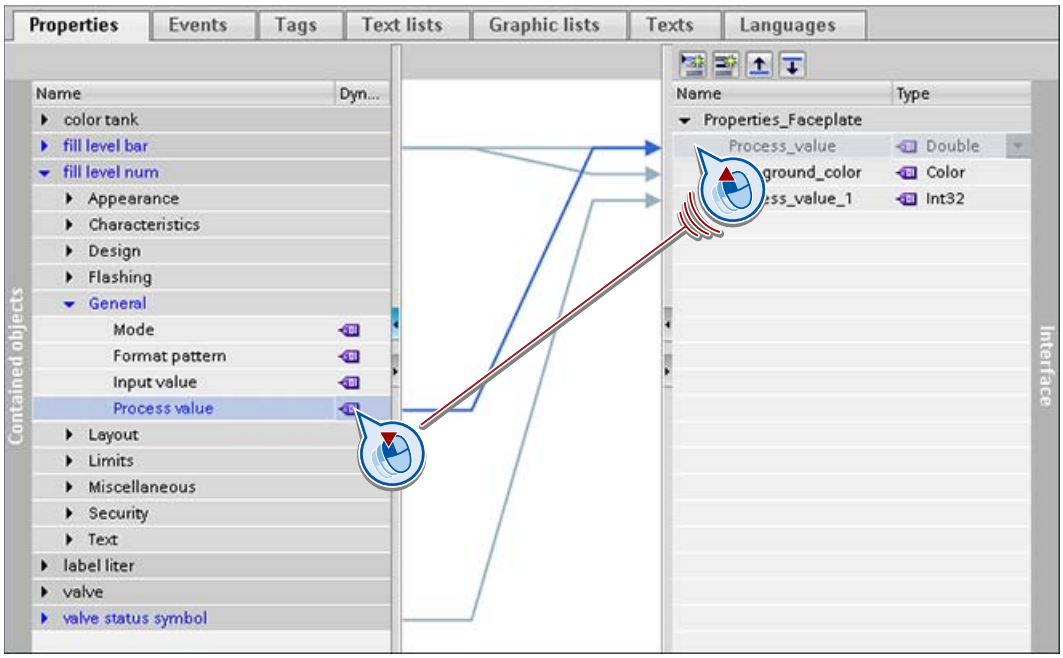
     "fill level bar > Appearance > Foreground color"

     "fill level bar > General > Process value"

     "valve status symbol > General > Process value"



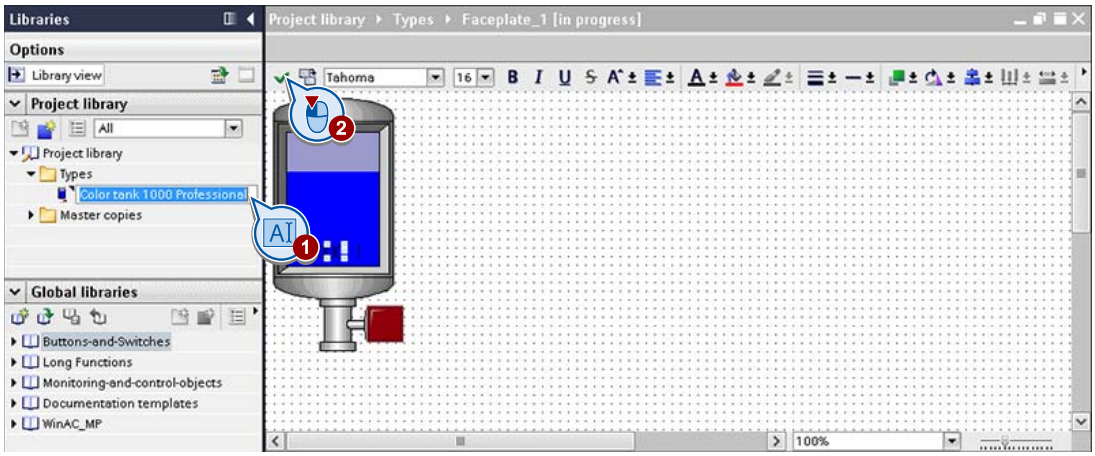
– The process value of the numerical fill display is controlled via the same tag as the graphic fill level of the graph. Drag a line from "fill level num > General > Process value" to the already created faceplate property "Prozesswert".



Note

The properties of the faceplate contain the same names as the corresponding properties of the original objects. You can change these names to provide unique names for the properties of the faceplate.

4. Enter a name for the faceplate type, for example "Color tank 1000 Professional", and release the faceplate type.



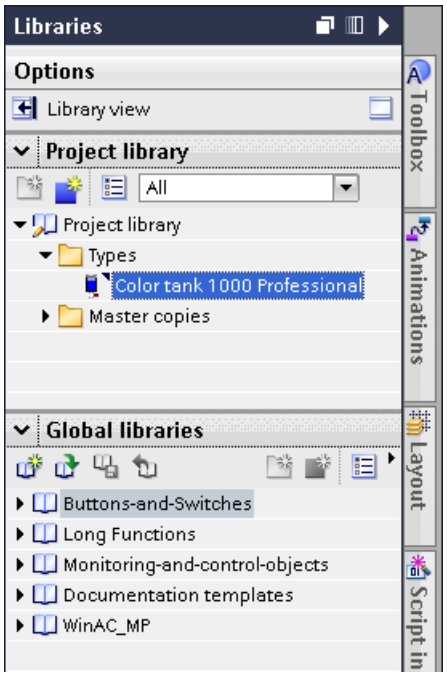
Note

Use the shortcut menu to delete connections or properties.

If you wish to change the faceplate type later, load it from the project library to the faceplate editor using the shortcut menu. Remember to explicitly release the changes you made to the faceplate type: ✔. Otherwise, your changes will not be transferred to the instances of this faceplate type, i.e. the reuses.

Result

The faceplate type is stored in the project library.



2.2.4 Using faceplates

Introduction

You need four color reservoirs. Your creation of the new faceplate type in the start screen automatically provides you with the first type instance in the screen.

Insert three more color reservoirs and configure the various instances.

Tags

You need the following tags for communication between the HMI device and the controller:

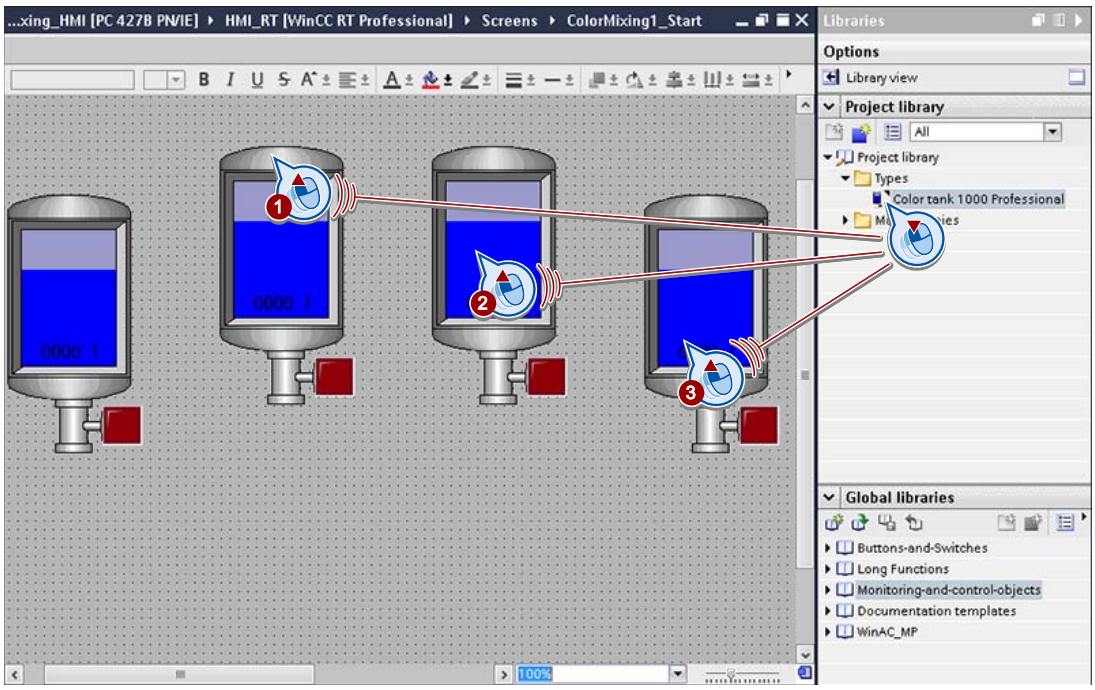
Meaning	Name	Controller connection	Data type	Acquisition cycle
Fill level cyan	HMI_Filling_FillingLevel_CMYK_C	Yes	DInt	1 s
Fill level magenta	HMI_Filling_FillingLevel_CMYK_M	Yes	DInt	1 s
Fill level yellow	HMI_Filling_FillingLevel_CMYK_Y	Yes	DInt	1 s
Fill level black	HMI_Filling_FillingLevel_CMYK_K	Yes	DInt	1 s
Valve status cyan tank	HMI_Valve_Color_C	Yes	Bool	1 s
Valve status magenta tank	HMI_Valve_Color_M	Yes	Bool	1 s
Valve status yellow tank	HMI_Valve_Color_Y	Yes	Bool	1 s
Valve status black tank	HMI_Valve_Color_K	Yes	Bool	1 s

Requirement

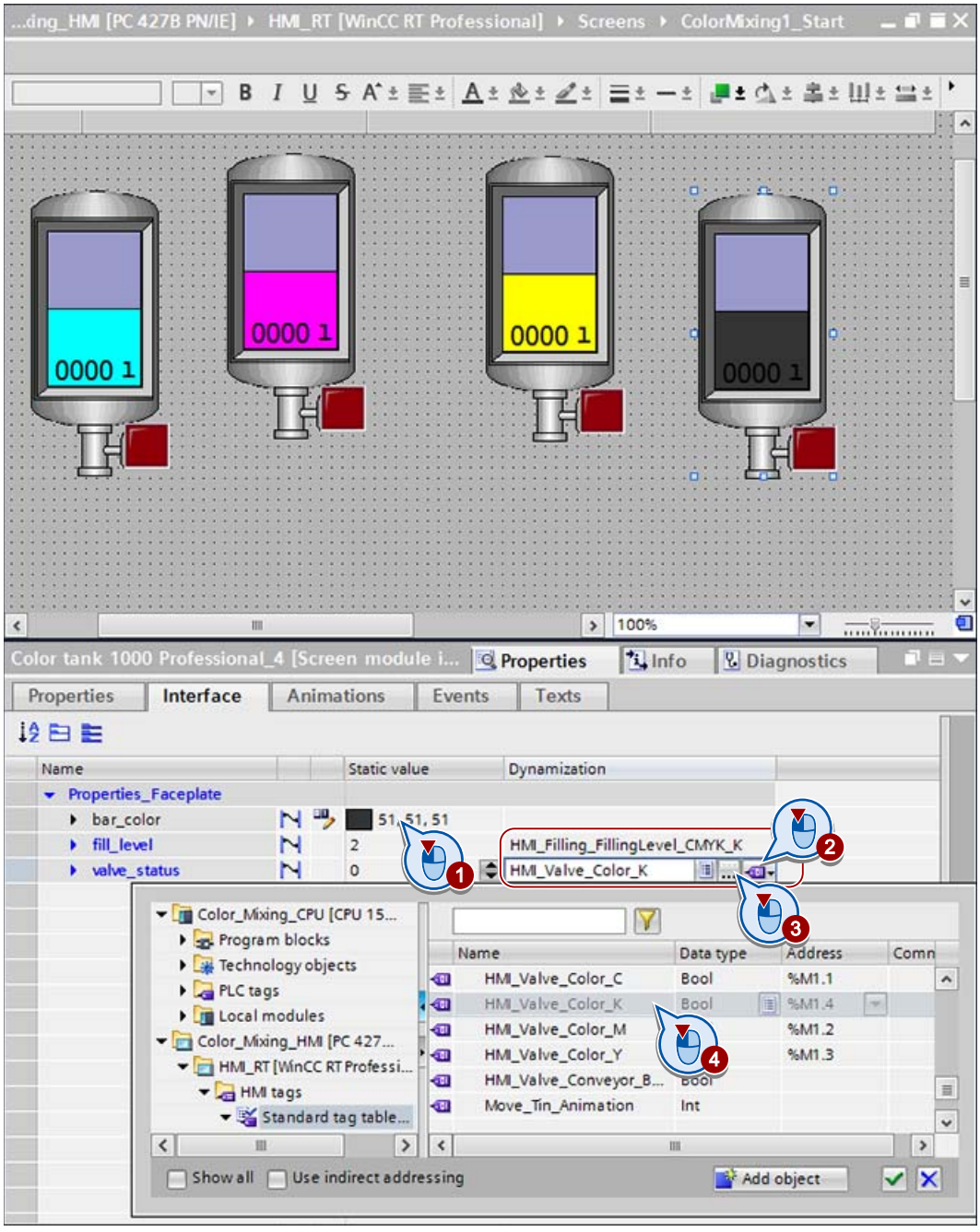
The control tags and HMI tags have been created.

Procedure

1. Drag-and-drop three more color reservoirs from the library to the "ColorMixing1_Start" screen of the HMI device.



2. Select "Properties > Interface" in the inspector window to configure the customizable properties of each color reservoir:
   - Select the foreground color for the bar graph.
   - Select the process tag that displays the fill level, e.g. "HMI_Filling_FillingLevel_CMYK_K" for the black color, both for the graphic and the numerical fill level display.
   - Select a process tag that shows the valve status, e.g. "HMI_Valve_Color_K" for the black color tank.



Result

You have successfully configured a reservoir with display of the fill level and valve state in the screen for each printing color.

2.3 Visualizing the mixer

Introduction

Configure a mixing tank with mixer. The process steps are animated:

- The mixer icon flashes while mixing is in progress.
- While the mixed color is being filled, the color discharge from the tank is visualized by a rectangle with the corresponding color.

Objects

You need the following objects:

- Mixing tank
- Mixer
- Rectangle for visualizing color discharge

Tags

You need the following tags for communication between the HMI device and the controller:

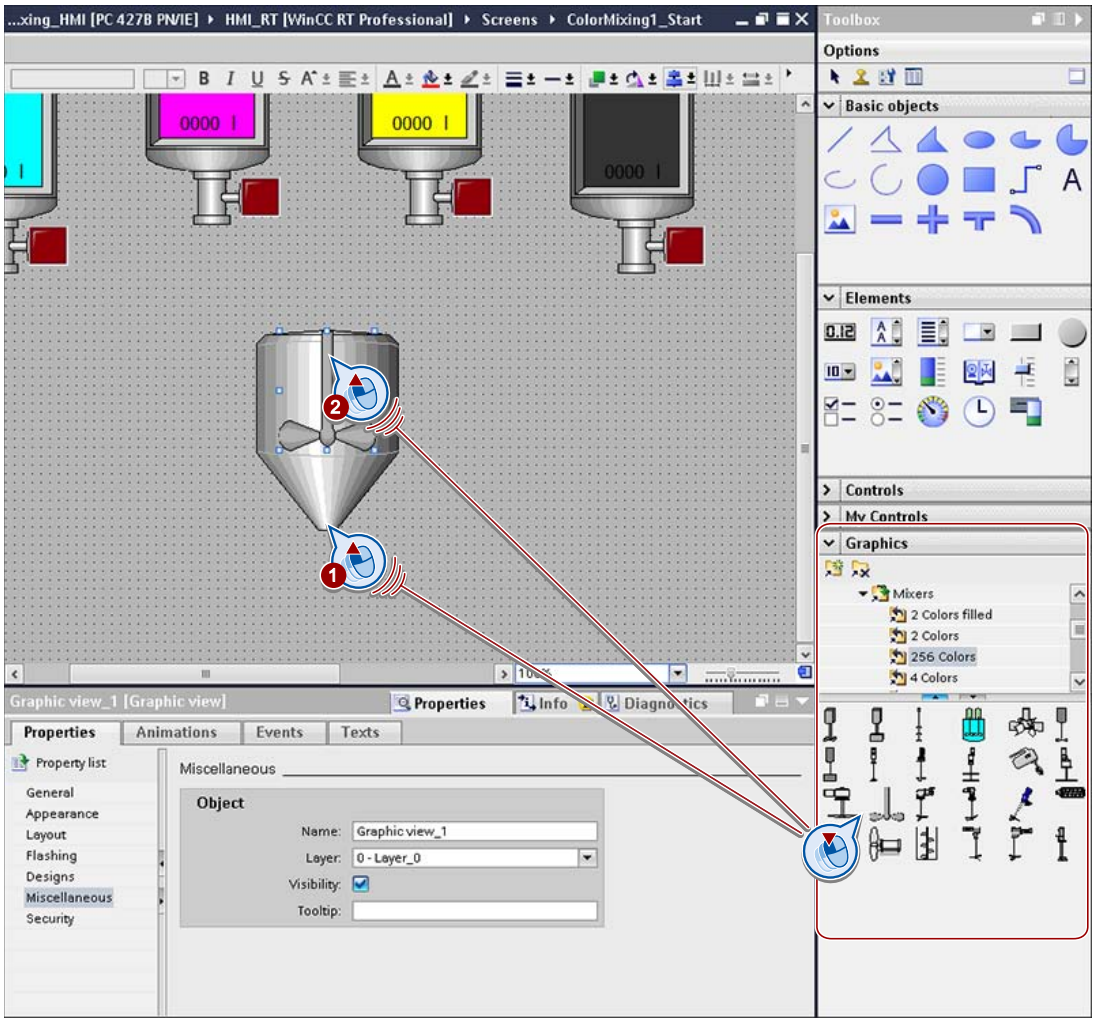
Meaning	Name	Controller connection	Data type	Acquisition cycle
Mixer status	HMI_Mixer_ON	Yes	Bool	1 s
Status of the mixing tank valve	Filling_Valve_Open	Yes	Bool	1 s

Requirement

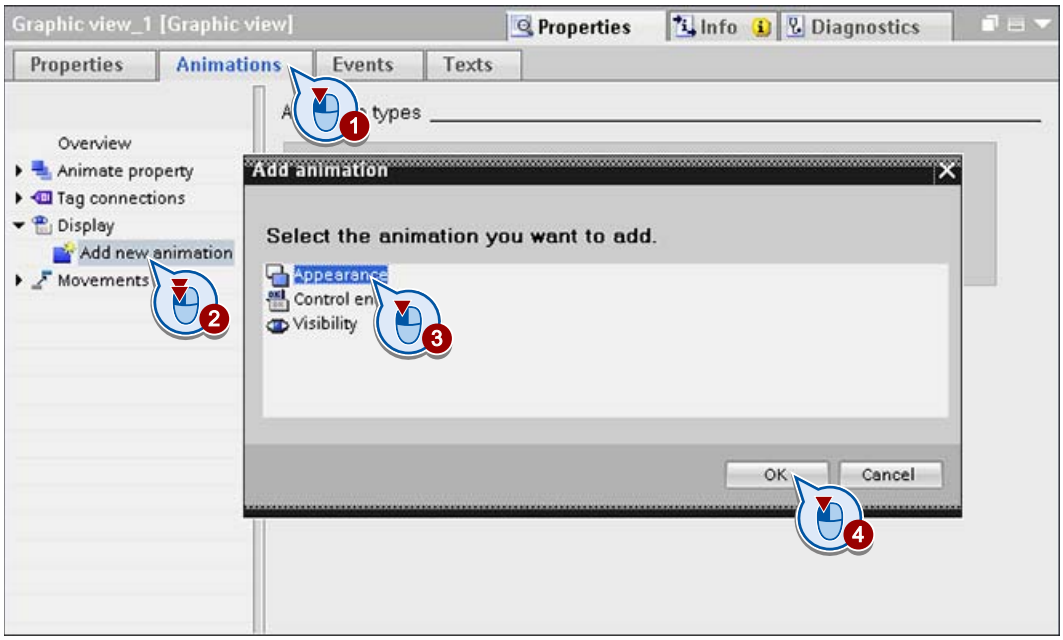
The control tags and HMI tags are set up.

Procedure

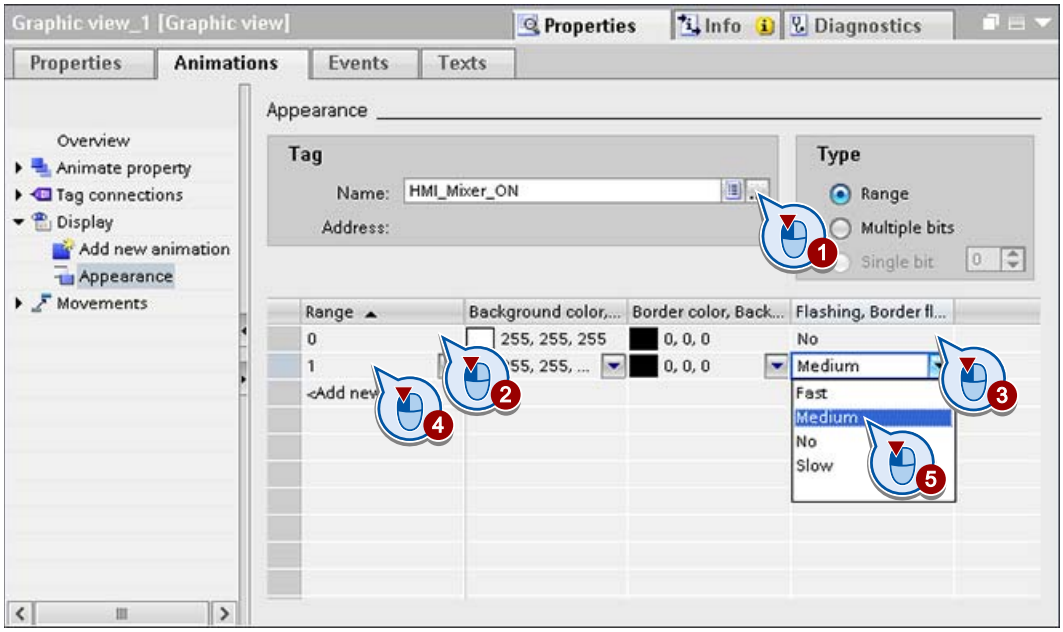
1. Drag the selected mixing tank into your screen. Mixing tanks are located in the graphics library, e.g. under "WinCC graphics folder > Automation equipment > Tanks > 256 colors".
2. Place the desired mixer over the mixing tank. Mixers are located, for example, in the "WinCC graphics folder > Automation equipment > Mixers > 256 colors".



3. To ensure that the mixer only flashes when there is activity, configure an "Appearance" type animation for the mixer.



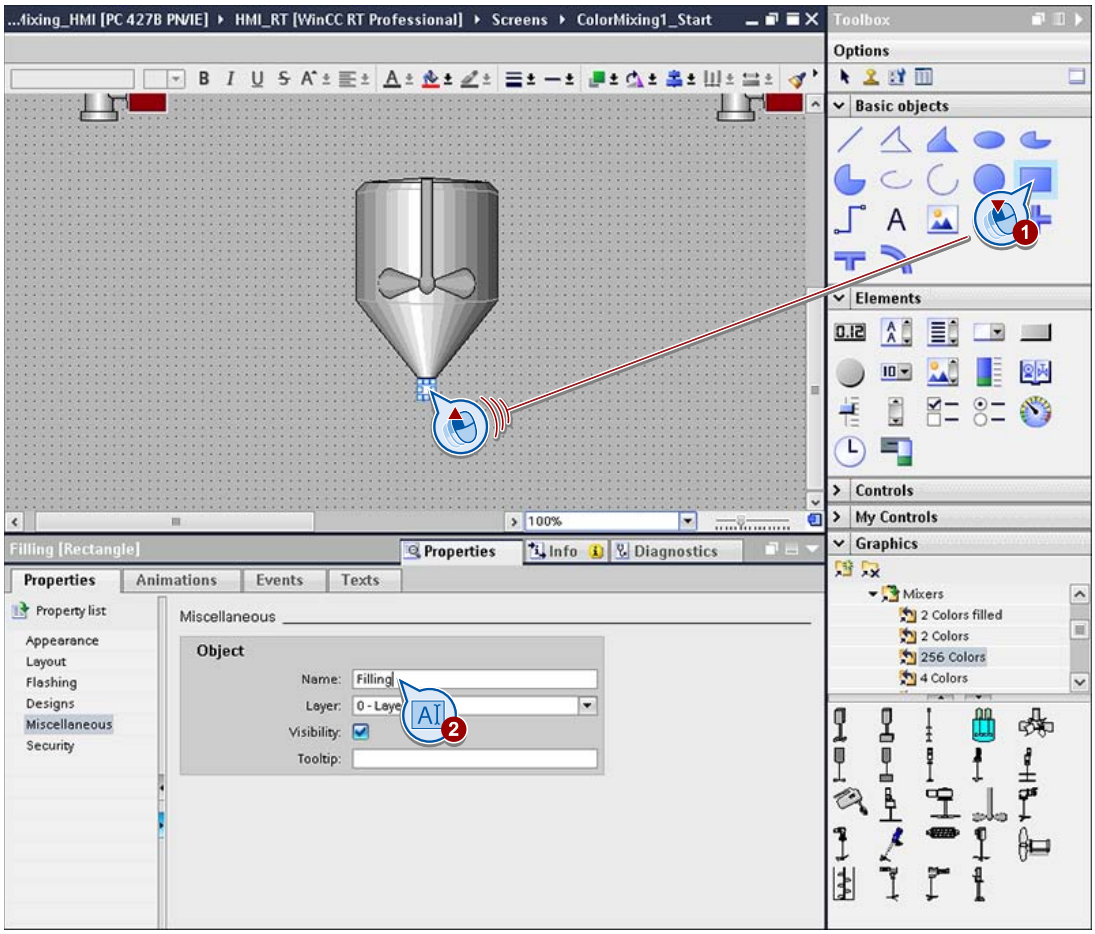
4. In the inspector window, select a tag that displays the state of the mixer, for example "HMI_Mixer_ON", and make the display flash when the tag value is "1".



Note

If an object should flash regardless of the tag value, configure the flashing in the inspector window at "Properties > Properties > Flashing".

5. To visualize color discharge from the tank, place a rectangle below the filling tank and assign it the name "Filling".



6. Configure the rectangle in the inspector window as follows:
   - Insert an animation of the type "Visibility", similar as for the flashing mixer.
   - Select the tag that displays the valve status, e.g. "Filling_Valve_Open".
   - Specify that the rectangle is visible only when the valve is open, i.e. when the tag value is "1".

Result

You have successfully configured the mixing tank and mixer. Later, you will use a user-defined function to display the rectangle in the currently mixed color.

2.4 Visualizing the feed piping

Introduction

Feed pipes between the color reservoirs and the mixing tank generate a more realistic impression.

Join the individual predefined pipe segments to construct the feed pipes. Use the proposed keyboard shortcuts to do this.

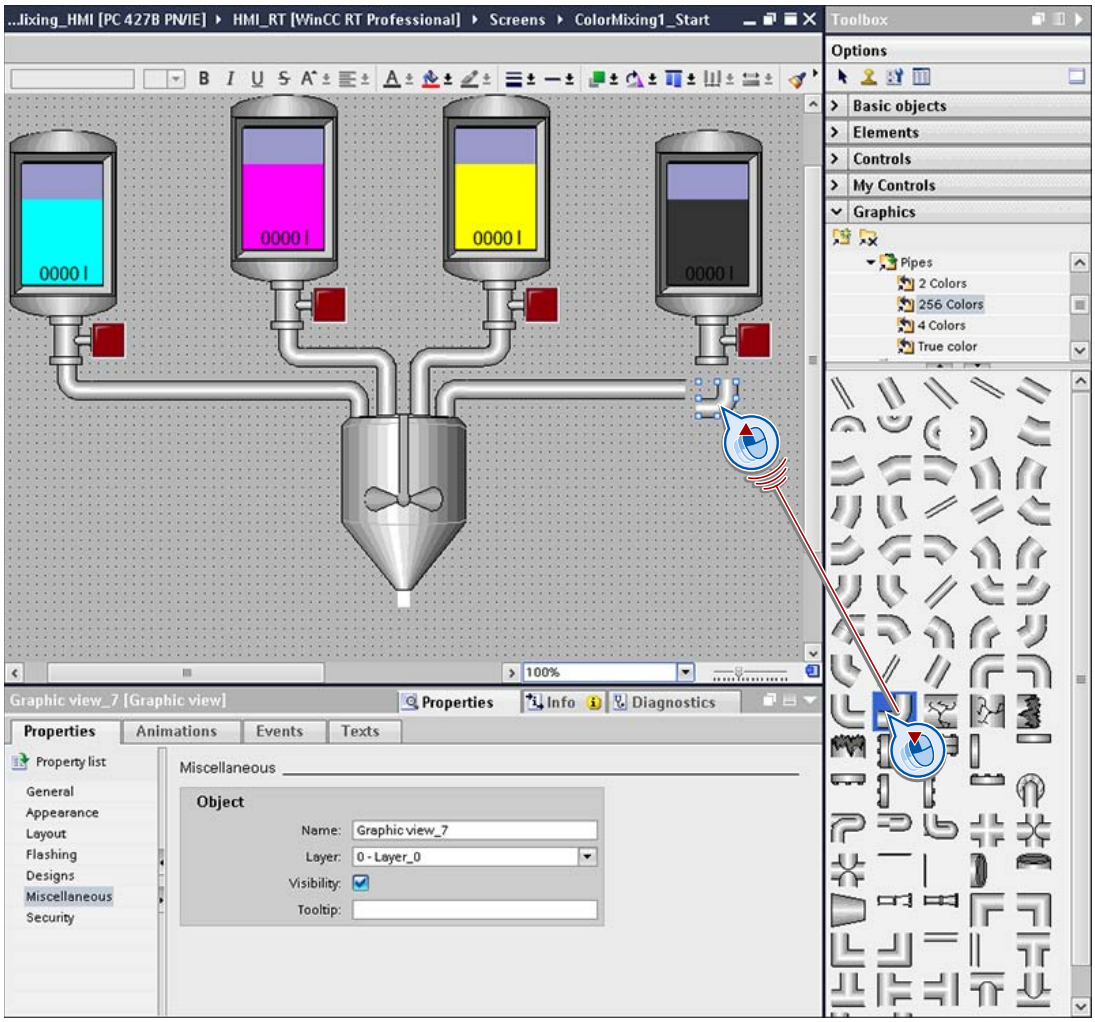
Objects

You need the following objects:

- Pipe segments

Procedure

1. Join suitable pipe segments to construct the feed pipes.



Utilize the following options:

- You can scale an object by dragging the corner handle of the rectangle enclosing the object to obtain the relevant size.
- You can fit the pipe segments precisely using the following keys:

Target	Keys
Shifting in micro steps	<Arrow keys>
Scaling in micro steps	<Ctrl+arrow keys>

Result

The feed piping is visualized on the screen.

2.5 Visualizing the conveyor, including counter and emergency stop switch

Introduction

The mixed color is filled into tins. A conveyor positions empty tins beneath the discharge nozzles of the mixing tank and transports the filled tins to the palletizing station.

Use a standard graphic from the library to visualize the conveyor belt. A counter indicates the current number of tins filled. Operators can stop the entire process by means of the emergency stop switch.

Objects

You need the following objects:

- Conveyor
- Text box and I/O field for the counter
- Off switch

Graphic icons

You need the following two icons to display the off switch:

Meaning	Name
Switch is pressed	Pushbutton_Stop_On_256c
Switch not pressed	Pushbutton_Stop_Off_256c

Tags

You need the following tags for communication between the HMI device and the controller:

Meaning	Name	Controller connection	Data type	Acquisition cycle
Off switch status	HMI_STOP_RESET_Filling_Process	Yes	Int	1 s
Number of filled color tins	HMI_Filling_Counter	Yes	Int	1 s
Conveyor status	Start_Conveyor	Yes	Bool	100 ms

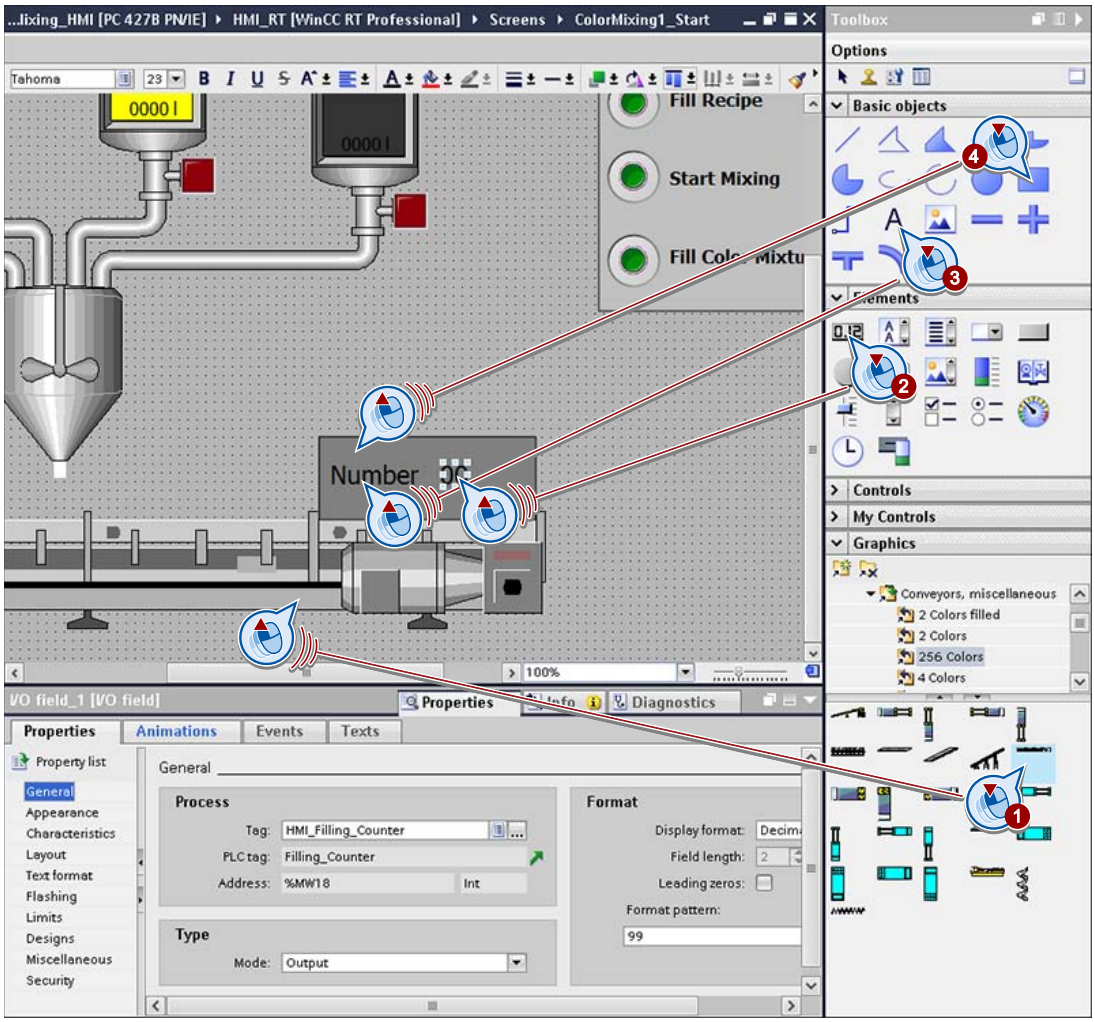
Requirement

The control tags and HMI tags are set up.

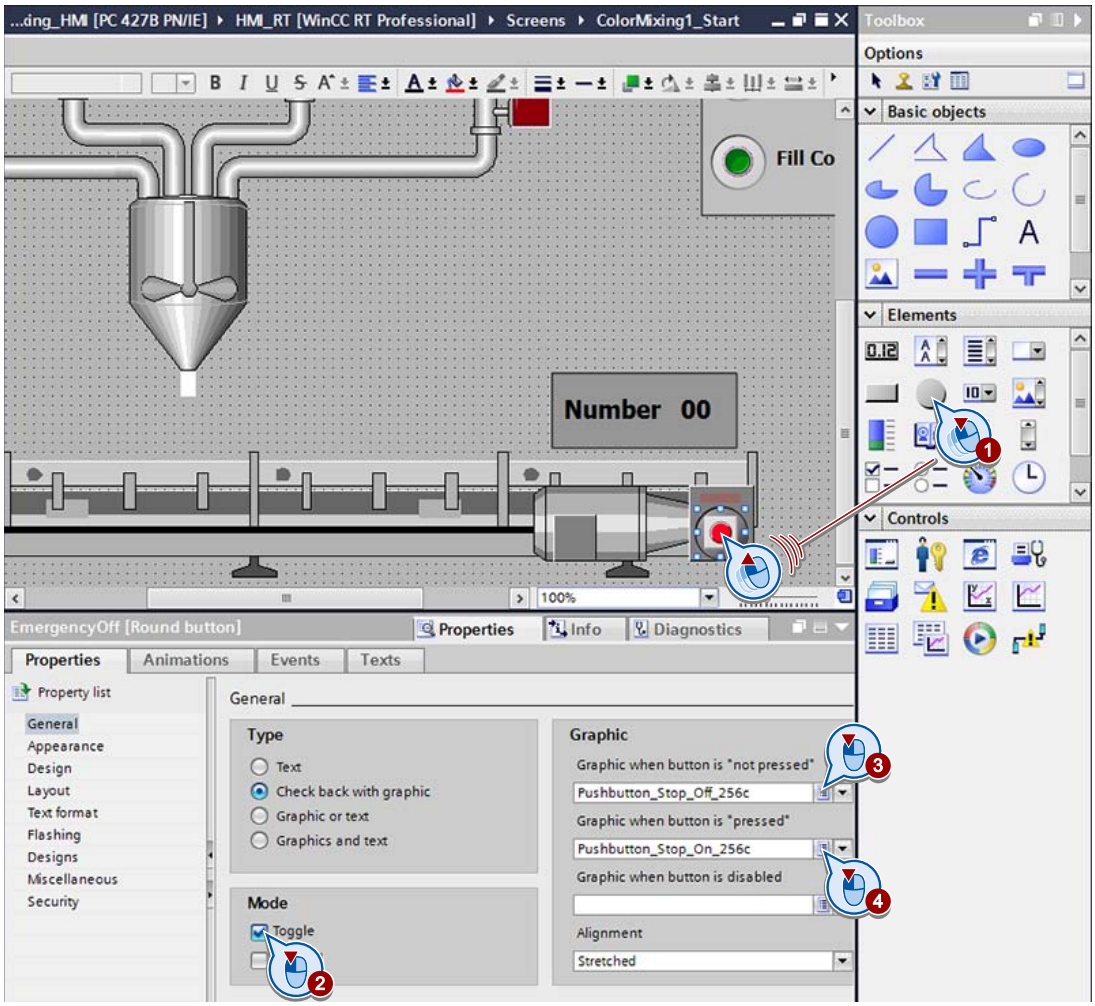
Procedure

1. Insert the icons named above into the graphics to display the off switch.
2. Drag the selected conveyor from the graphics library to the screen, e.g. from "WinCC graphics folder > Automation equipment > Conveyors, miscellaneous > 256 colors", and adjust the size as required (1).

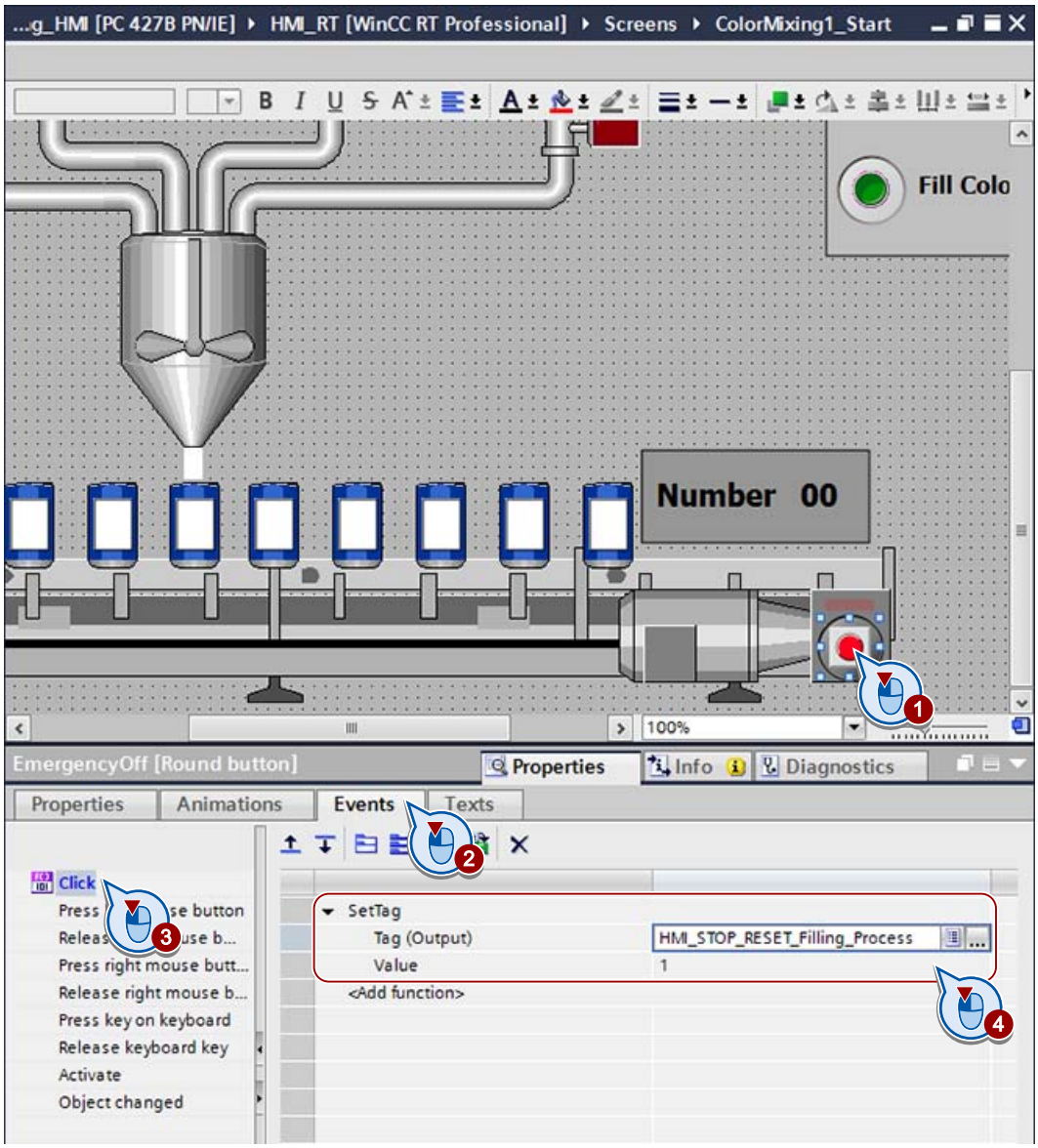
3. To display the number of filled tins:
   - Insert an I/O output field with the "Output" mode to output the "HMI_Filling_Counter" process tags (2).
   - Insert a "Number" text field as labeling (3).
   - Use a rectangle to frame this display (4).



4. For the emergency stop switch:
   - Insert a "Round button" object.
   - Select "Toggle" under mode and select the graphics to visualize the "Not pressed" and "Pressed" states.



– Finally, connect the emergency stop switch to the "HMI_STOP_RESET_Filling_Process" tag:



Result

The conveyor is configured with counter and emergency stop switch.

2.6 Animating the filling process

Introduction

The filling process is animated in the screen:

- Movement of the tins on the belt is stopped when the conveyor is switched off.
- When the conveyor is switched on, the tins are moved from left to right on the conveyor.

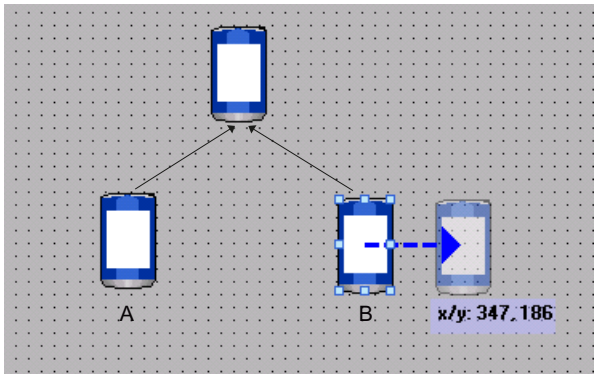
There is space for 12 tins on the conveyor. It is advisable to use a template from the project library for the tins.

Copy template "Group_Tin_Animation" for filling tanks

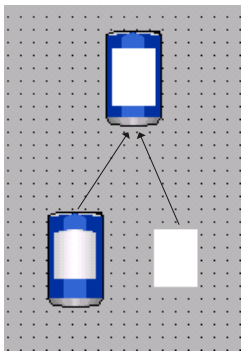
The actions listed above are configured for separate cans that are positioned directly on top of each other and are visible at different times.

- Tin A is only visible when the conveyor is switched off.
- Tin B is only visible when the conveyor is switched on. Tin B is moved one position further.

Visibility and horizontal movement are configured as animations in the inspector window.



To color the labels of the filled cans in the respective color, overlay the filling tanks from the graphic library with a rectangle, for example, as new, modifiable label. The rectangles can then be colored accordingly using a user-defined function.



If you want to reconfigure the template, note that you have to configure the visibility of the animation for both the filling tank graphic and the rectangle. Use functions such as grouping for effective configuration.

Tags

You need the following tags for communication between the HMI device and the controller:

Meaning	Name	Controller connection	Data type	Acquisition cycle
Conveyor status	Start_Conveyor	Yes	Bool	100 ms
Trigger for animation of the filling process	Move_Tin_Animation	No	Int	1 s

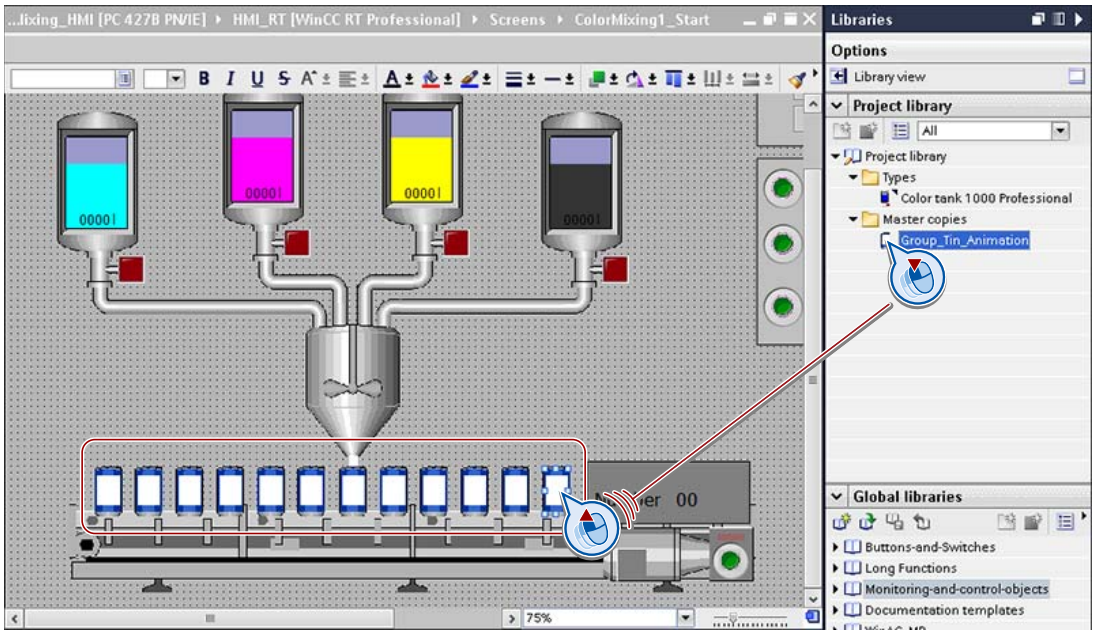
Requirement

The control tags and HMI tags are set up.

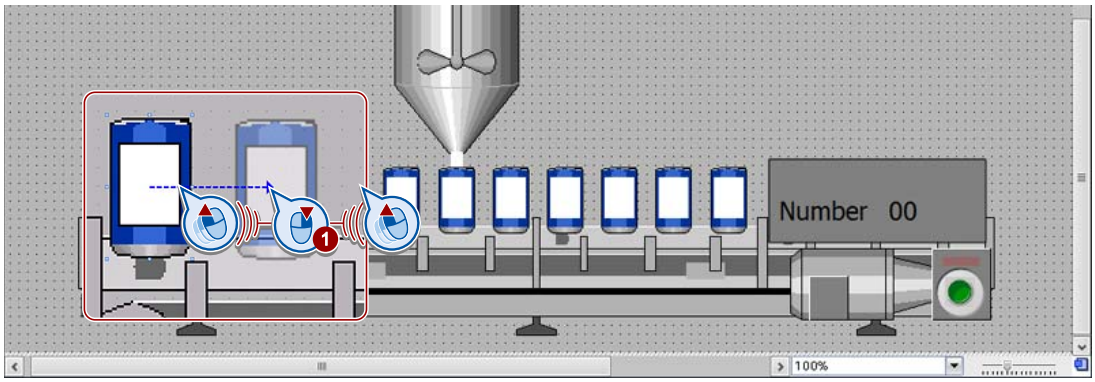
Procedure

1. To display the tins, drag 12 copies of the template "Group_Tin_Animation" from the project library to the screen.
   - Position a copy in each field of the conveyor from left to right, one after the other.

     Make sure that the copies are numbered consecutively from 1 to 12. The copy templates with numbers 7 to 12 are referenced in the "startscreen" script by their names. Open the "startscreen" script, if necessary, and run a syntax check. Information on working with the "Scripts" editor is available in this Getting Started in the chapter "Creating user-defined functions".
   - Select all tins by enclosing the objects within a rectangle.
   - Use the alignment tools to align the tins with even horizontal spacing in a horizontal line (top or bottom).



2. To set up the precise traversing distance, adjust the target position for the "Tin_Group_Move[n]" group contained in each copy so that the preview overlaps the respective next tin at the target position. To do this, drag the tip of the arrow in the screen to the left or right, for example.



3. Position the conveyor in such a way that one tin is positioned directly underneath the filling nozzle, with six tins to the left of it and five tins to the right.

Result

The filling process is animated in the screen.

Use a user-defined function "startscreen" to set the currently mixed color for the rectangles for the labels on the filled tins:

- Labels for the six tins to the left of the mixing tank - white
- Labels for all other tins - in the current color

Testing

The configuration should be tested in runtime following each extensive editing step. This way, inconsistencies and display errors can be detected and removed in due time.

A detailed description is available in the section "Testing and loading a project" of this Getting Started.

Configuring screen navigation 3

3.1 Basics on screen navigation

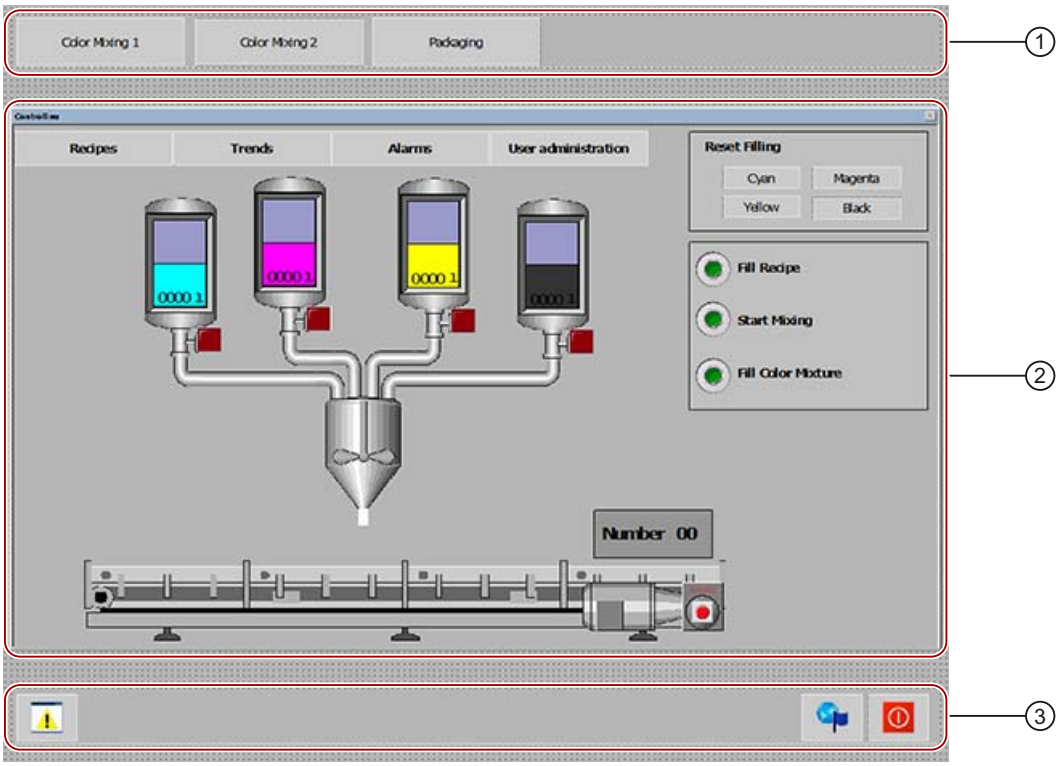
Introduction

In our example, you operate several plant parts (two color mixing stations and the packaging station) with one HMI device.

The operating screens of the plant parts should be created independently of one another for easier integration of the application. It must be possible to change between operating screens of the plant parts without affecting already configured operating screens. For this reason, navigation between plant parts and global functions is implemented separately from the operating screens. We use the screen window method for this purpose. Screen windows are placed in a global plant overview screen in which a pure navigation screen, a screen for triggering global functions, and the operating screens of the plant parts are displayed next to each other.

Implementing the screen navigation

To operate the plant, we need a basic screen with three areas on the HMI device:



① Plant navigation to switch between the three plant parts. This navigation area always shows the same content.

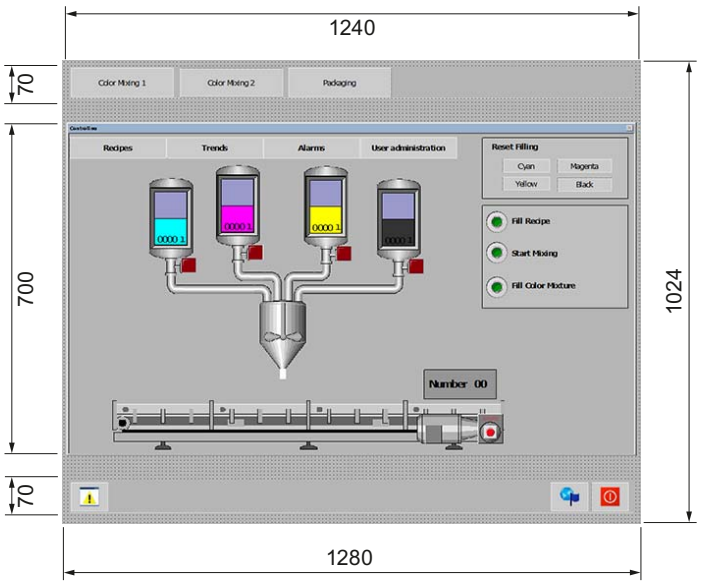
You create the "PlantNavigation" screen shown in this screen window during the configuration.

② Work area to operate and monitor the selected plant part. This area shows the operating screens in the navigation area of the selected plant part. Screen changes within a plant part are executed in the work area.

③ Functions for calling globally valid functions and screens. This area always shows the same content:

- Display of pending alarms
- Language switches
- Exit of runtime

This Getting Started requires a screen with 1280 x 1024 pixels. The following sizes are therefore available for the three navigation areas:



The screen windows at the top and bottom each have a height of 70 pixels. The work area in the center is 700 pixels high. The width is the same everywhere, with 1240 pixels. All screens in the project have been optimized for the respective area so that they can be displayed in the associated screen window without scaling. This means that all operating screens for the work area have been created with 1240*700 pixels; the screens for navigation and global functions with 1240*70.

Screen change with screen windows

If you use the screen window method, the screen change configuration changes: You must also specify in which screen window the screen change will be executed. The following system functions are available for this purpose:

- ActivateScreenInCurrentScreenWindow

  Executes the screen change in the active screen window. This system function is used to change between the screens of the selected plant part in this Getting Started. This navigation is already prepared in the sample project.

- ActivateScreenInScreenWindow

  Executes the screen change in any screen window. In this Getting Started, this system function is used to change between plant parts.

  Important parameters:

  – Screen name: Screen in which the screen window is located
  – Screen window: Screen window in which the screen change is executed
  – New screen name: Screen in which the screen window is to be displayed

Task

The main screen "PlantOverview" is already contained in the sample project. You insert two additional screen windows into this screen:

- One screen window to display the operating screens of the various plant parts
- One screen window with a screen for navigation between the plant parts

Steps to complete

- Create a screen to change between the plant parts
- Extend the main screen "PlantOverview" with two additional screen windows
- Configure screen changes on the navigation buttons

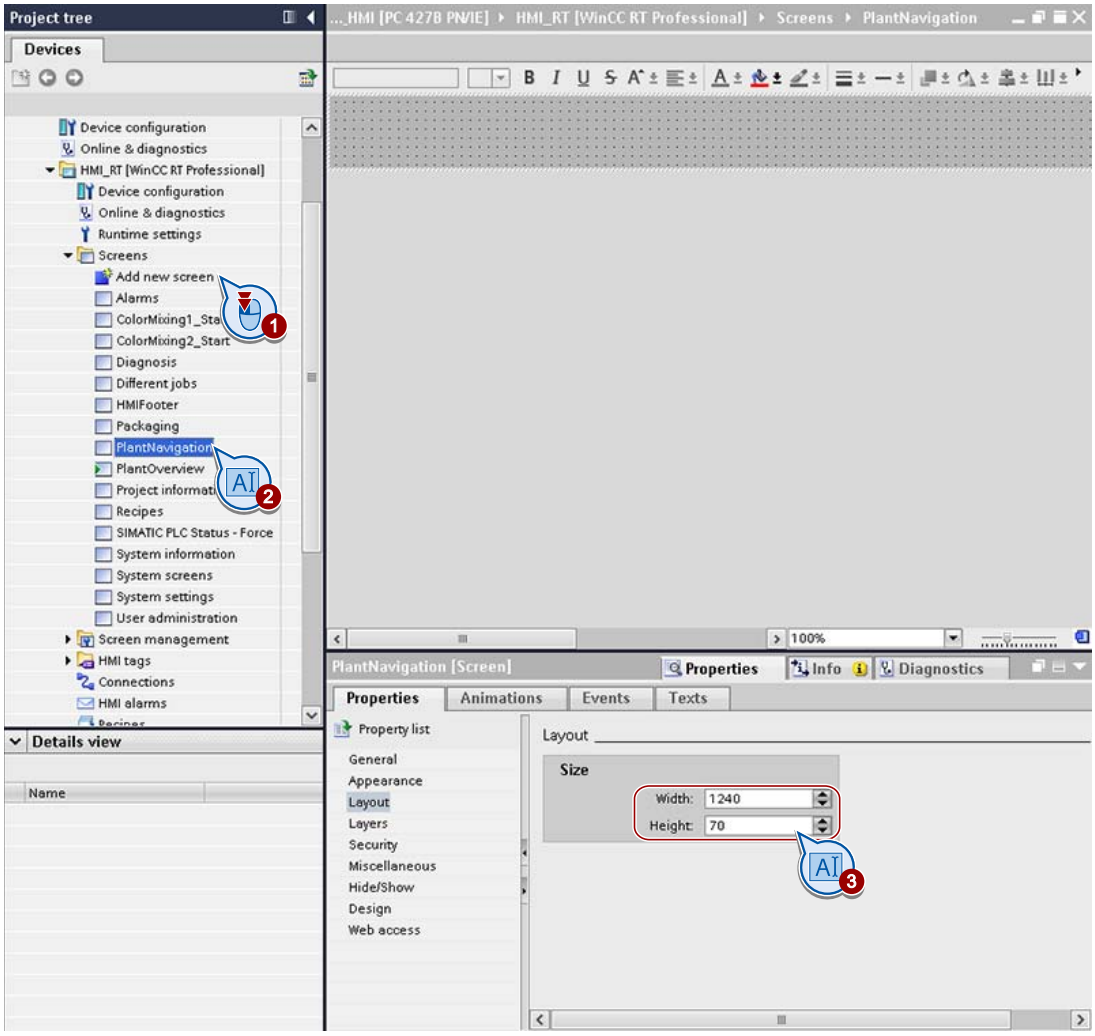
3.2 Creating a screen to display other plant screens

Introduction

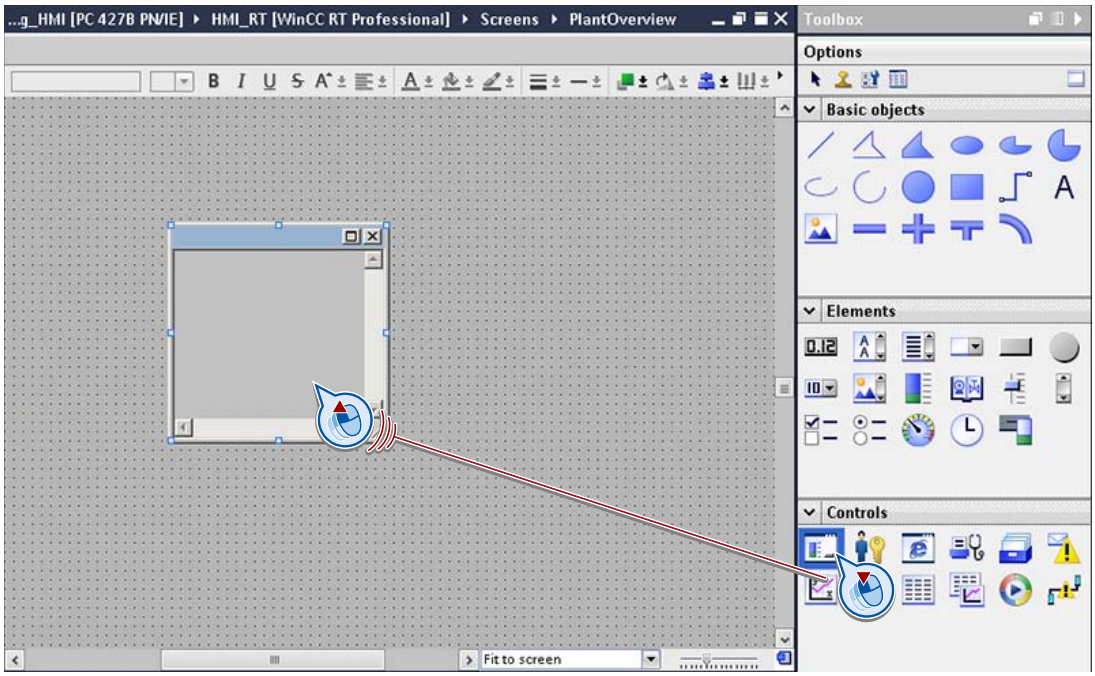
Three plant parts are operated and monitoring using this HMI device. For this purpose, first create a screen to change between the plant parts. Then add two screen windows for displaying the navigation and work area to the main screen "PlantOverview".

Procedure

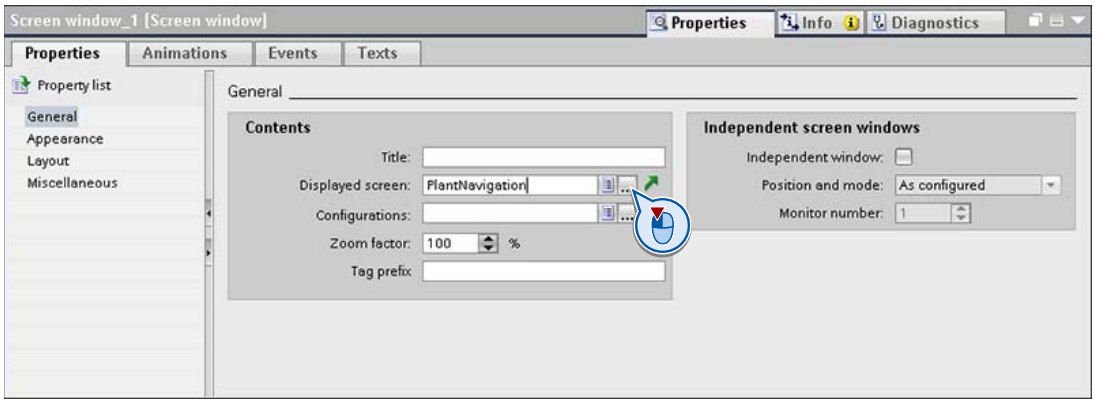
1. Create the "PlantNavigation" screen and adjust its size.



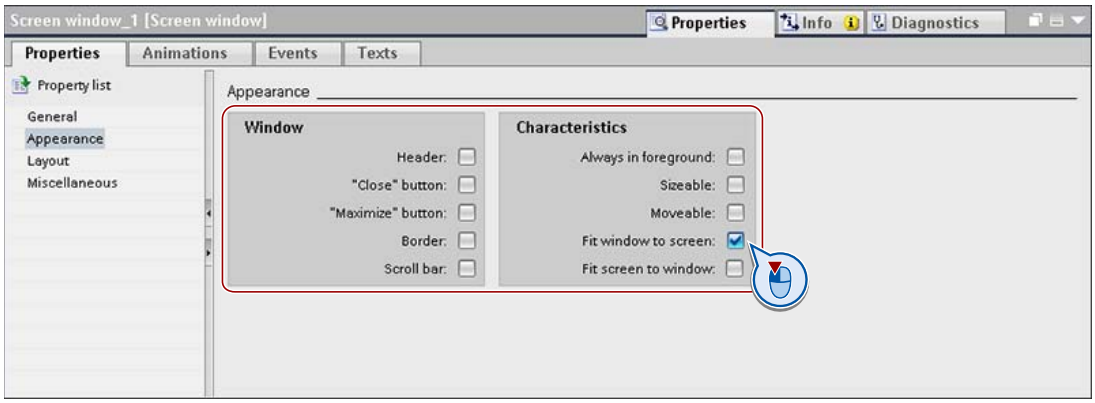
2. Open the "PlantOverview" screen and insert a screen window.



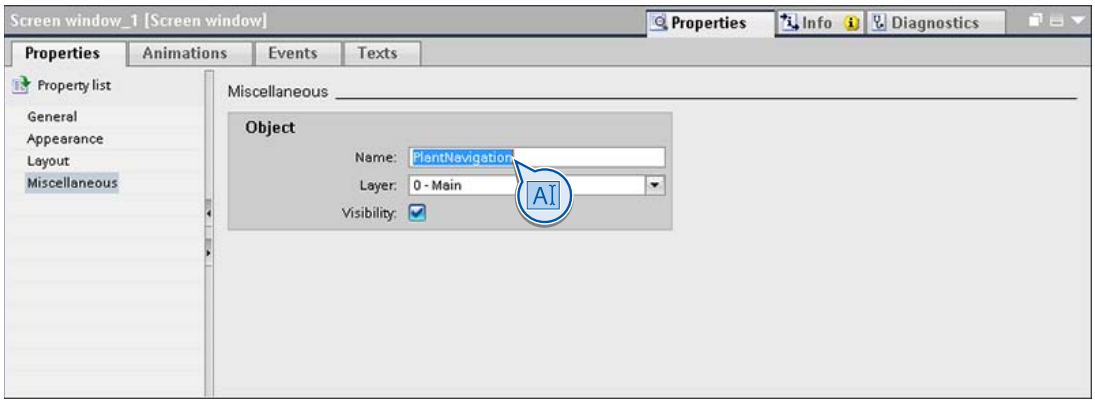
3. Configure the screen window in the inspector window:
   – Select the screen to be displayed in the screen window.



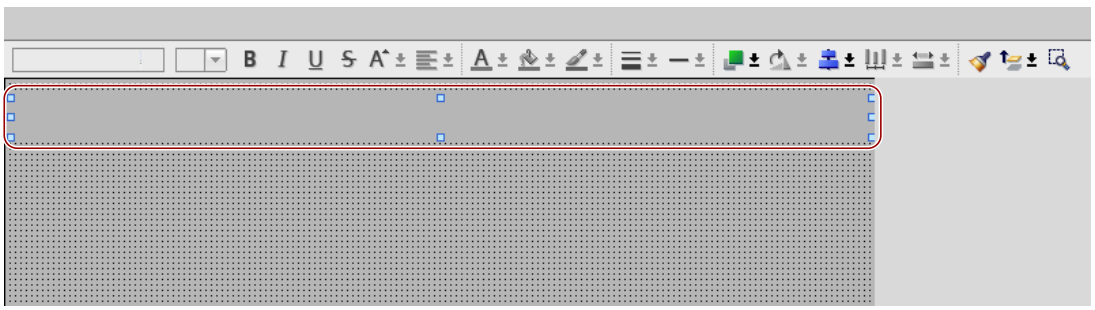
– Adjust the appearance of the screen window.



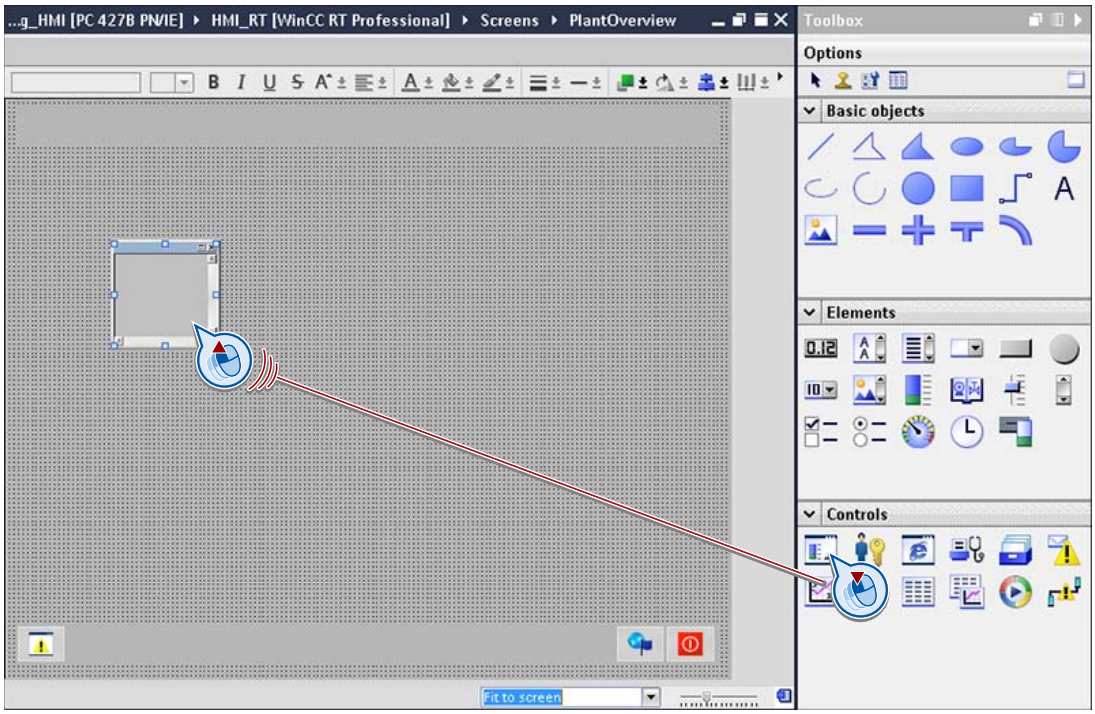
– Enter the name of the screen window.



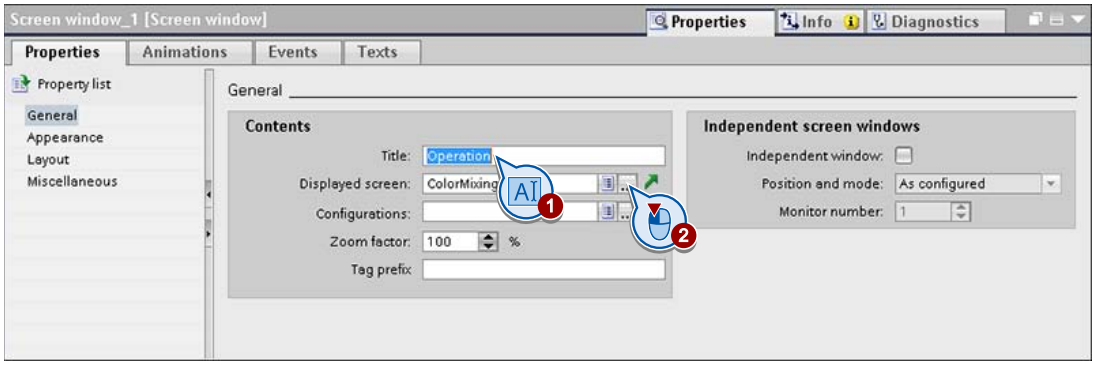
4. Position the screen window at the top edge of the screen.



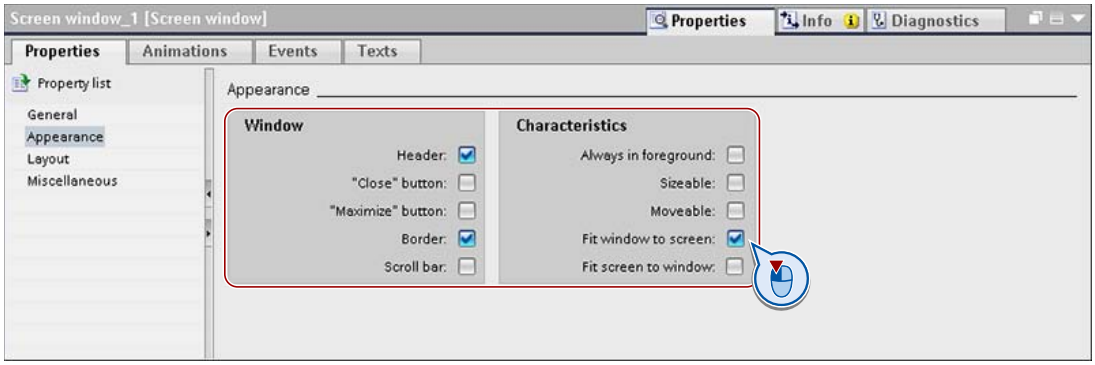
5. Insert the screen window to display the operation of the plant.



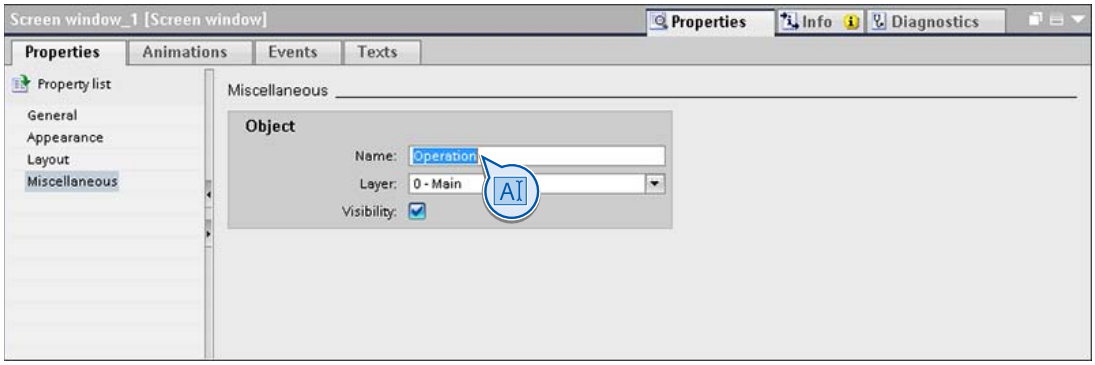
6. Configure the screen window in the inspector window:
   – Enter a title and select the screen to be displayed in the screen window.



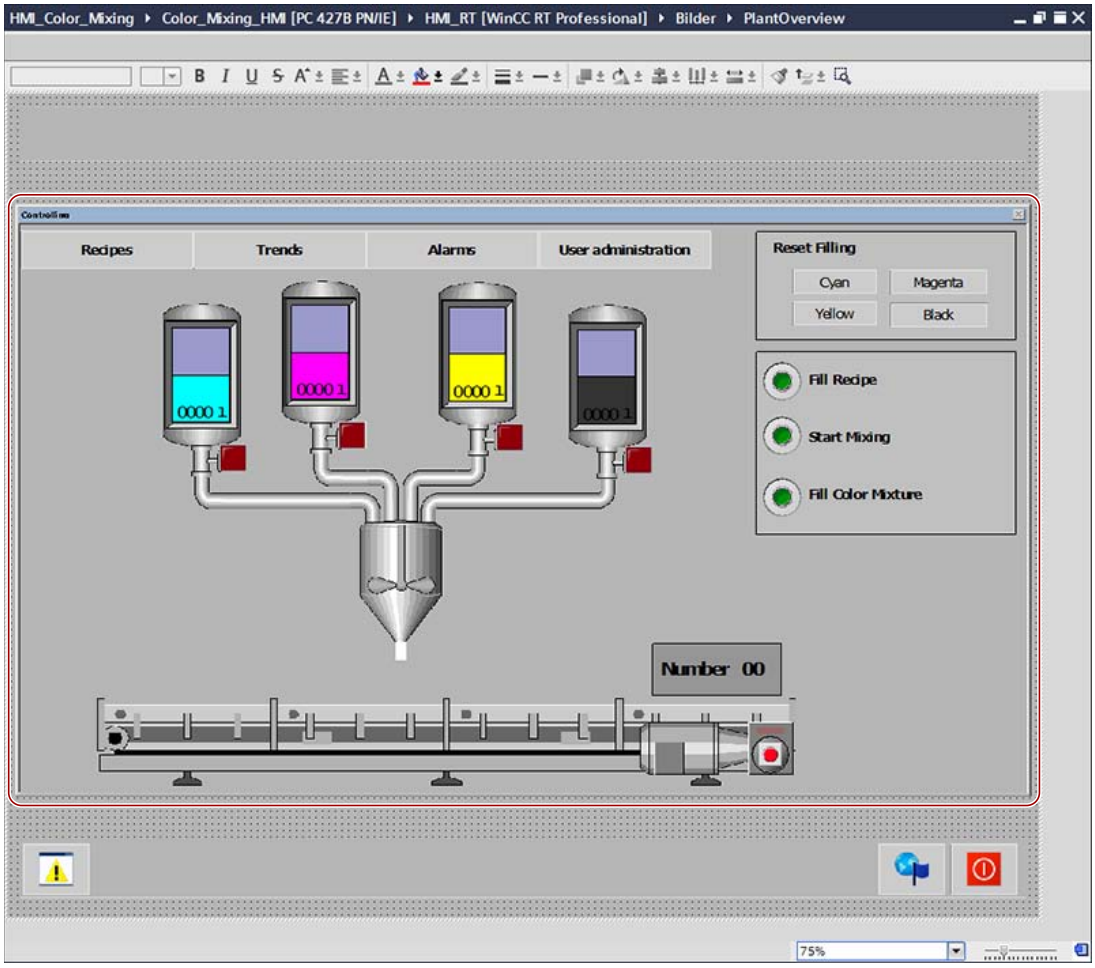
– Adjust the appearance of the screen window.



– Enter the name of the screen window.



7. Finally, position the screen window.



Result

Two screen windows have been added to the main window.

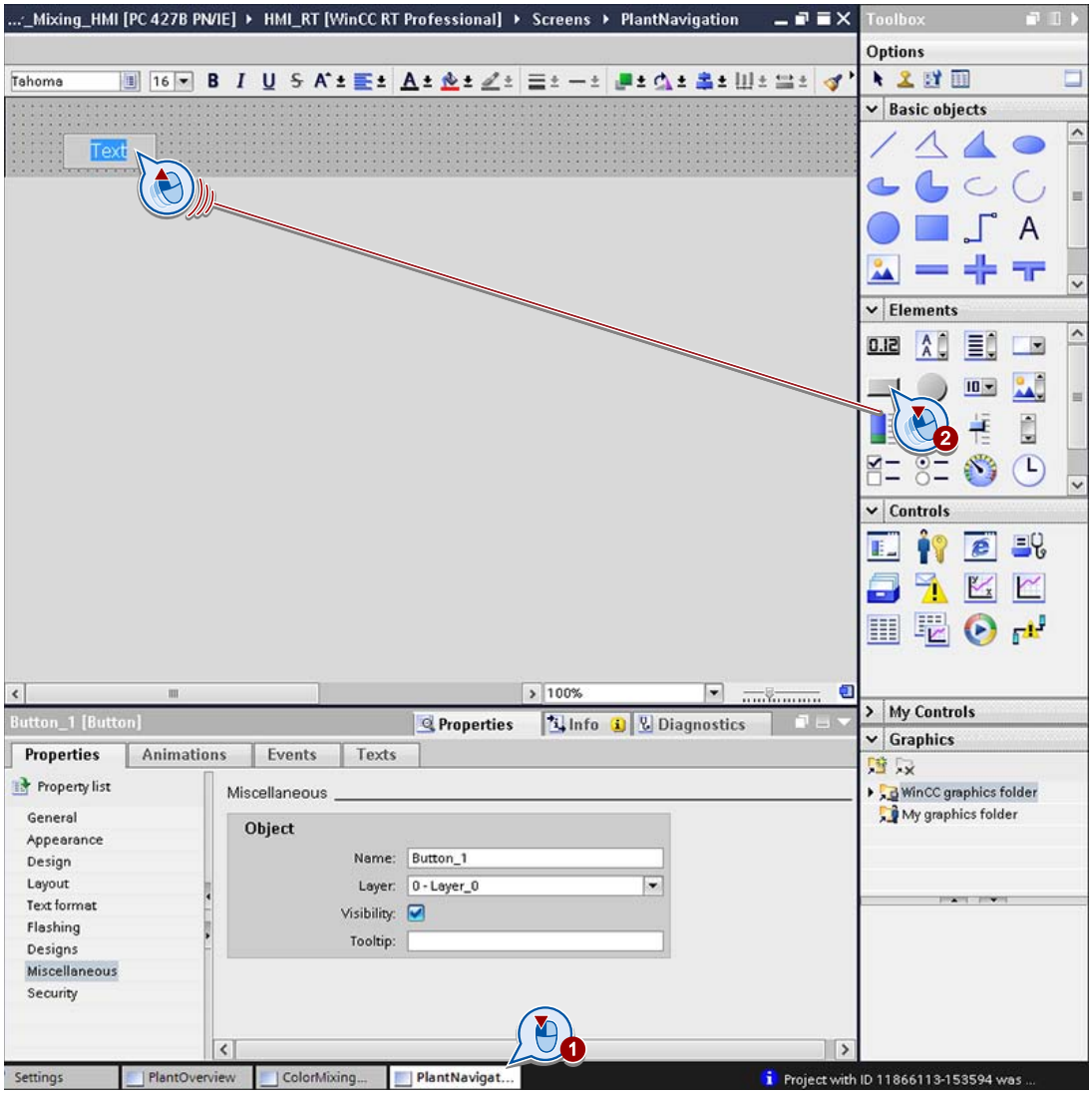
3.3 Creating navigation buttons

Introduction

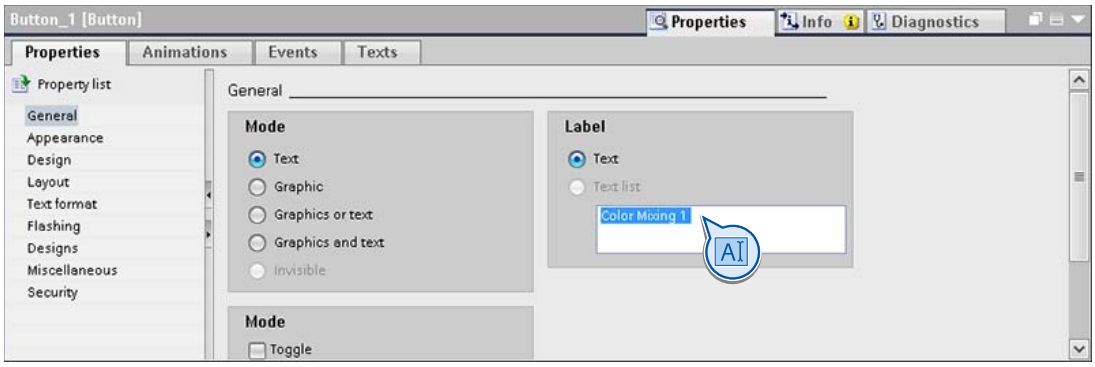
To switch between the plant parts in the main screen "PlantOverview", insert three buttons in the "PlantNavigation" screen.

Procedure

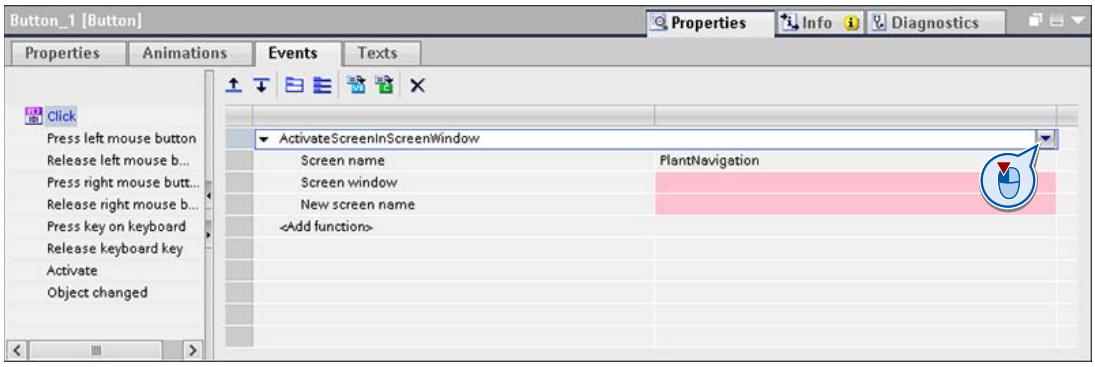
1. Switch to the "PlantNavigation" screen and insert a button.



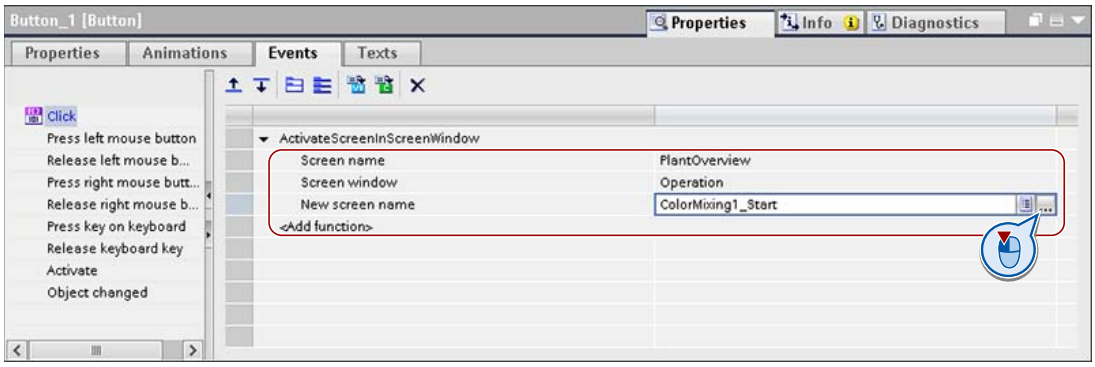
2. Configure the button in the inspector window:
   – Label the button.



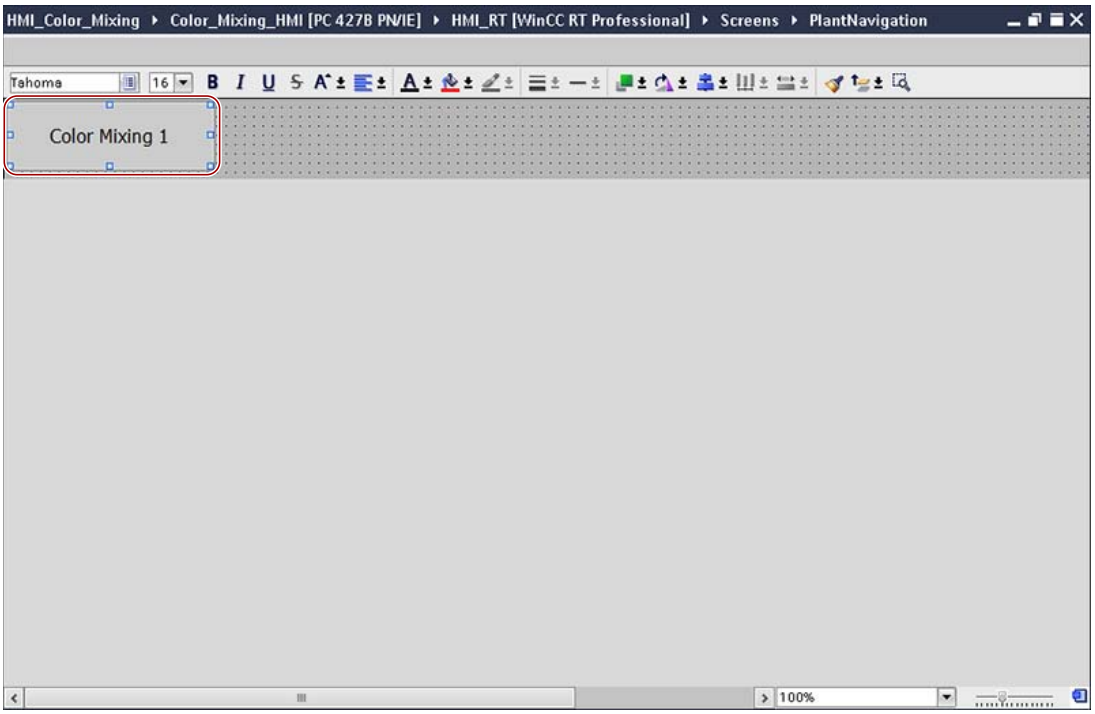
   – Add the screen change function.



   – Assign parameters to the function.



3. To conclude, adjust the size of the button.

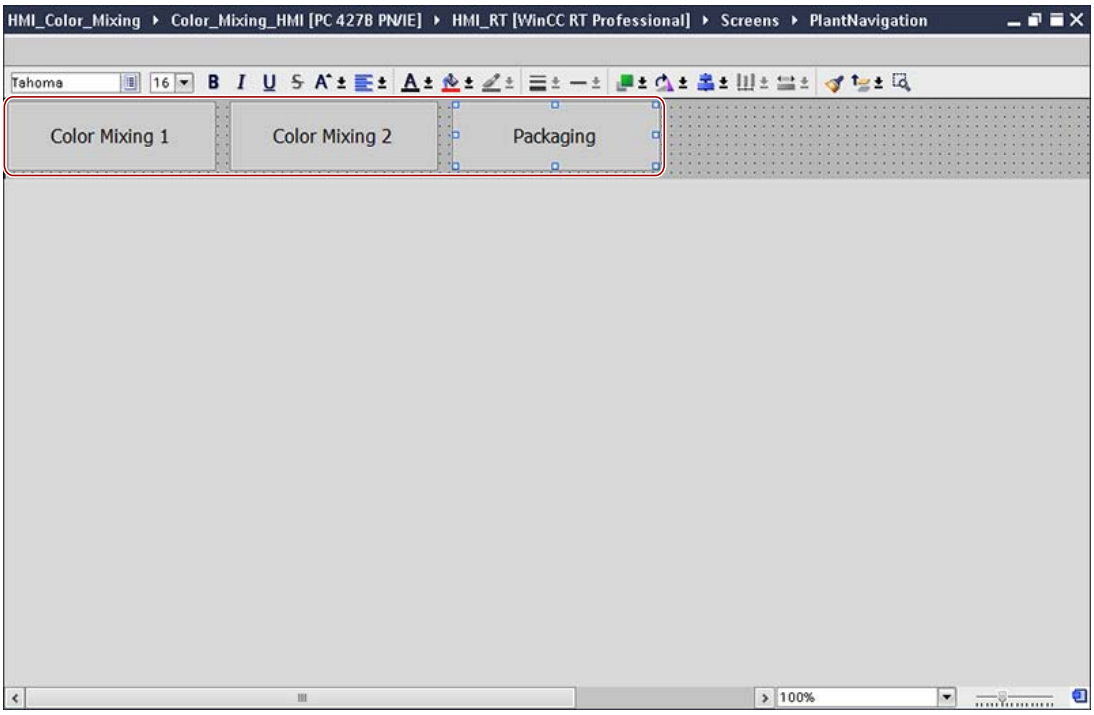


You will use this button as the basis for all additional screen changes. In each copy, adjust the following:

– Labeling

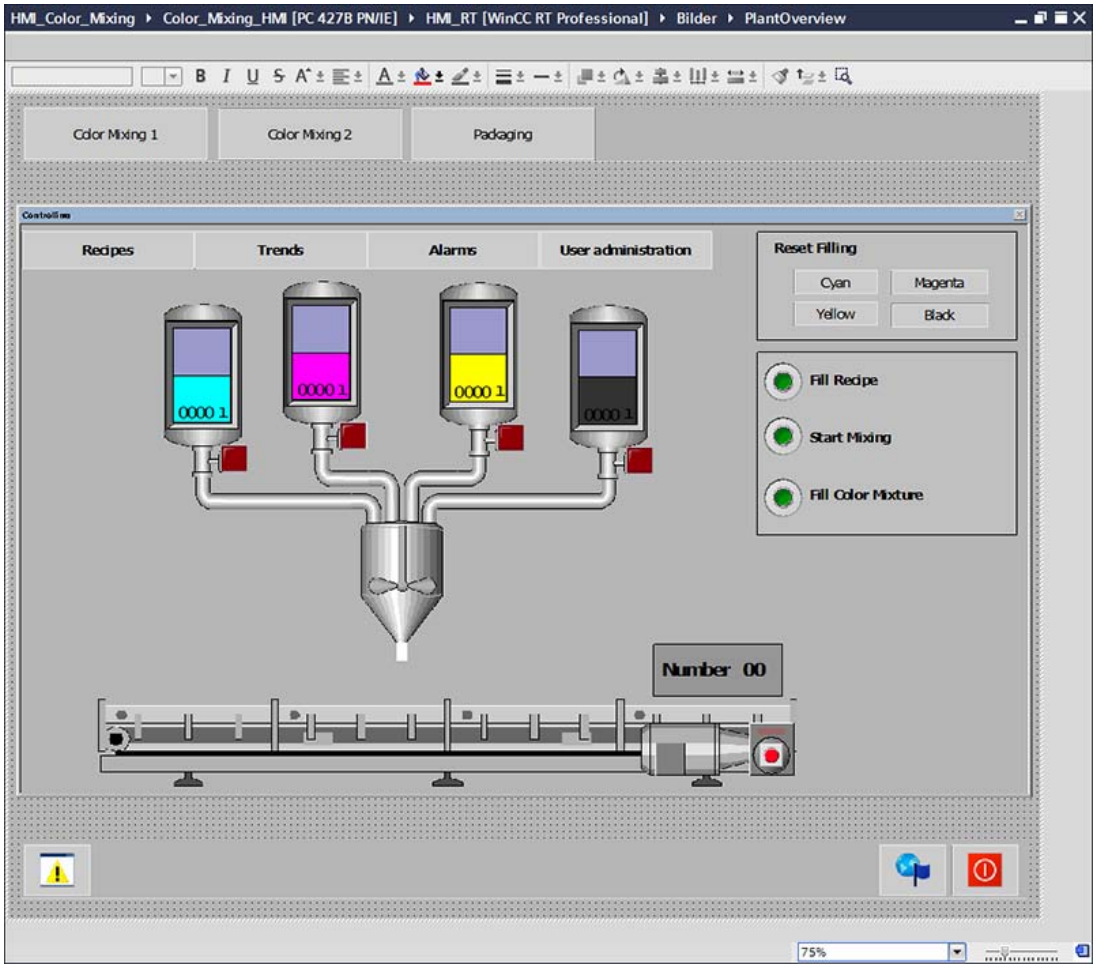
– "New screen name" function parameter: Screen to be activated.

4. Create the screen change to the following plant parts in the "PlantNavigation" screen:
   – "Color mixing system 2". New screen name: "ColorMixing2_Start"
   – "Palletizing". New screen name: "Packaging"



Result

Configuration of the screen navigation is complete. You can view the result in the "PlantOverview" screen:



Configuring recipes

4

4.1 Introduction

What are recipes?

Recipes combine associated production parameters. A recipe contains recipe elements and recipe data records.

- Recipe elements are, for example, machine parameters or product components.
- Recipe data records contain the composition and mixing ratio of the recipe elements for a specific product or product version.

The values of the current recipe are transferred to the controller using tags.

Example:

The color mixing station is capable of producing mixed colors from the basic colors "Magenta", "Cyan", "Yellow" and "Black". A recipe element is recorded for each basic color. The mixing ratio of the basic colors is stored in a recipe data record created for each mixed color.

How to work with recipes

The structure of a recipe is defined once at the configuration stage. The recipe data record required for the production is transferred from the HMI device to the controller in runtime.

- In automatic production, the control program triggers the transfer from the HMI device to the controller.
- In manual production, the operator loads the recipe data record to the HMI device, changes the data if necessary, and then transfers this data to the controller.

Communication with the controller

The controller and HMI device can exchange recipe data records in different ways - using control tags or using a raw data tag.

In the sample project, control tags are used for transfer. Additional information is available in the WinCC online help in the TIA Portal under the keywords "Control tag" and "Raw data tag".

Note

No communication

You can also use recipes to save data which is available locally on the HMI device without connection to the controller. In such cases, configure "No communication".

Tasks

In this chapter, you will learn the following:

- How to record a recipe for multiple color hue mixtures
- How to configure a screen with recipe view for manual production
- How to create data records with production data

The color mixing station is capable of producing the mixed colors "Orange", "Amber", "Green" and "Red". A recipe data record is created for each color. The recipe data record specifies the ratio for the basic colors that produce the relevant mixed color.

The recipe consists of the relevant parameters and the recipe data records in which the mixing ratios for the various color hues are stored.

In this chapter, you are going to create a recipe containing recipe data records for several color hue mixtures and configure a screen with recipe view.

4.2 Creating a recipe for colors

4.2.1 Creating recipes and recipe elements

Introduction

The color mixing station is capable of producing the mixed colors "Orange", "Amber", "Green" and "Red".

Before you create recipe data records for the mixing colors, create a recipe with recipe elements for the basic colors "Magenta", "Cyan", "Yellow" and "Black".

Requirement

The following tags are created:

Meaning	Name	Controller connection	Data type	Acquisition cycle
Portion of Cyan	HMI_Recipe_CurrentValue_C	Yes	DInt	1 s
Portion of Magenta	HMI_Recipe_CurrentValue_M	Yes	DInt	1 s
Portion of Yellow	HMI_Recipe_CurrentValue_Y	Yes	DInt	1 s
Portion of Black	HMI_Recipe_CurrentValue_K	Yes	DInt	1 s

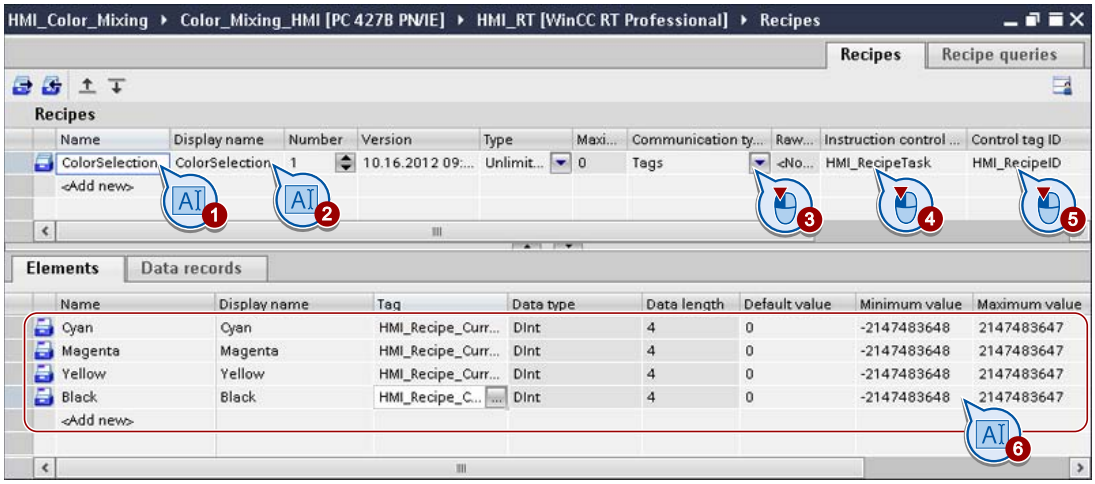
The following tags are created for synchronization with the controller:

Meaning	Name	Controller connection	Data type	Acquisition cycle
Number of the recipe data record	HMI_Recipe_ID	Yes	Int	1 s
Job/return value	HMI_RecipeTask	Yes	Int	1 s

Procedure

1. Create a new recipe "ColorSelection".
   - Enter the name and the display name.

     The display name appears in runtime and can be translated into other languages.
   - Define the "Tags" communication type and select the "HMI_RecipeTask" tag for the "Instruction control tag" and the "HMI_RecipeID" tag for the "Control tag ID".
2. Create the recipe elements "Cyan", "Magenta", "Yellow" and "Black".



Result

You have created the recipe and the recipe elements.

4.2.2 Creating recipe data records

Introduction

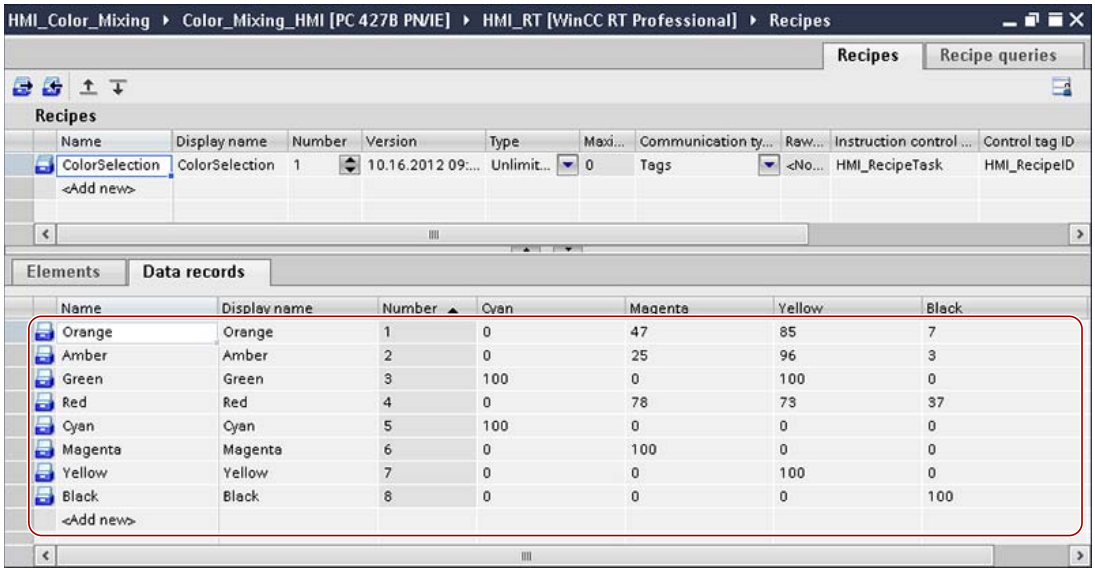
You need to set up a different ratio of basic colors for the "Orange", "Amber", "Green" and "Red" mixed colors. Save these mixing ratios to the recipe data records.

Requirements

You have created the recipe elements.

Procedure

1. Activate the "Data records" tab, create the recipe data records, and enter the basic color quantities.



Note

You may also record the mixing ratios in an external program and import them into the HMI device as CSV file.

For more information, refer to the WinCC online help in the TIA Portal.

Result

You have successfully created the recipe data records

4.3 Configuring a HMI screen for recipe selection

4.3.1 HMI screen for recipe selection

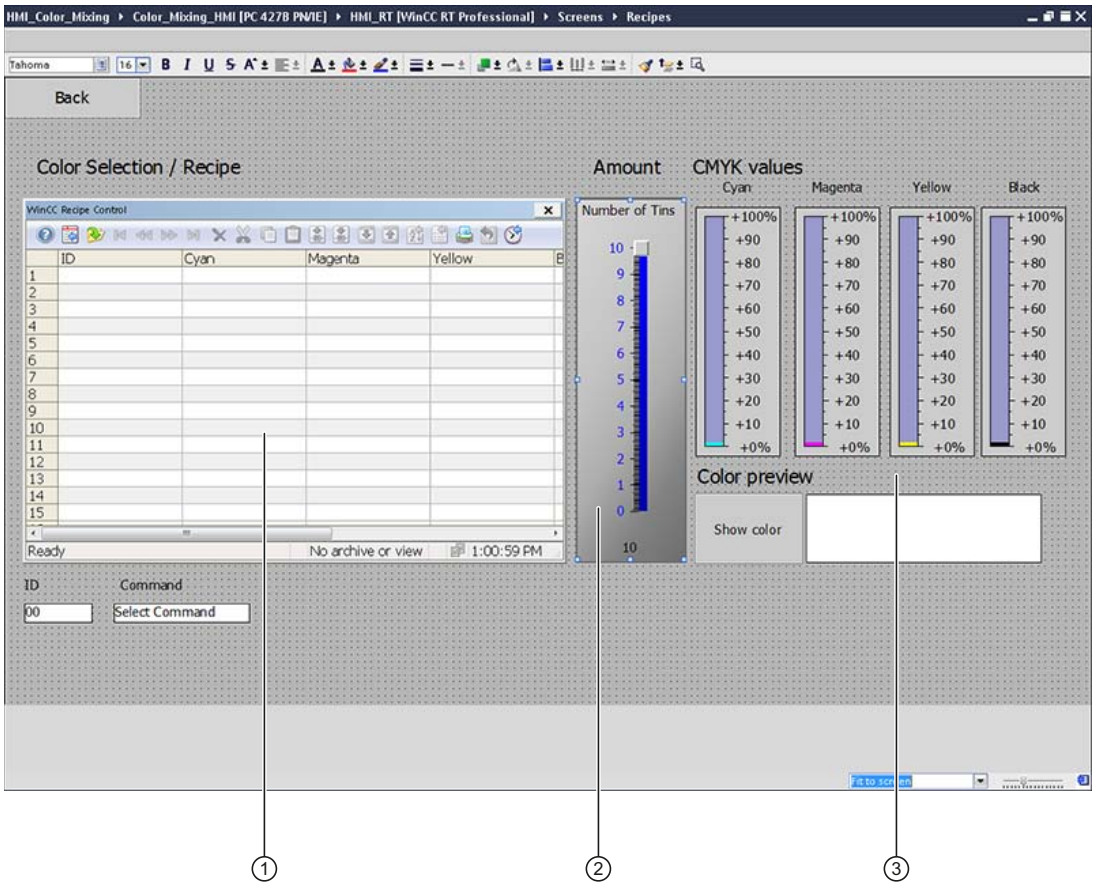
The following actions should be possible on the HMI device:

- Create recipe data record
- Save recipe data record
- Delete recipe data record
- Read recipe data record from color mixing system
- Transfer recipe data record to color mixing system

All these actions are possible with a recipe display. In addition, the operator needs to be able to input the number of tins. The required portions of the basic colors and the resulting mixed color should be visualized on the HMI device.

Layout of the "Recipes" screen

The "Recipes" screen comprises the following parts:



① Recipe view and selection fields to select recipes and recipe data records
② Slider to specify the number of tins
③ Visualization of the resulting color portions and representation of the selected color

4.3.2 Configuring a recipe view

Introduction

Configure a recipe view and suitable input/output fields in the screen to display, select and edit recipe data records and forward these to the controller. The recipe view shows the data in tabular form. Each row contains a recipe data record.

The "ID" input field indicates the number of the recipe data record; in the "Command" input/output field, you select the command to edit the recipe data record and to obtain information from the controller about the execution.

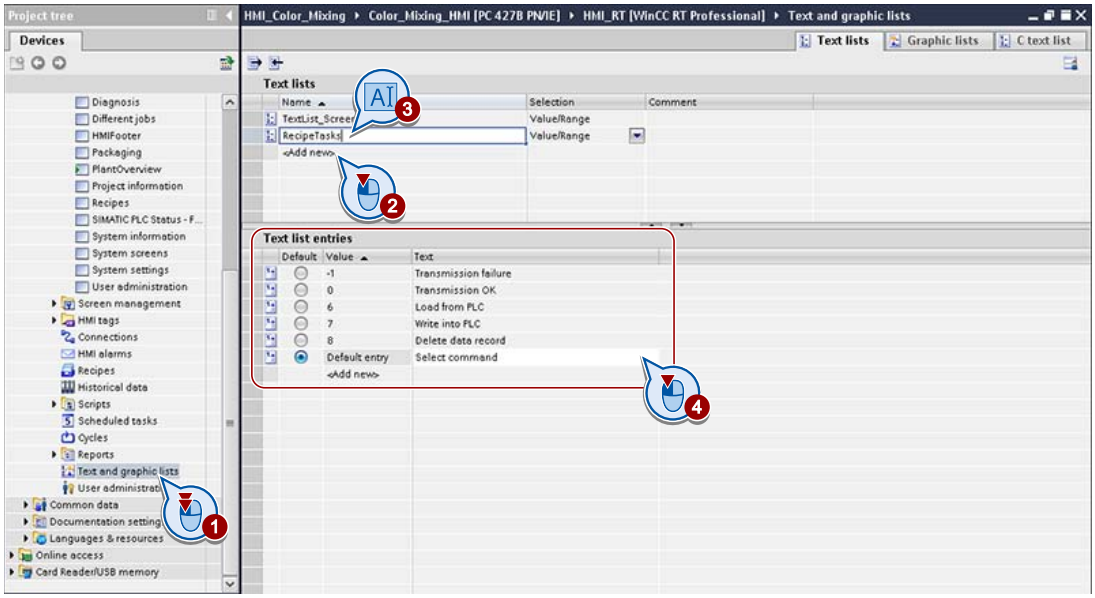
Requirements

The following tags are created:

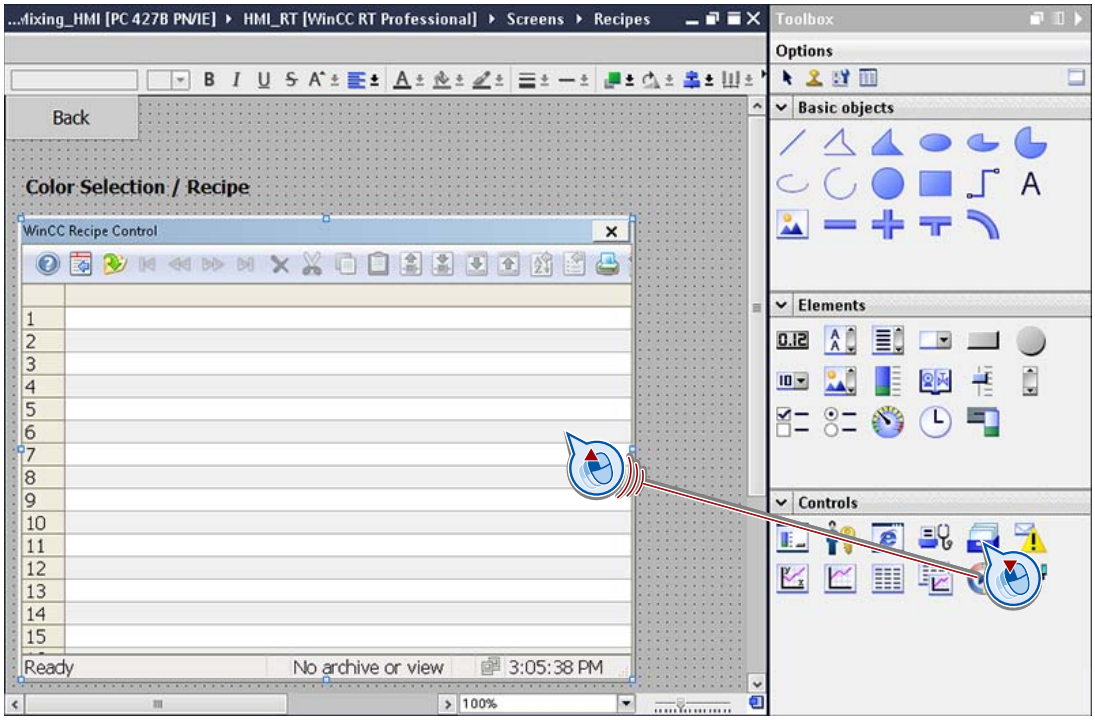
Meaning	Name	Controller connection	Data type	Acquisition cycle
Number of the recipe data record	HMI_Recipe_ID	Yes	Int	1 s
Job/return value	HMI_RecipeTask	Yes	Int	1 s

Procedure

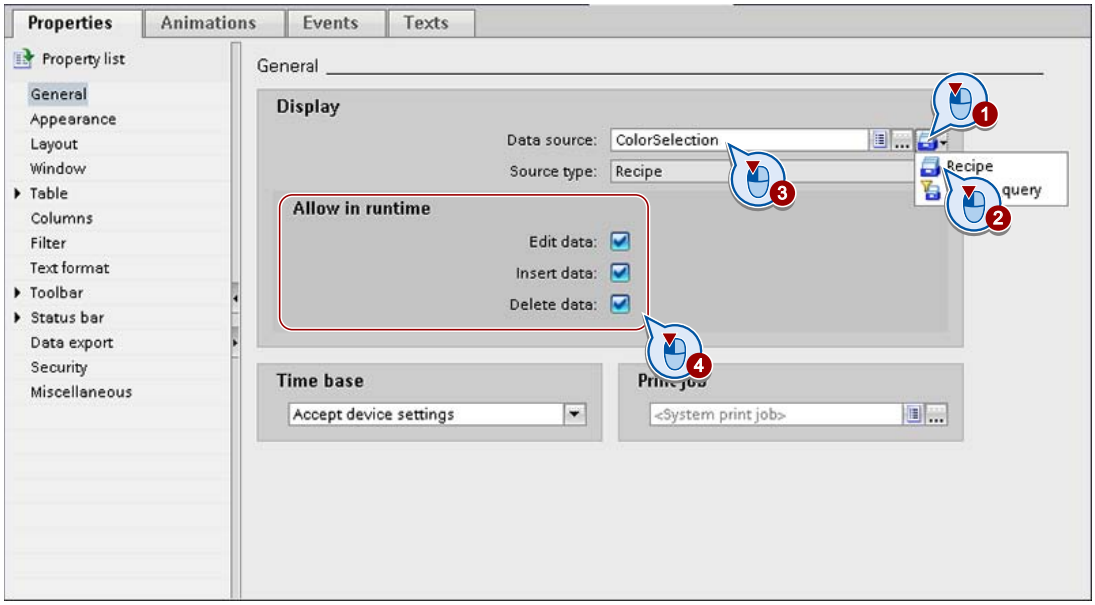
1. Configure a "RecipeTasks" text list with entries for transferring and editing the recipe data records.



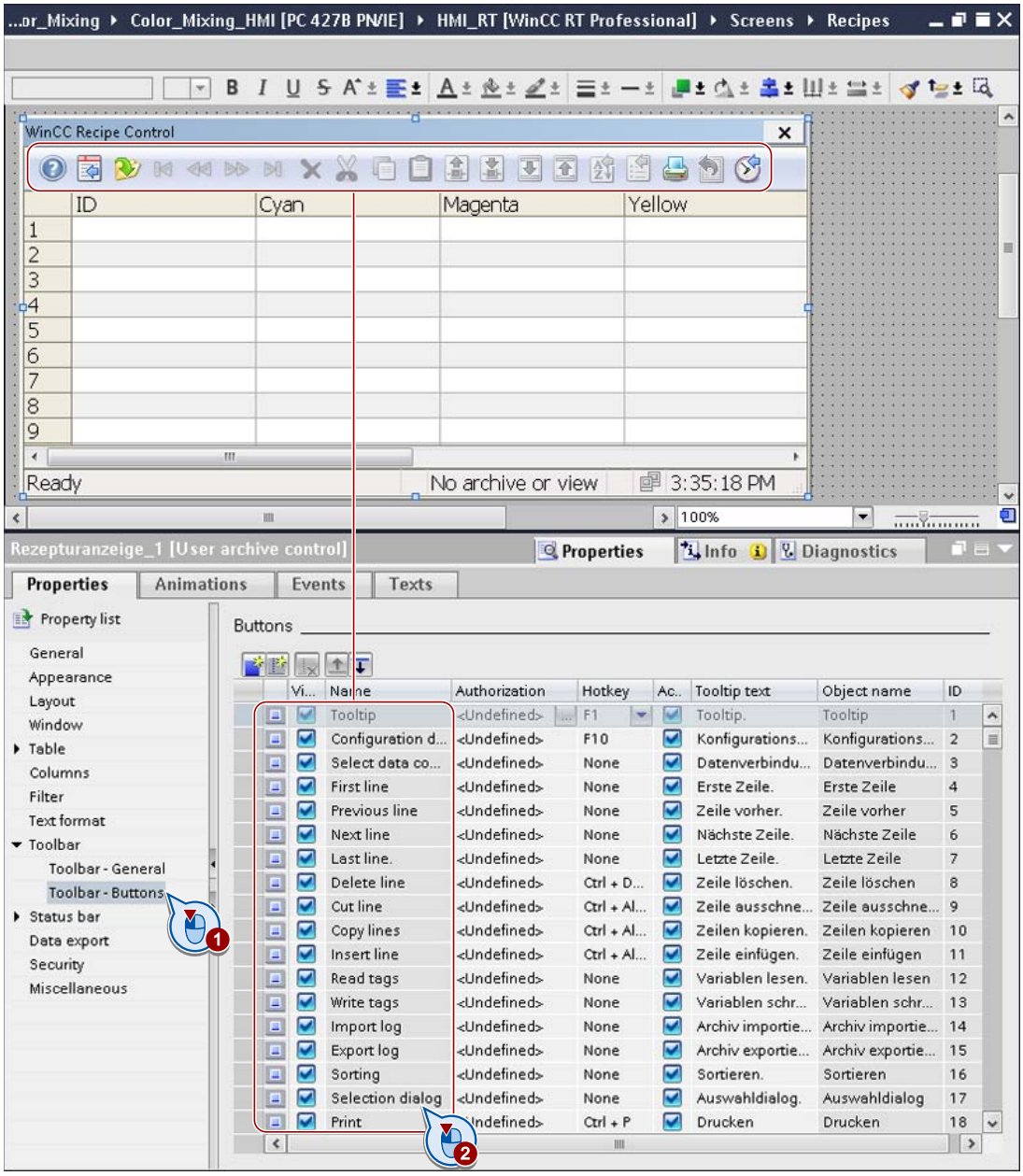
2. Open the "Recipes" screen and insert the recipe view and a header in the work area.



3. Select the data source in the inspector window and specify that recipe data records may be edited, created and deleted in runtime.

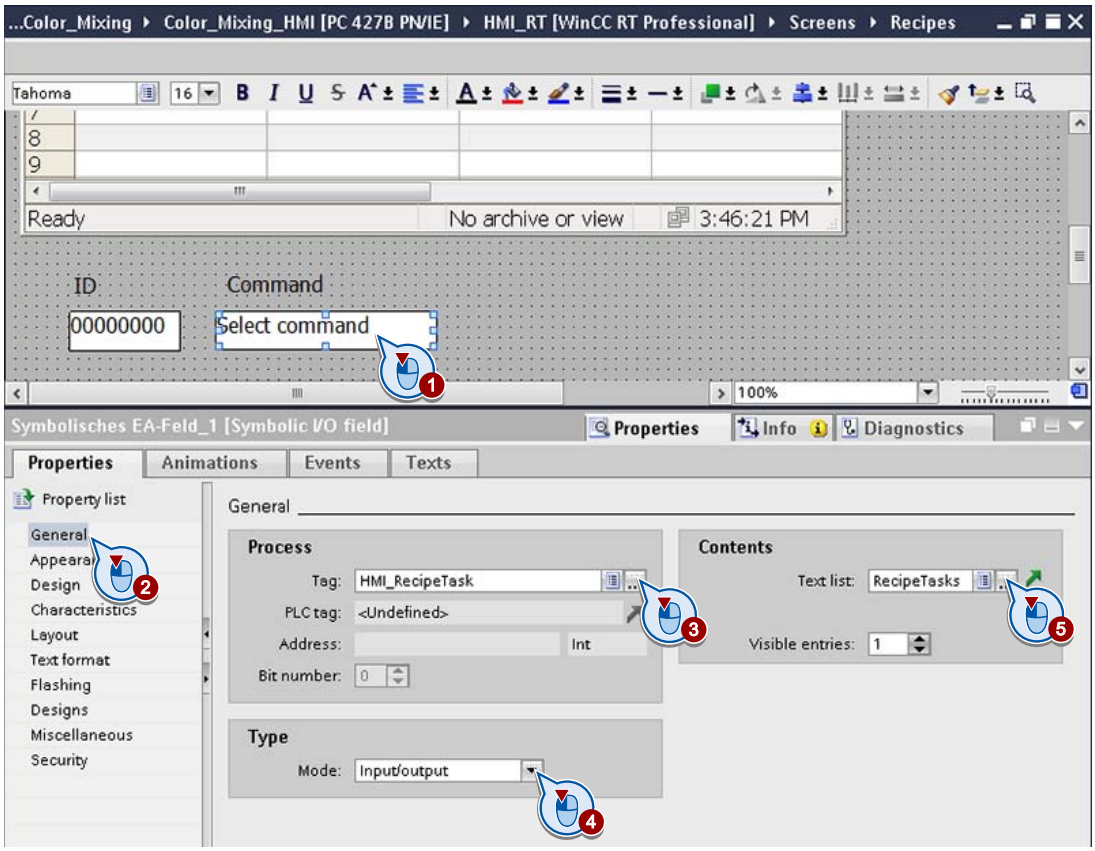


4. Adapt the recipe view toolbar to the requirements of your project. All commands are activated by default.

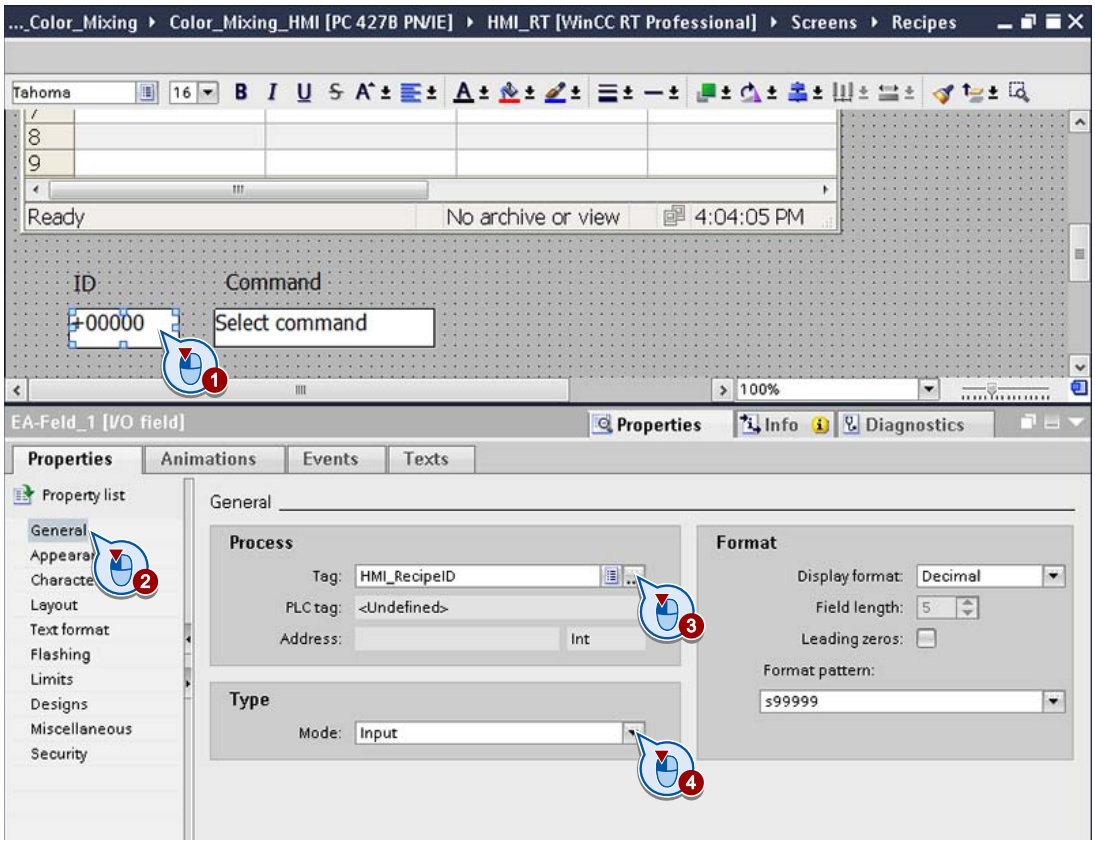


5. Insert an I/O field and a symbolic I/O field below the recipe view and label these with "ID" and "Command".

6. Assign the "HMI_RecipeTask" tag, "Input/Output" mode and the "RecipeTasks" text list to the symbolic I/O field.



7. Assign the "HMI_RecipeID" tag and "Input" mode to the I/O field.



Result

A recipe view and the fields required for editing and transferring data records are configured in the "Recipes" screen.

4.3.3 Configuring sliders

Introduction

Operators are to use a slider to select how many tins of a certain color will be produced.

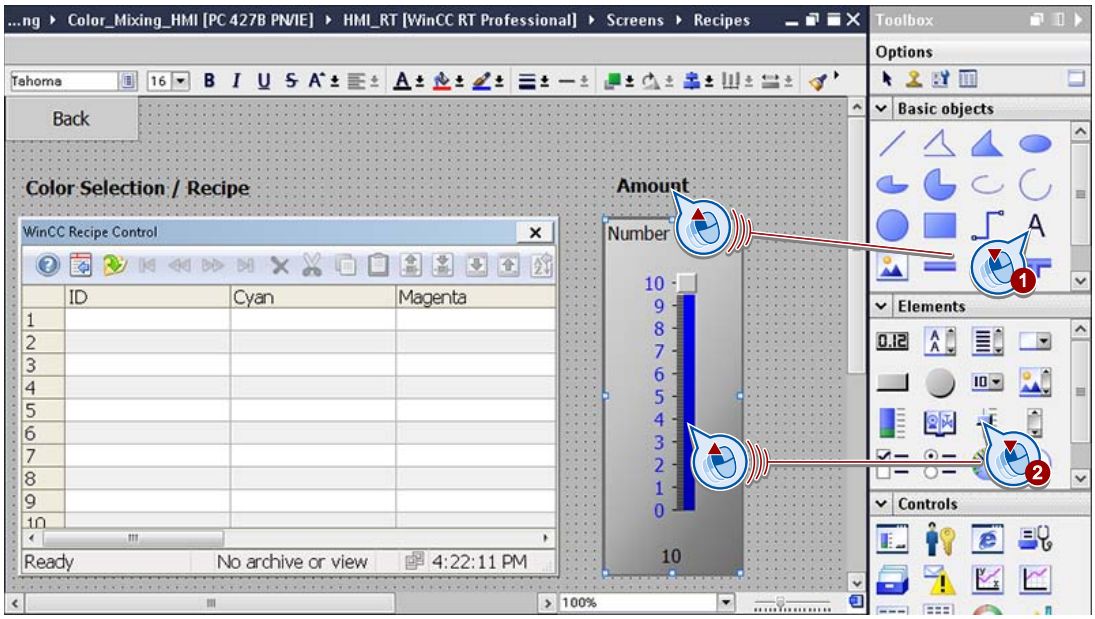
Tags

You need the following tags for communication between the HMI device and the controller:

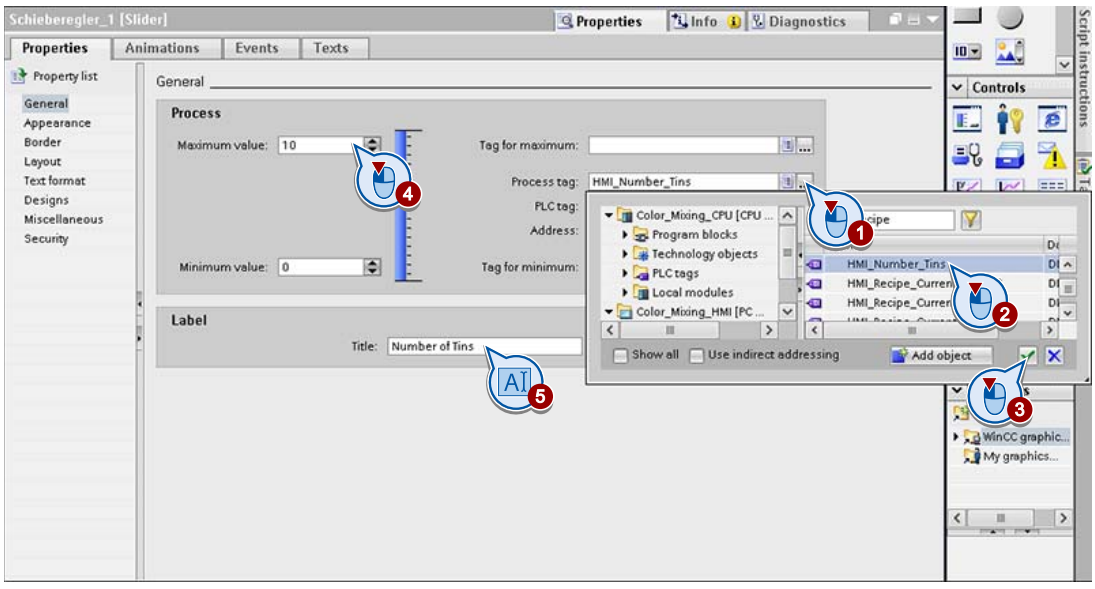
Meaning	Name	Controller connection	Data type	Acquisition cycle
Number of tins	HMI_Number_Tins	Yes	DInt	1 s

Procedure

1. Insert a header and a slider in the screen.



2. In the inspector window, configure the "Number_Tins" tag at the slider, and format and label the slider as shown.



Result

You have successfully configured the slider to select the number of tins to be produced.

4.3.4 Visualizing recipe data record and color

Introduction

The mixing ratio of the basic colors is visualized in the "Recipes" screen for the selected recipe data record. The bars show the basic color portions.

The operator can also show the resulting mixed color on the HMI device.

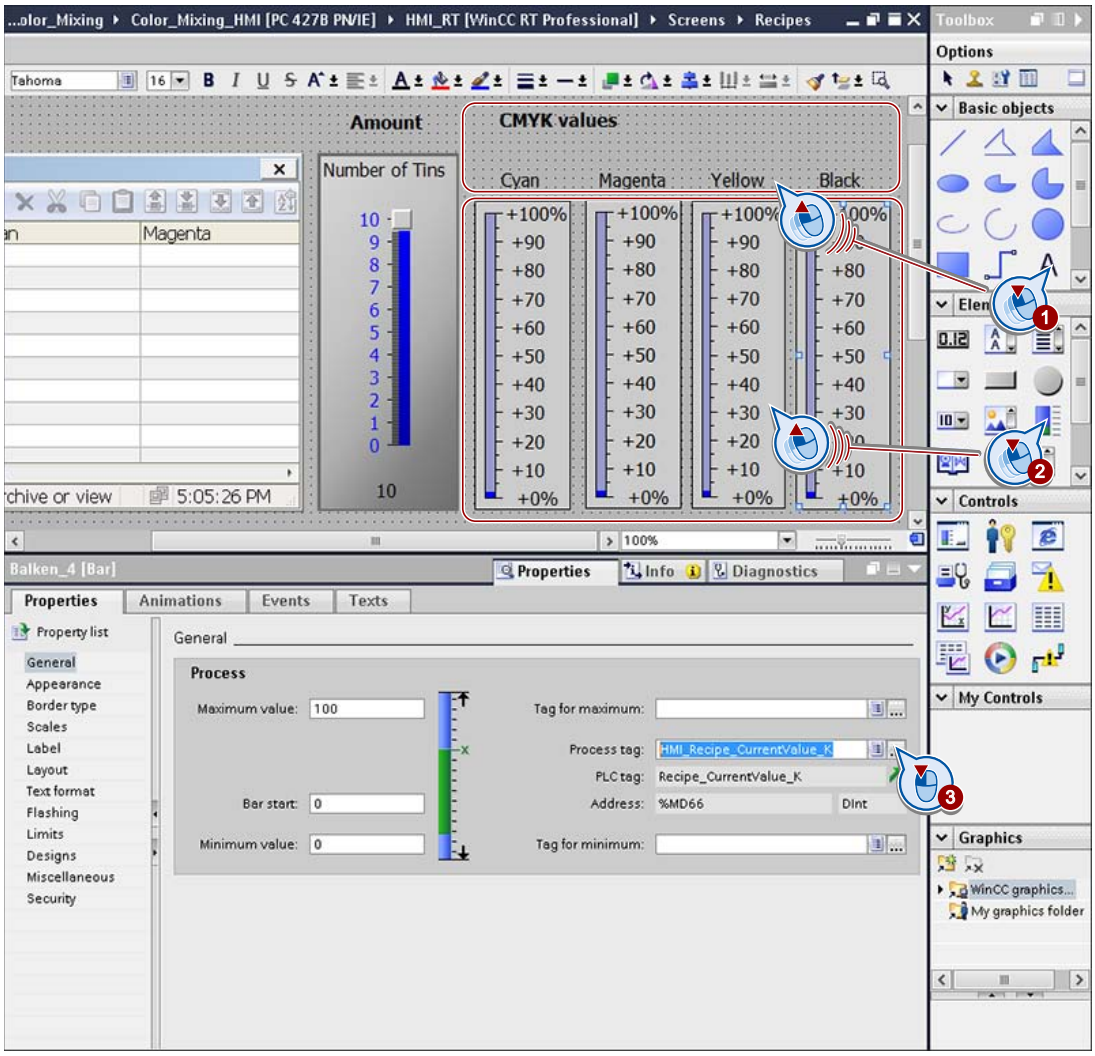
Tags

You need the following tags for communication between the HMI device and the controller:

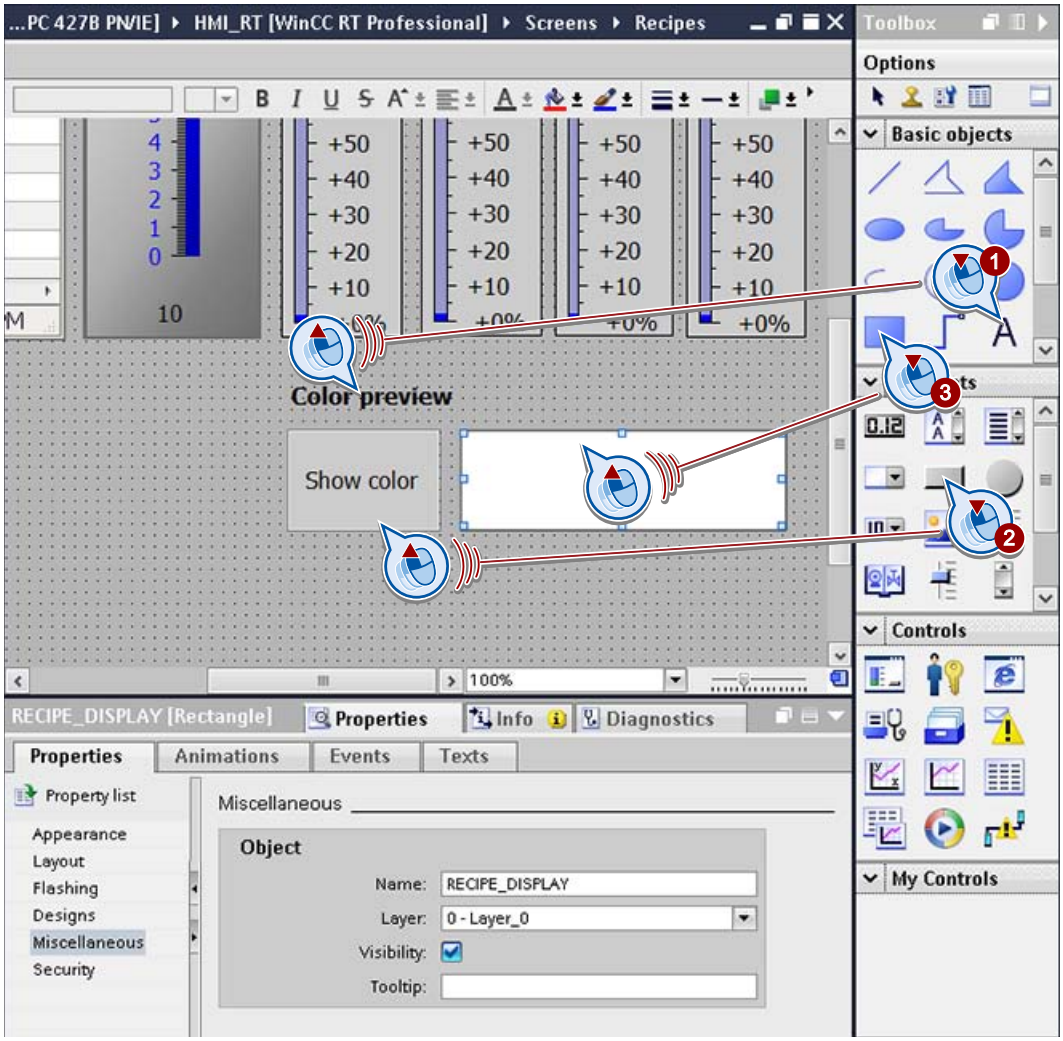
Meaning	Name	Controller connection	Data type	Acquisition cycle
"Cyan" color portion	HMI_Recipe_CurrentValue_C	Yes	DInt	500 ms
"Magenta" color portion	HMI_Recipe_CurrentValue_M	Yes	DInt	1 s
"Yellow" color portion	HMI_Recipe_CurrentValue_Y	Yes	DInt	1 s
"Black" color portion	HMI_Recipe_CurrentValue_K	Yes	DInt	1 s

Procedure

1. Insert a header.
2. Insert a bar and a header for each basic color and configure one of the above-named tags, for example HMI_Recipe_CurrentValue_K" for the black color portion, on each bar.



3. To preview the mixed color, add a button and a rectangle with the name "RECIPE_DISPLAY" to the screen.



Note

RGB representation of the colors is required for visualization on the display. In the "Creating user-defined functions" chapter, you will learn how to configure a script at the button that converts CMYK values into RGB values and colors the "RECIPE_DISPLAY" rectangle accordingly.

Result

You have configured the "Recipes" screen with all required operating elements.

4.4 Recipe with product data

Introduction

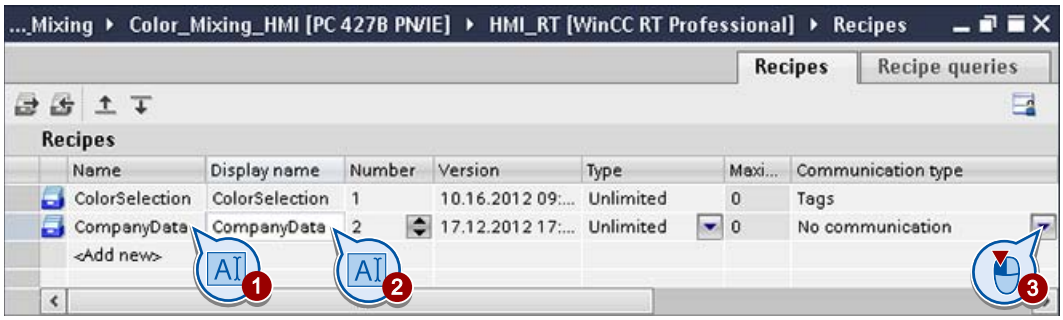
An additional recipe with product data such as manufacturer, filling date and best-before date is to be added to the existing "ColorSelection" recipe.

Note

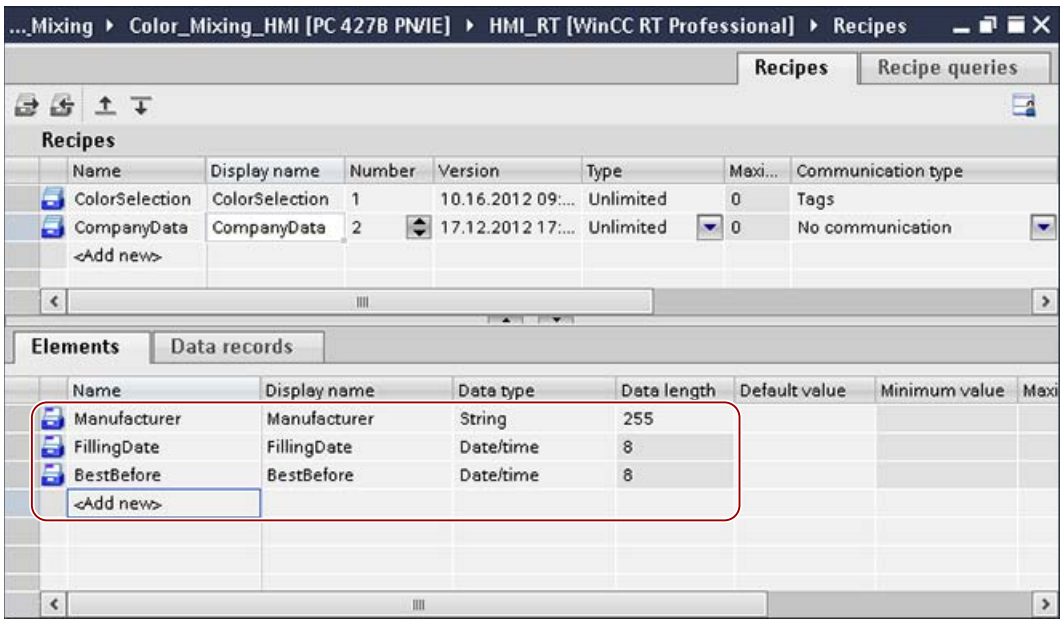
This product data is needed only for label printing and not for the production. Therefore, it does not need to be transferred to the controller. Consequently, no communication with the controller is required.

Procedure

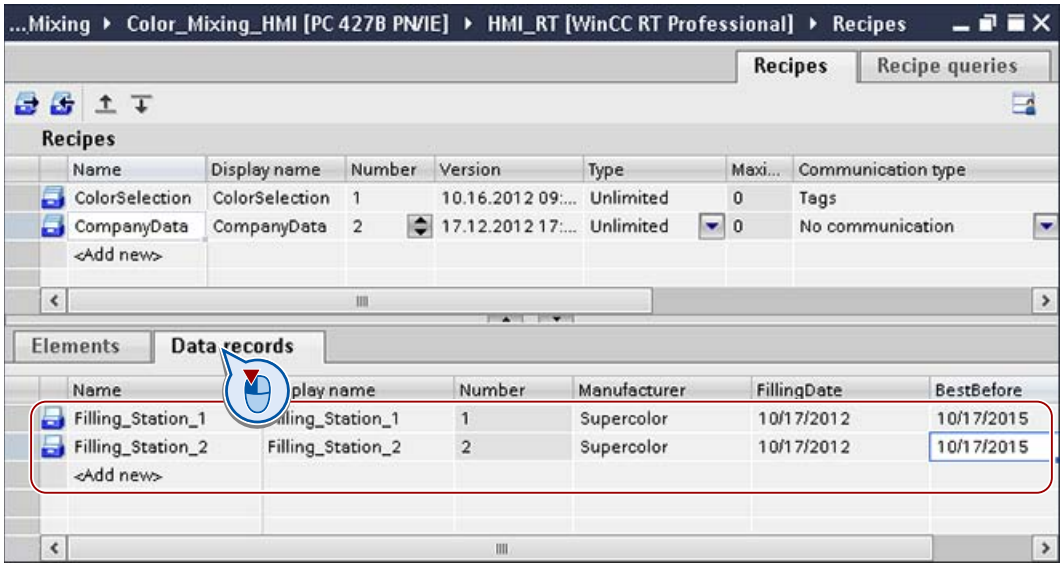
1. Open the "Recipes" editor and create the "CompanyData" recipe.



2. Create the following recipe elements: "Manufacturer", "FillingDate" and "BestBefore".



3. Create two data records for the two fillers.



Result

You have created a recipe with product information for label printing.

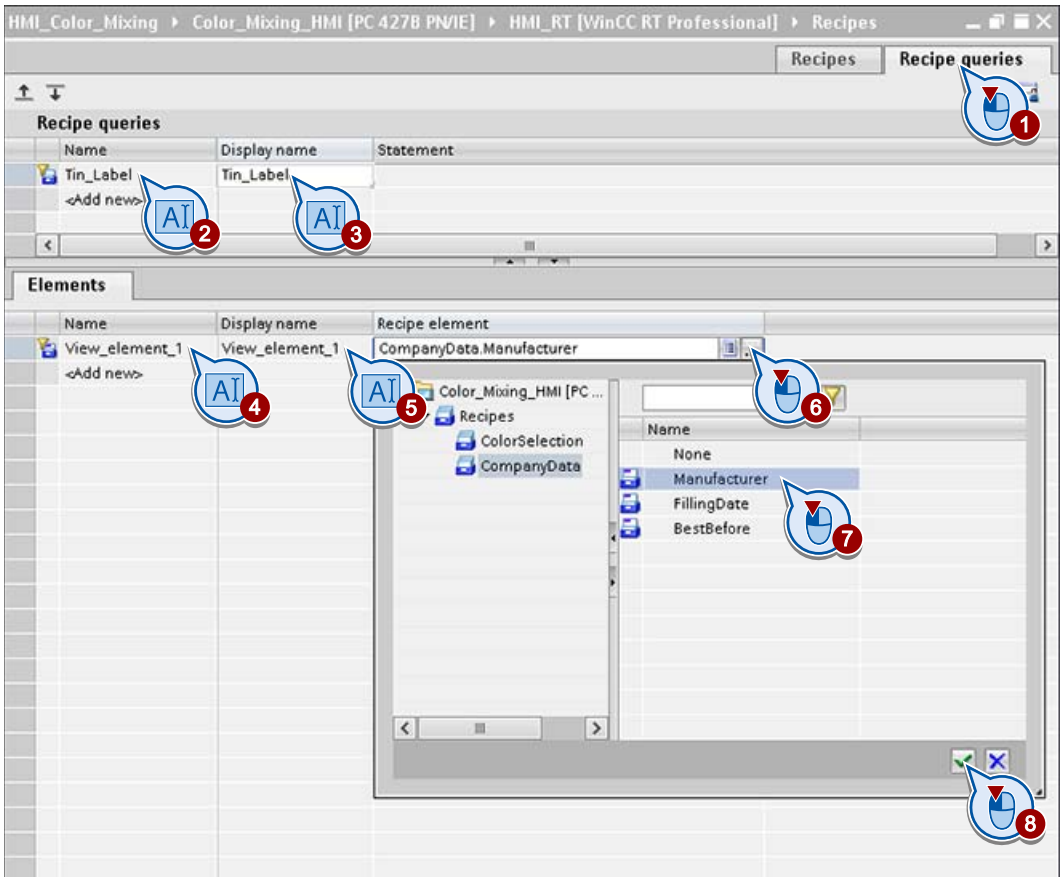
4.5 Recipe query for label printing

Introduction

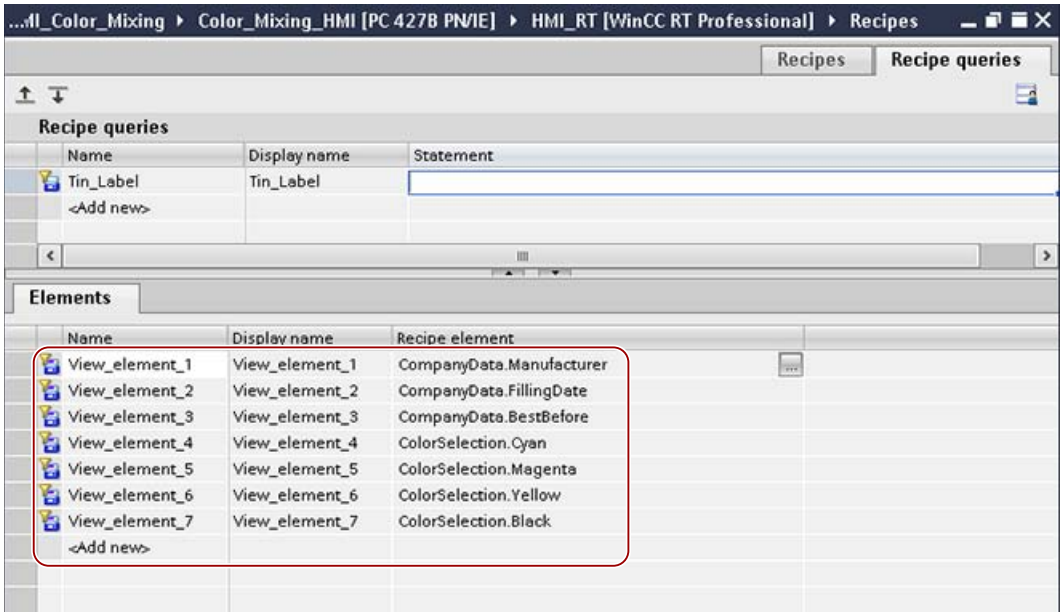
You can use recipe queries to combine separate items of data from different recipes with one another. This allows you to combine them on information that will fit on a label.

Procedure

1. For label printing, create a new "Tin_Label" recipe query and the first element of the query.



2. Create the additional elements of the query and assign the corresponding recipe elements to them.



Result

You have created the "Tin_Label" recipe query.

Testing

The configuration should be tested in runtime following each extensive editing step. This way, inconsistencies and display errors can be detected and removed in due time.

A detailed description is available in the section "Testing and loading a project" of this Getting Started.

Configuring alarms

5

5.1 Introduction

Alarms display events or operating states that occur or predominate in the color mixing station. You can use alarms for diagnostic purposes to correct errors, for example.

Types of alarm

WinCC distinguishes between the following alarm types:

User-defined alarms

- **Discrete alarms** are triggered by the controller and indicate status changes in the plant. They indicate whether a valve is open or closed, for example.
- **Analog alarms** indicate limit violations. An analog alarm is triggered, for example, when the speed of a motor drops below a certain value.
- **User alarms** indicate operator actions on the HMI device in Runtime Professional. An alarm is automatically triggered by the operator action.
- **Controller alarms** indicate controller states. They are configured using STEP 7 in the controller and can be edited in WinCC.

System-defined alarms

- **System-defined controller alarms** provide information on the internal states of the controller.
- **System events** on the HMI device provide information about internal states of the HMI device. You cannot configure system events.

Alarm classes

Each alarm belongs to an alarm class. The alarm class defines how the alarm is represented in runtime, as well as the acknowledgment concept.

5.2 Configuring analog alarms

Introduction

The storage tanks of the color mixing station empty at different speeds depending on the recipe used. However, the fill level must not fall below a specific value. As soon as this value is reached, the tank should be refilled.

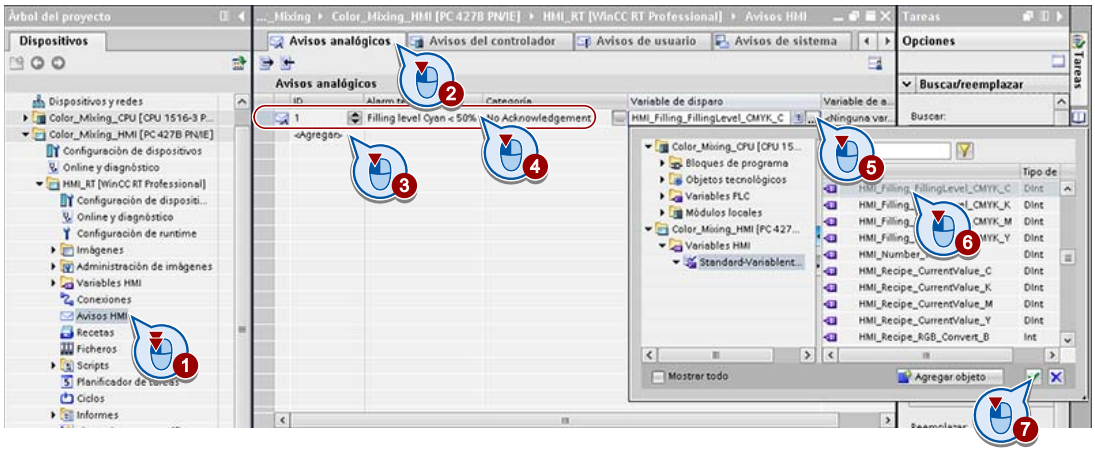
To ensure that the operator is informed in good time, configure alarms for the violation of different fill levels in the tanks. The fill levels of the tanks are stored in the tags "HMI_Filling_FillingLevel_CMYK_C", "HMI_Filling_FillingLevel_CMYK_M", "HMI_Filling_FillingLevel_CMYK_Y" and "HMI_Filling_FillingLevel_CMYK_K" with the "Cyclic continuous" acquisition mode.

In the "HMI alarms" editor, configure the alarms for fill level low limit violations of 50% and 25% as alarms without mandatory acknowledgment. Configure the alarms for low limit violations of the fill level of 10% as mandatory acknowledgment alarms. You can also configure alarms directly at the tag in the "Tags" editor.

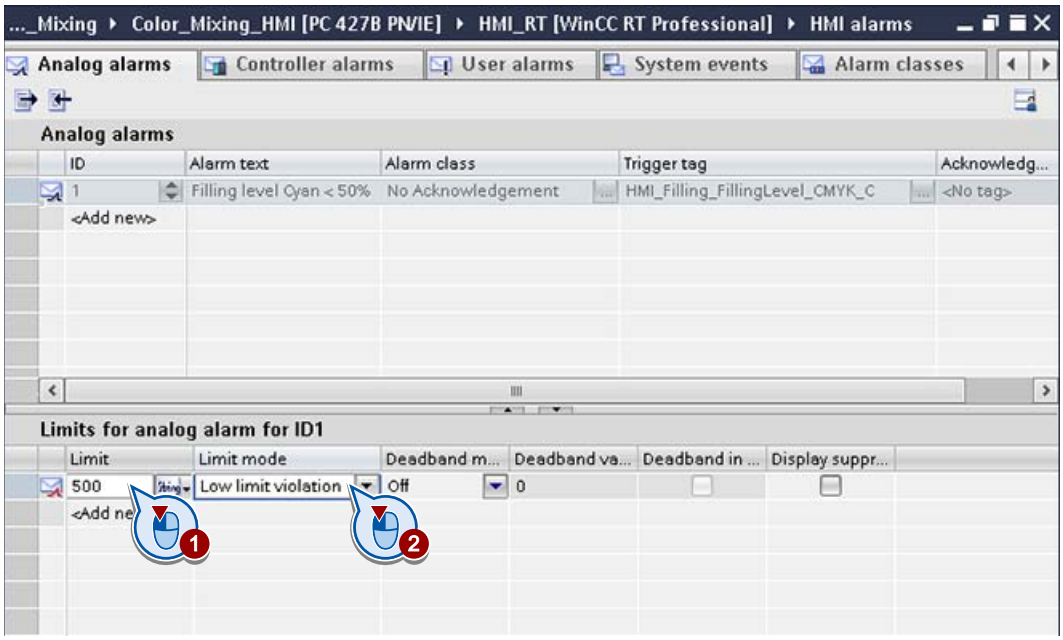
To do this, you use the already created "No Acknowledgement" and "Acknowledgement" alarm classes.

Procedure

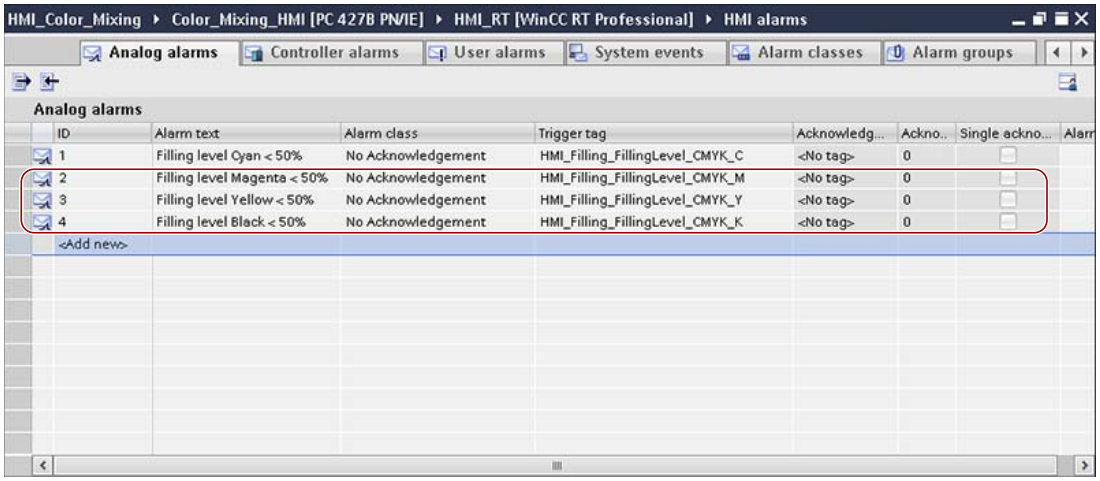
1. Create a new analog alarm. Specify the alarm text, the alarm class "No Acknowledgement" and the tag "HMI_Filling_FillingLevel_CMYK_C".



2. Assign a limit of 500 and the "Low limit violation" mode to the alarm.



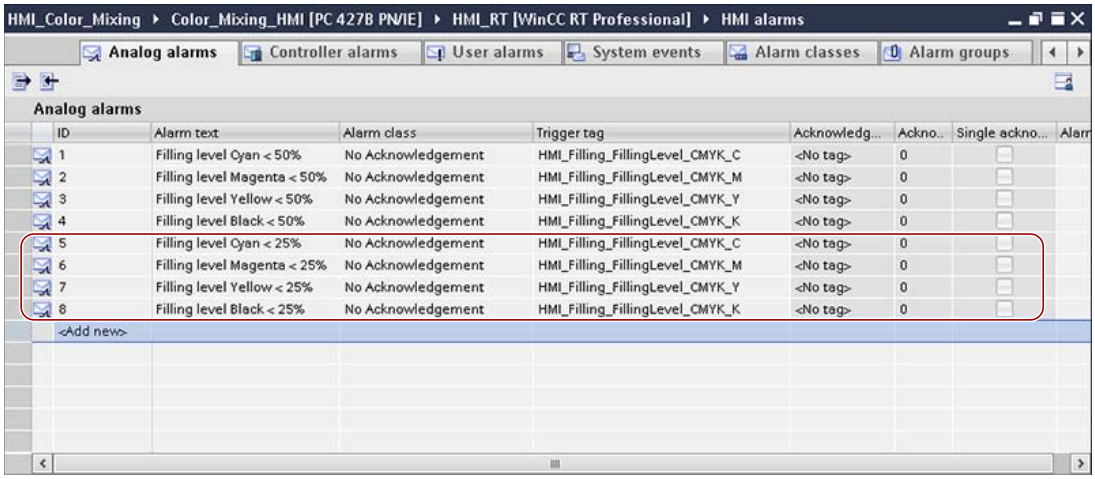
3. Likewise, create alarms for the other storage tanks.



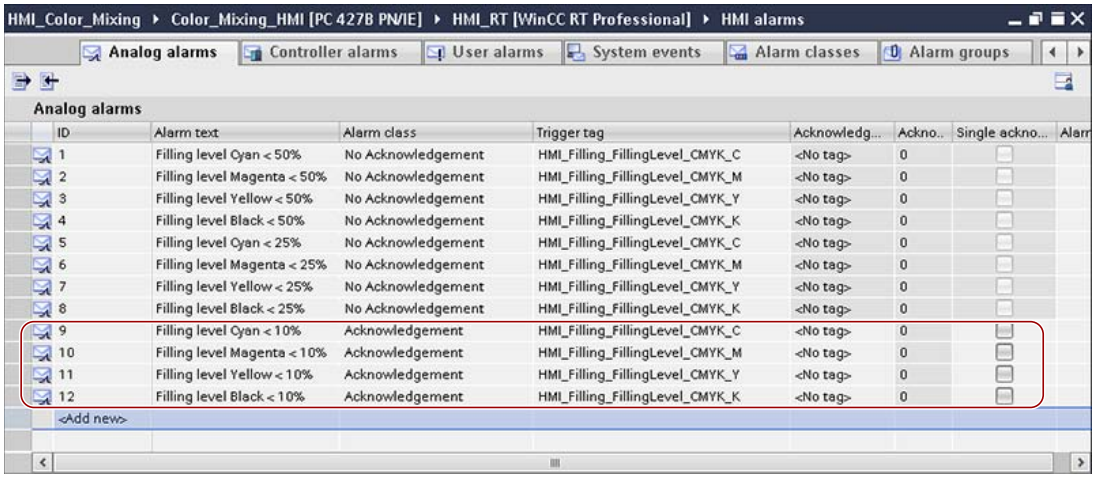
Note

When you create the second analog alarm by double-clicking the empty row, the settings are taken over from the first analog alarm. The alarm number is incremented automatically.

4. In the same way, create the alarms for low limit violations of the fill level of 25%.



5. In the same way, create the alarms for low limit violations of the fill level of 10% with the "Acknowledgement" alarm class.



Logging alarms and process values

6

6.1 Introduction

Logging basics

Reporting of plant production is realized by saving the alarms and process values generated during production to log files. These logs enable the supervisor to call information related to exceptional events occurred in the course of the last shift directly on the HMI device.

You can then analyze the alarm and process data backup at your own time. This analysis provides you with operational and technical data for assessing the operational state of the plant. You can use this information to:

- Optimize wait state cycles
- Safeguard quality standards
- Enhance product quality

Tasks

This chapter shows you how to log and display alarms and process values. Use the alarm view to display logged alarms on the HMI device; use the trend view to display the logged process values.

6.2 Logging in Runtime Professional

In WinCC RT Professional, you determine both the physical storage and the logical structural of the data to be logged.

Physical log types

The following physical log types are available in WinCC RT Professional for the logging of process data:

- "Fast data log" for saving all values whose logging cycle is <= 1 minute.
- "Slow data log" for saving all values whose logging cycle is > 1 minute .
- "Alarm log" for saving alarms.

You can configure the basic properties of physical logs, such as the segment size of the data log, in the "Runtime settings > Logging" dialog.

Logical log types

You specify the content of the log, and thus its logical structure, when you set up the log. You can save tags in the data logs and in the compressed logs. Compressed logs provide you with consolidated data and a better overview. Sum, mean value, maximum and minimum are available as mathematic functions.

You configure the logs in the following program areas in WinCC RT Professional:

- You configure logging of alarms in the "HMI alarms" editor in the "Alarm classes" tab.
- You configure the logging of tags in the "Logs" editor.

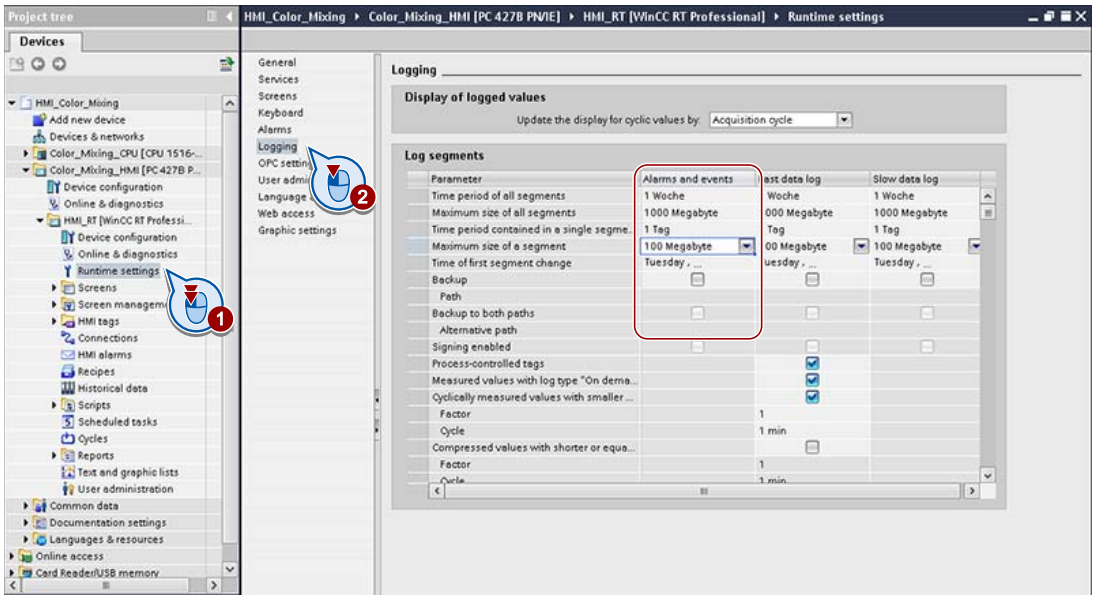
6.3 Logging alarms

Introduction

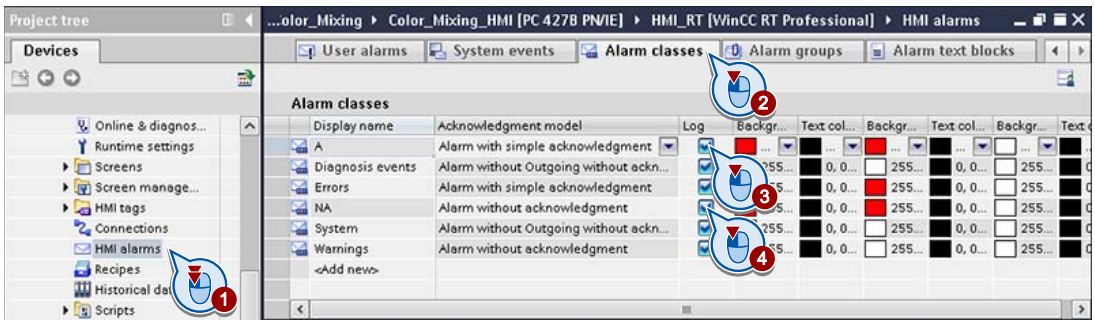
The movement of the color reservoir fill levels in the color mixing system is to be documented. To do this, specify the settings for the logs and activate the logging of the fill level alarms.

Procedure

1. Select "Runtime settings > Logging" and set the basic properties of the alarm and event logs.



2. Open the "HMI alarms" editor. By default, all alarm classes are logged. If necessary, activate logging of the alarm classes "Acknowledgment" and "No Acknowledgment".



Result

You have configured the logging of alarms for low limit violations of the fill level.

6.4 Visualizing logged alarms

Introduction

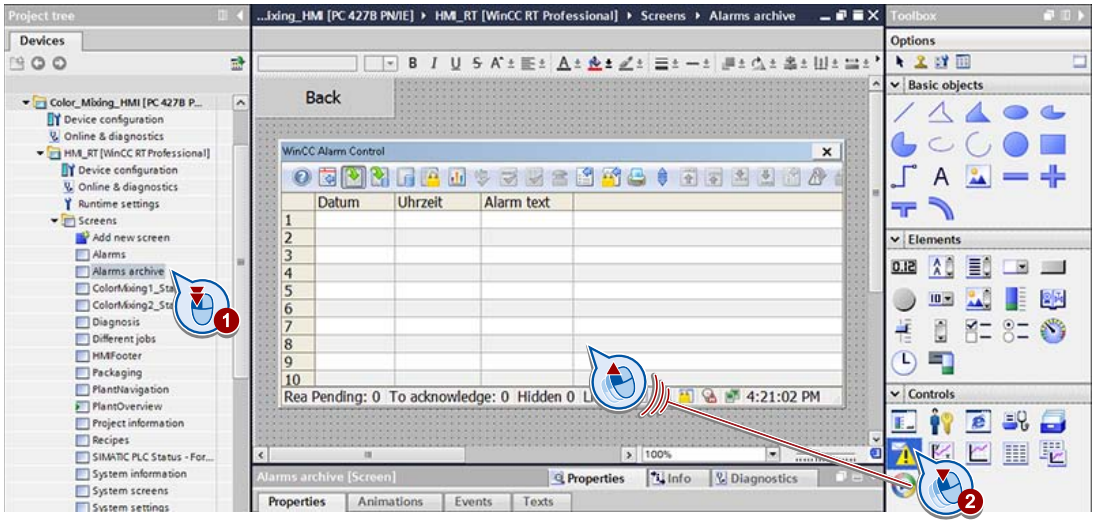
You are now going to learn how to configure an alarm view to display fill level alarms.

Requirement

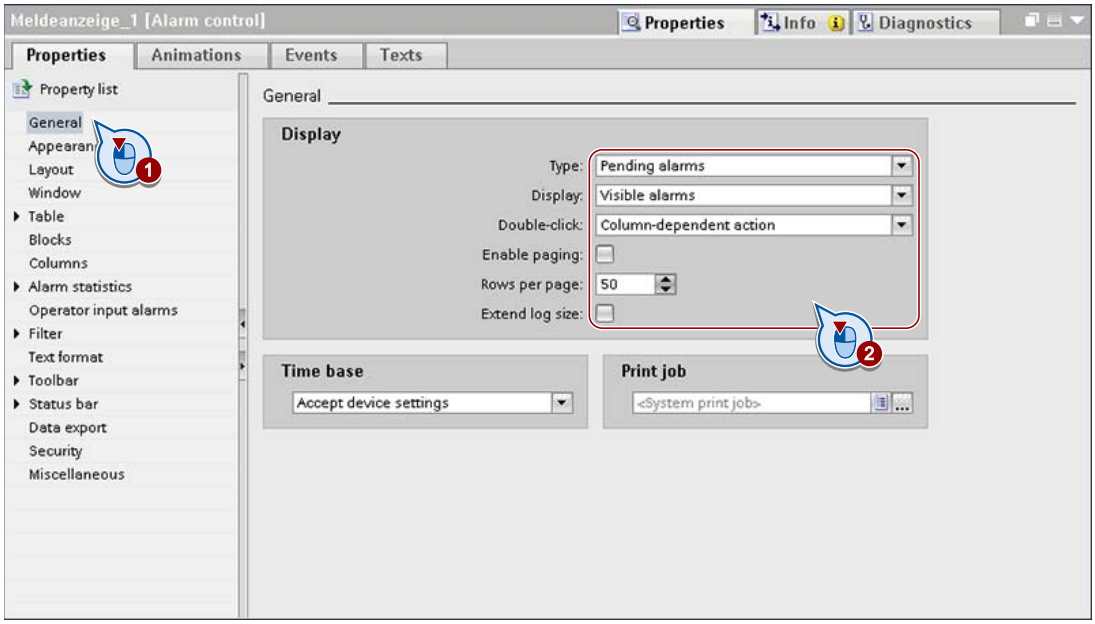
Logging of alarms of the alarm classes "No Acknowledgment" and "Acknowledgment" has been configured.

Procedure

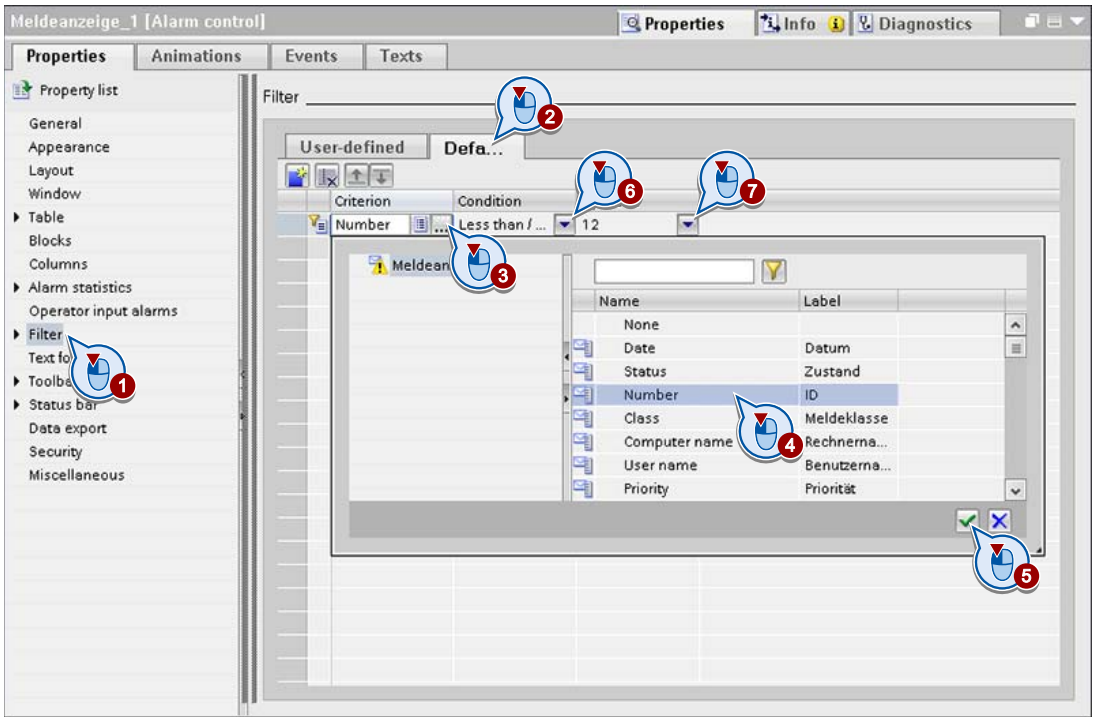
1. Open the "Alarms archive" screen and insert an alarm view.



2. Configure the alarm view to display the pending alarms.



3. Use a filter to limit the display of the alarms to the fill level alarms with the numbers 1 to 12 (the numbers of the defined analog alarms).



Result

You have configured the alarm view for displaying the fill level alarms.

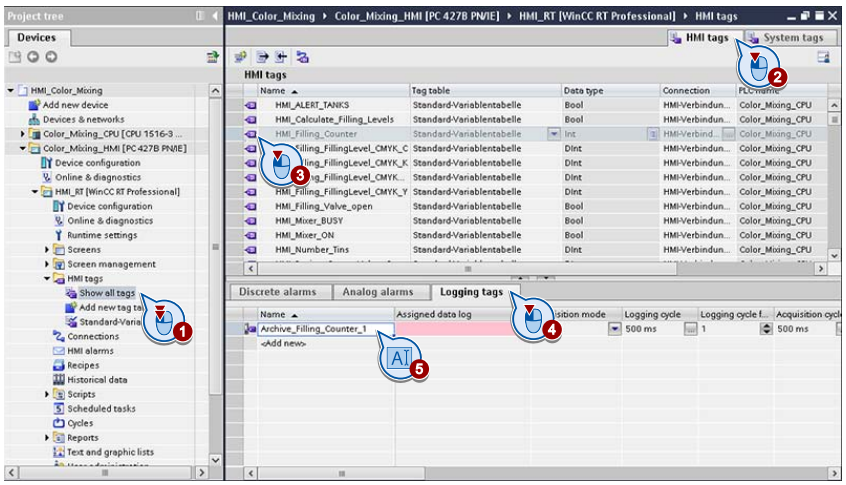
6.5 Logging process values

Introduction

The batch of filled tins produced in a shift is to be logged. The shift supervisor can thus obtain an overview of production output in the color mixing station directly on the HMI device.

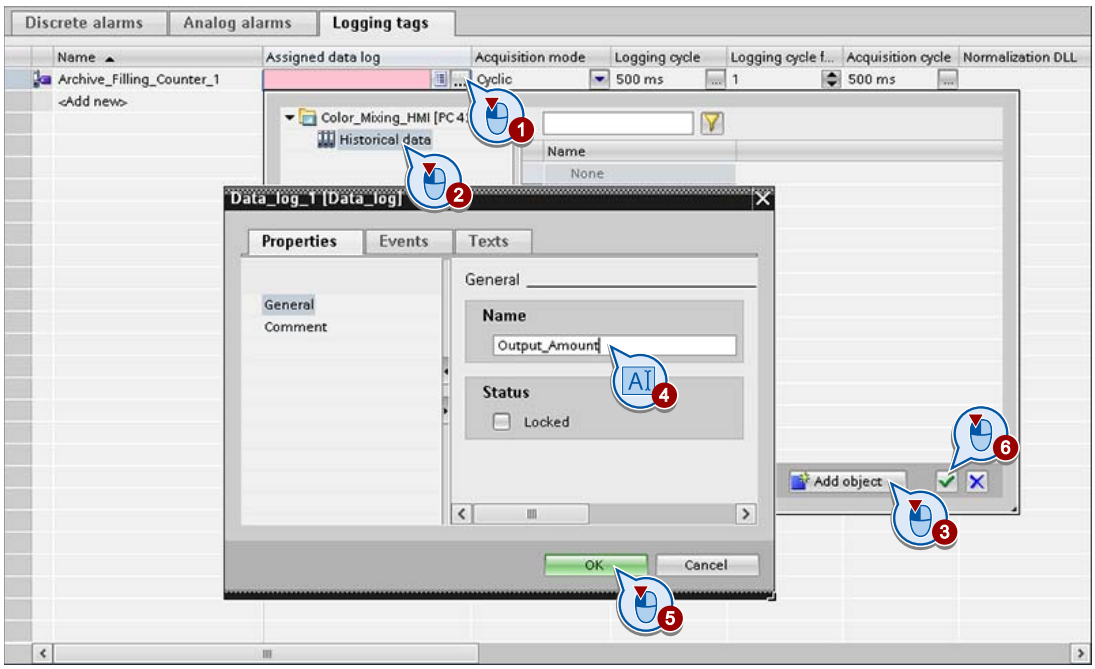
Procedure

1. Create a new logging tag for the "HMI_Filling_Counter" tag in the "HMI tags" editor.



You can also configure the logging tag in the "Logs" editor.

2. Configure the logging tag: Name it, create a new log, and assign this log to the logging tag.



Result

The output of the color mixing station is logged.

6.6 Visualizing process values

Introduction

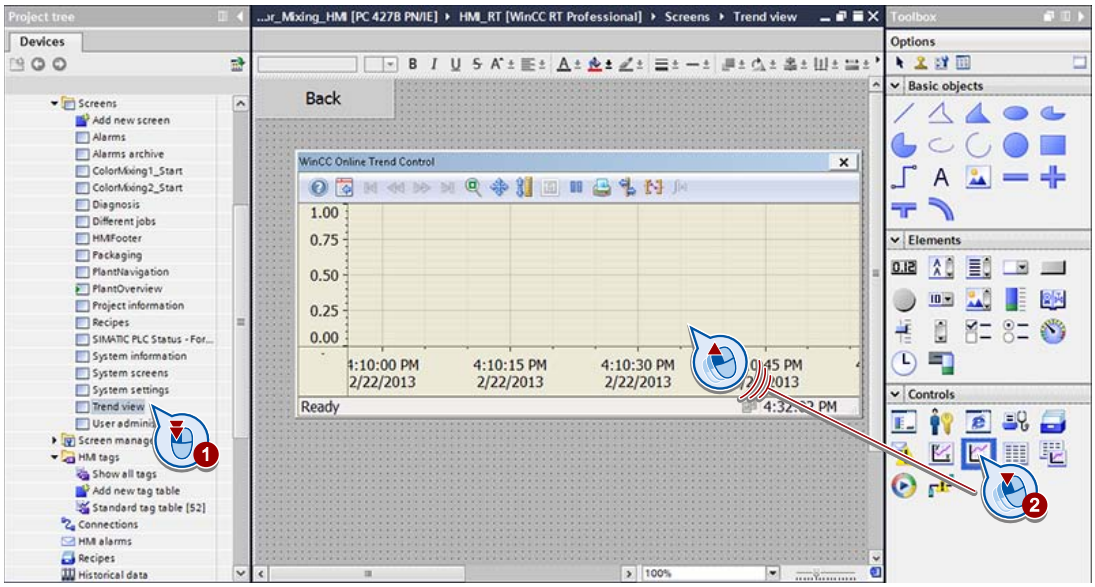
The number of filled tins is to be output as a trend on the HMI device.

Requirement

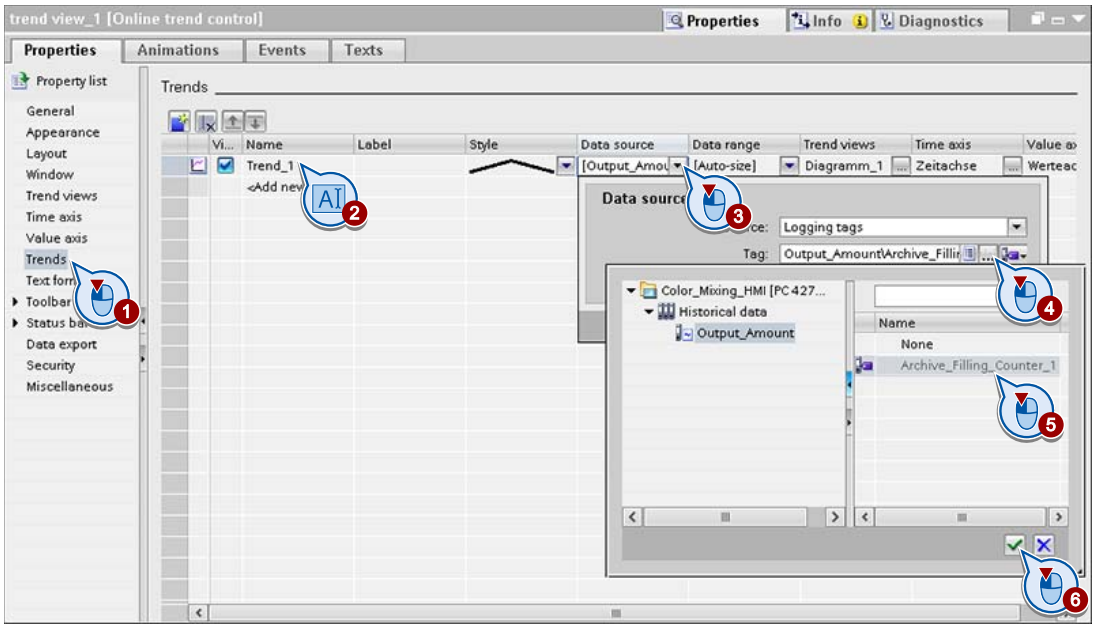
You have created the "Output_Amount" data log.

Procedure

1. Open the "Trend view" screen and insert an f(t) trend view.



2. Configure the trend view: Create a new trend, label it, and select the logging tag that will supply the trend with values.



Use the default value of 500 milliseconds for regular acquisition of the values.

Result

You have configured the output of the process values as a trend on the HMI device.

Configuring reports

7

7.1 Introduction

Reporting basics

Reports document the production process and are the basis for product testing and quality control. Tag values, alarms and recipe data can be logged in WinCC.

Introduction

You configure your reports in the "Reports" editor:

- Specify the formal layout of the report in the inspector window with document structure and page format.
- In the work area, specify which data must always be available for the report. To do this, insert the corresponding objects in the detail pages.
- With the print job, you then specify the output destination (printer or file), the printing time and the data scope of the report.

Tasks

In this chapter, you configure reports for alarms and recipes and define the print jobs.

You also configure the printing of labels triggered by the operator.

7.2 Reporting alarms

Introduction

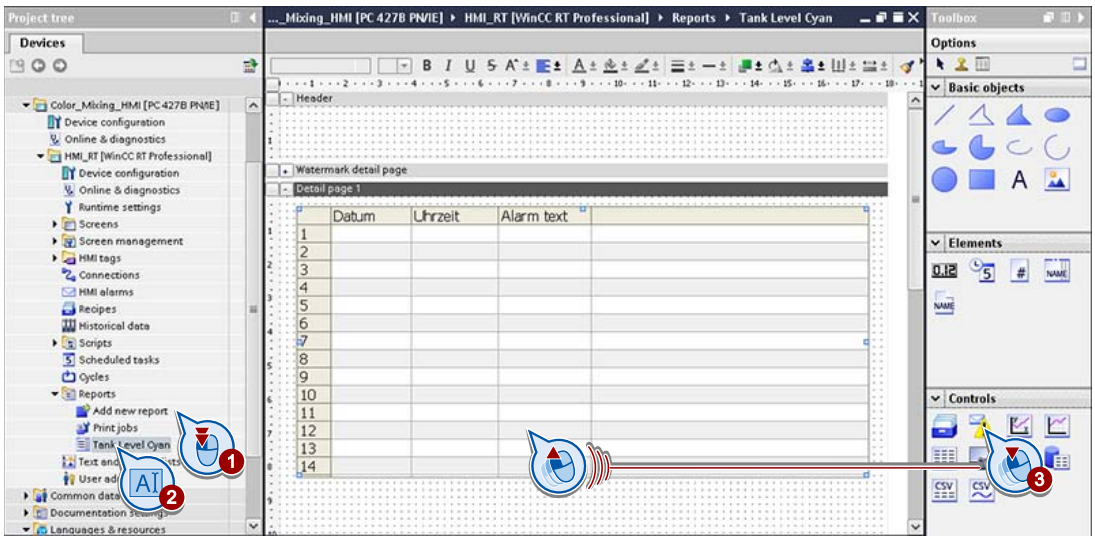
In the following chapter, you will create a report in which the alarms on low limit violations of the fill level of the "Cyan" tank will be output.

Requirements

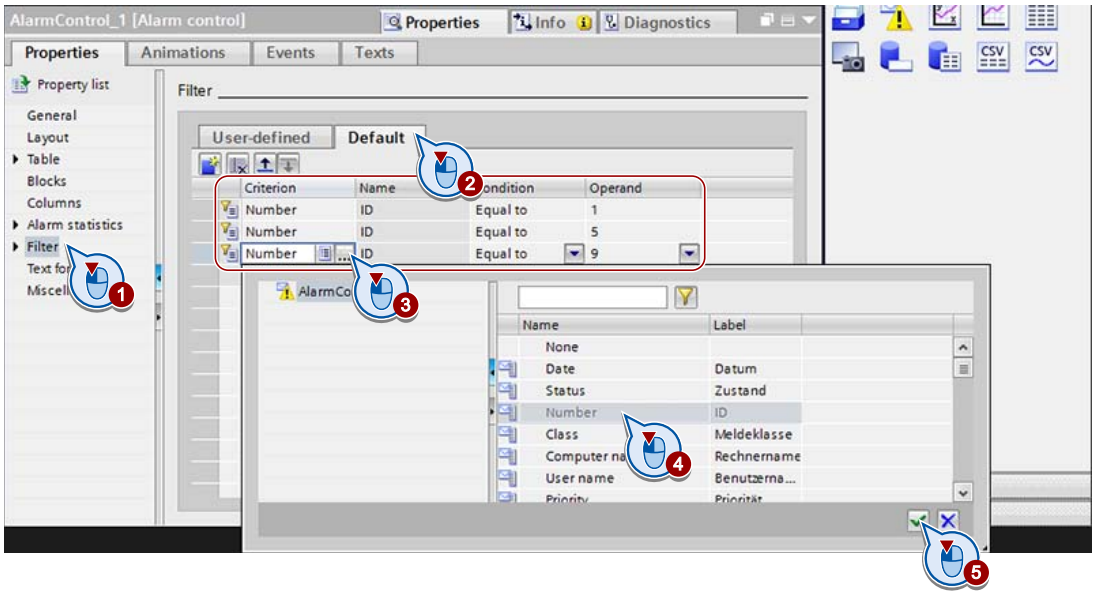
The fill level alarms of the "Cyan" tank have been configured and the alarm numbers have been assigned.

Procedure

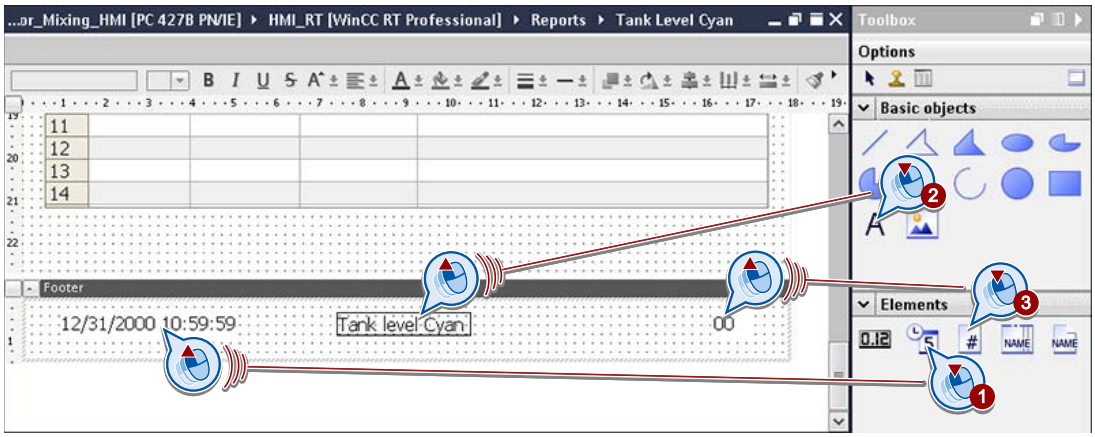
1. Create a new report, name the report "Tank Level Cyan" and insert an "Alarm view" object in the detail page of the report.



2. Configure the "Alarm view" object in such a way that the pending alarms are output with the alarm numbers 1, 5 and 9.



3. Insert the "Page number" and "Date/time field" objects, as well as a text field for the name of the log in the report footer.



Result

You have created a report for alarms concerning the fill level of the "Cyan" tank.

7.3 Reporting recipes

Introduction

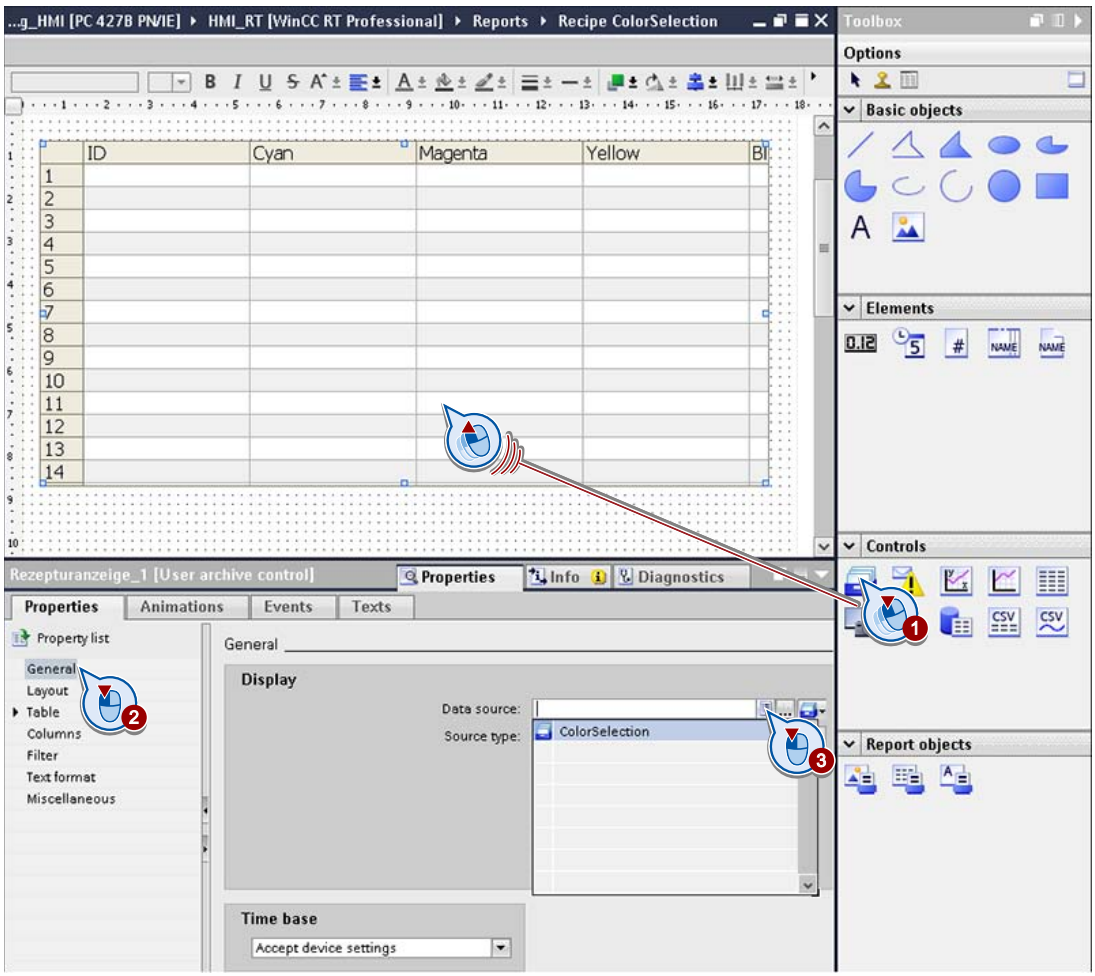
In this chapter, you will create a report in which the recipe data will be output.

Requirements

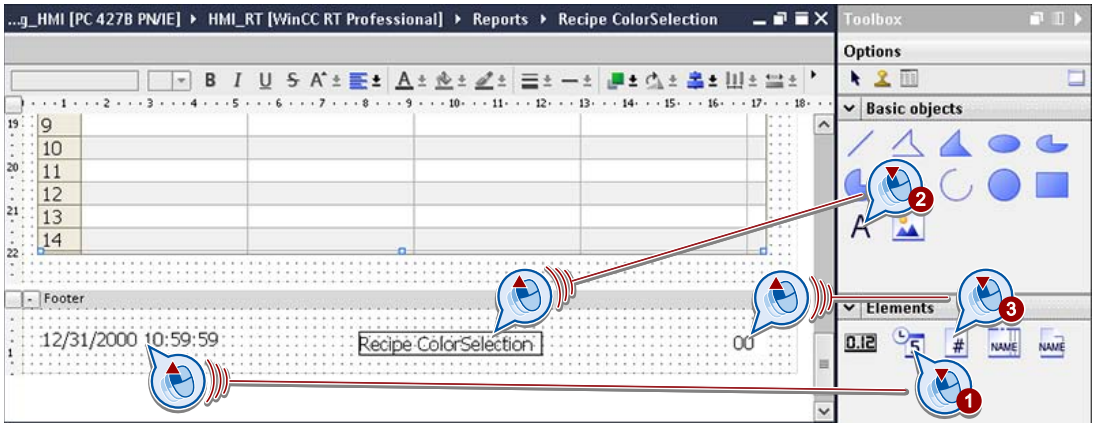
A recipe with structure recipe data records has been created.

Procedure

1. Create a new "Recipe ColorSelection" report.
2. Insert the "Recipe view" object in the detail page of the report and configure the "Recipe view" object to output all data records of the "ColorSelection" recipe.



3. Insert the "Page number" and "Date/time field" objects, as well as a text field for the name of the log in the report footer.



Result

You have created a report for the recipe.

7.4 Creating a print job

Introduction

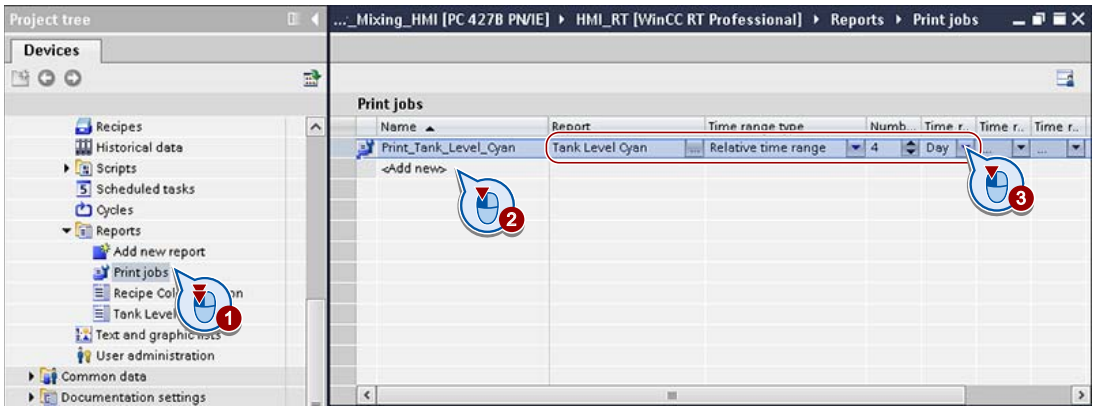
In WinCC RT Professional, you print reports using print jobs in which you define the exact data scope, the time period of the data and the time of printing. In this way, you can create regular shift reports, for example.

Requirement

The report to be printed has been created.

Procedure

1. Create the new print job "Print_Tank_Level_Cyan" and configure it: Select the report, the time range and the printer.



Result

The report is output at the set times - four times a day.

7.5 Creating a report for label printing

Introduction

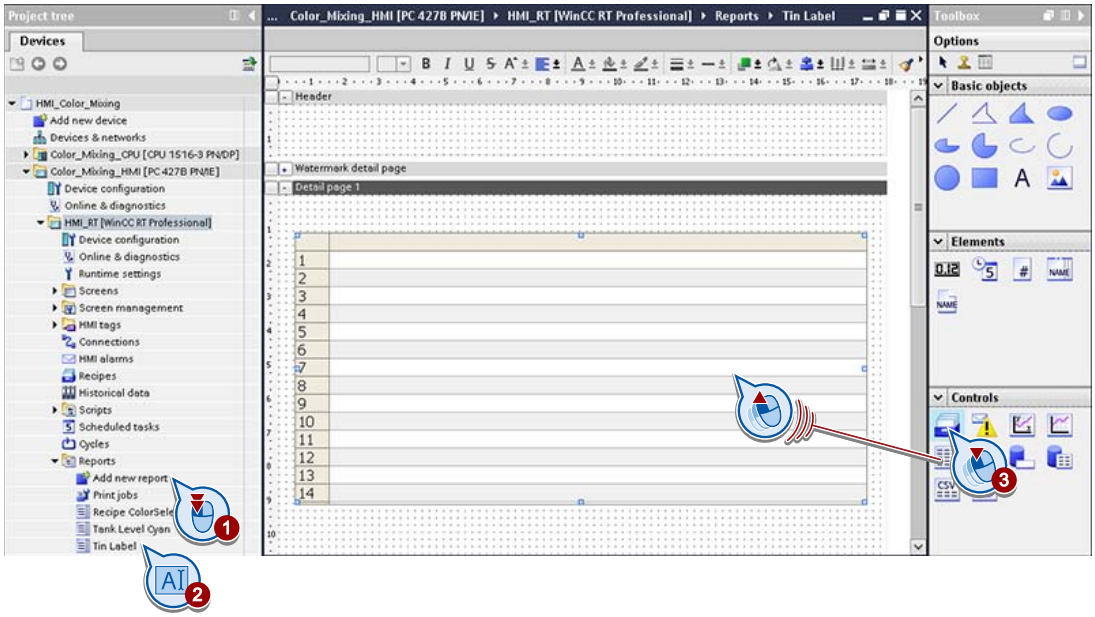
You print the current product information from the recipe data using the "Tin Label" report and a C script.

Requirement

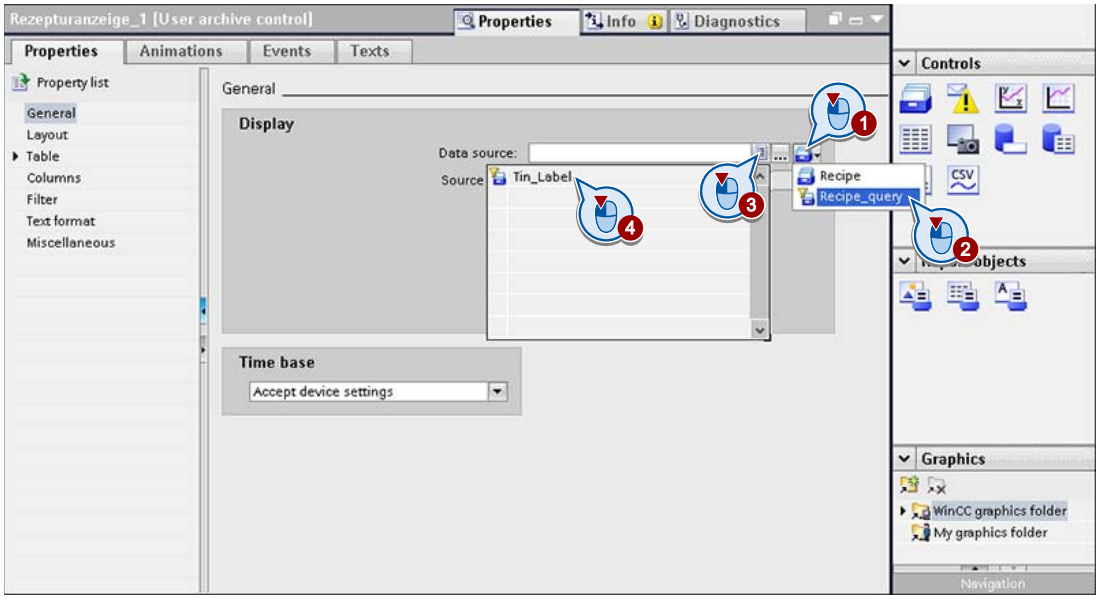
A recipe query "Tin_Label" with the data required for the label has been created for the "CompanyData" recipe.

Procedure

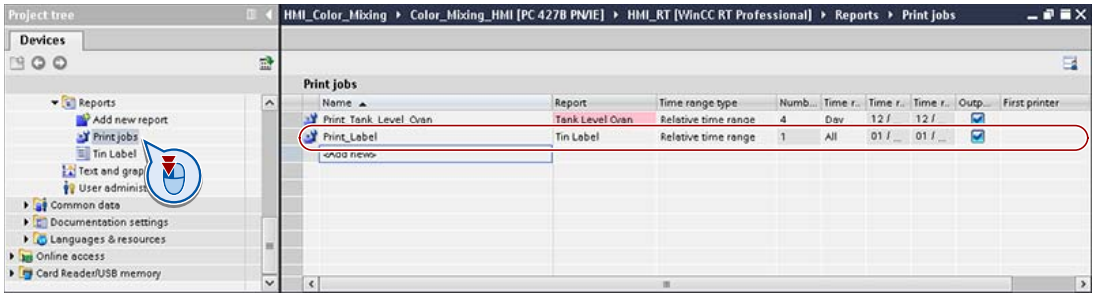
1. Create a new "Tin Label" report and insert it in the detail page of a recipe view.



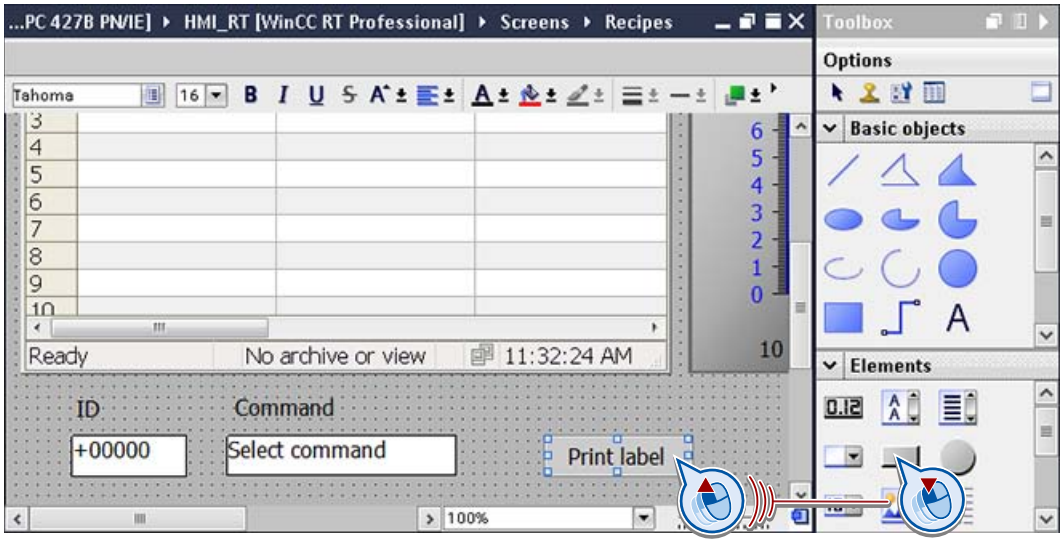
2. Configure the recipe view to display the recipe query "Tin_Label".



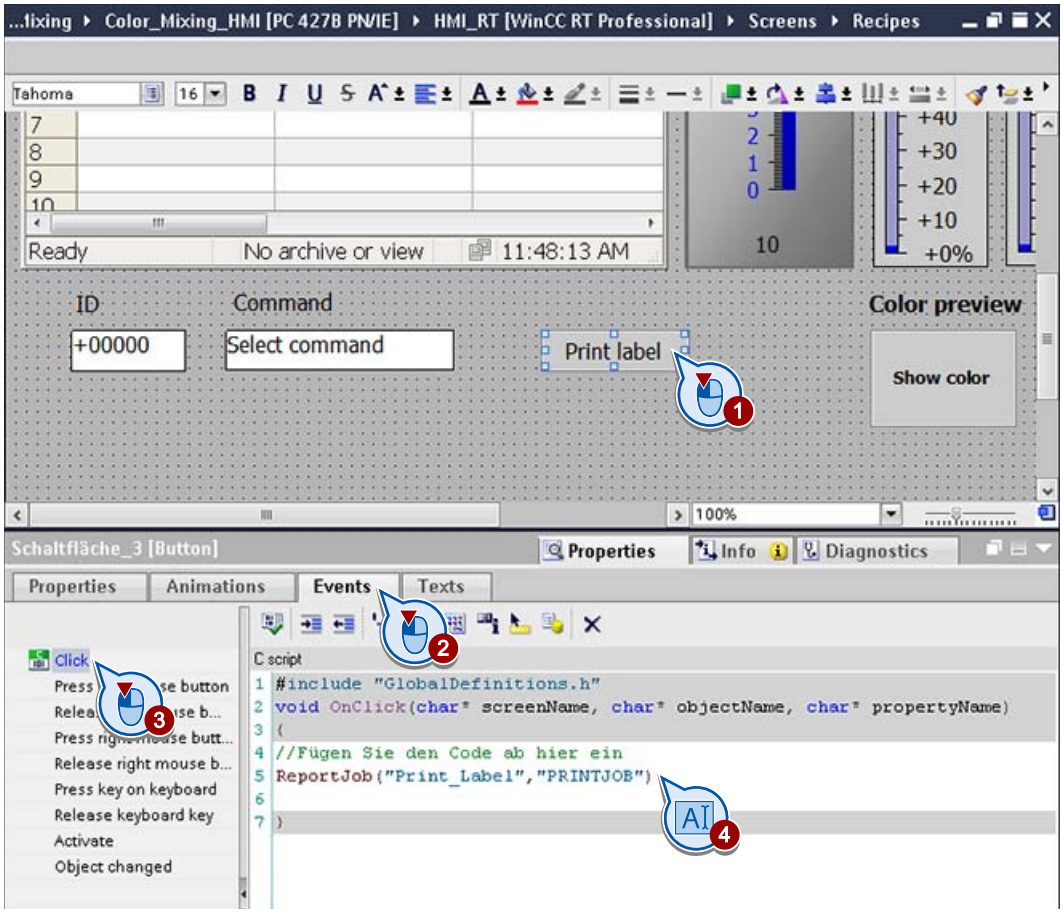
3. Create a new print job to print the "Tin_Label" report.



4. Configure the printing of the report with the recipe query:
   – Open the "Recipes" screen and insert a button with the "Print label" label.



– Open the Events tab, click the "C script" button and configure a C script to print the report at the "Click" event for the button.



Result

In runtime, the operator can start the printing of the recipe query with the product data in the "Recipes" screen, for example, at the end of a shift.

Creating user-defined functions

8

8.1 Introduction

Extend the available functionality on the HMI device by means of user-defined functions. WinCC RT Professional provides programming interfaces for VBS and for C.

- VBS is short for Visual Basic Script and is a script language of Microsoft. VBS is related to Visual Basic and Visual Basic for applications.
- C is a programming language that supports a wide range of applications.

VB scripts

This chapter does not provide any information on programming in VBS. It will, however, provide you with an insight into how VB scripts are integrated in WinCC. You do not need any knowledge of programming in VBS to understand the topic covered.

Note

VBScript

In the WinCC online help in the TIA Portal, you can find a description of all available objects, object types, properties and methods, including syntax examples, under the keyword "VBS object model".
You will find basic information on VBS and details concerning the language elements on the Microsoft homepage:
http://msdn.microsoft.com/en-us/library/t0aew7h6.aspx

C scripts

No separate C script will be presented in this chapter. In the "Configuring reports" chapter, you already learned how to configure a C script directly at an event and thus start printing, for example.

Note

C scripts

In the WinCC online help in the TIA Portal you can find, under the keyword "Runtime Scripting", more information and numerous application examples on integrating C scripts. Under the keyword "C Scripting", you will find system functions, structure definitions and the C library.

Tasks

The sample project uses two user-defined functions to display the mixed printing color in the screens on the monitor.

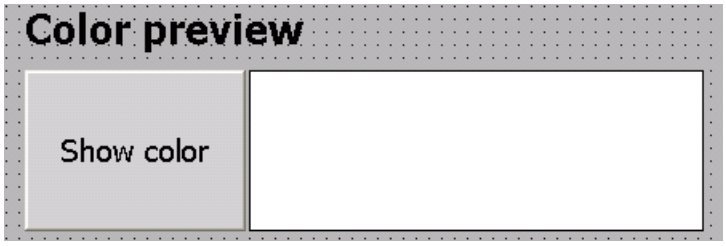
- "Recipescreen" sets the selected color for a rectangle in the "Recipes" screen.
- "Startscreen" sets the currently mixed color for the label of the filled tins, among other things, in the plant overview of the start screen.

Below, you will learn how to color a rectangle in the "Recipes" screen using a script.

8.2 Configuring the color selection view

Introduction

You can check the selected color in the "Recipes" screen on the HMI device. The objects are already created on the screen. A button click triggers the selected coloring of a rectangle. For this, a user-defined function is configured on the button.



Tags

The color mixing station mixes printing colors according to the CMYK color model. To enable the resulting color to be displayed on the HMI device screen, the controller converts the four separate color portions of the CMYK model into three separate RGB color portions that can be processed further by means of a user-specific function.

You need the following tags for communication between the HMI device and the controller:

Meaning	Name	Controller connection	Data type	Acquisition cycle
RGB value red	HMI_Recipe_RGB_Convert_R	Yes	Int	1 s
RGB value green	HMI_Recipe_RGB_Convert_G	Yes	Int	1 s
RGB value blue	HMI_Recipe_RGB_Convert_B	Yes	Int	1 s

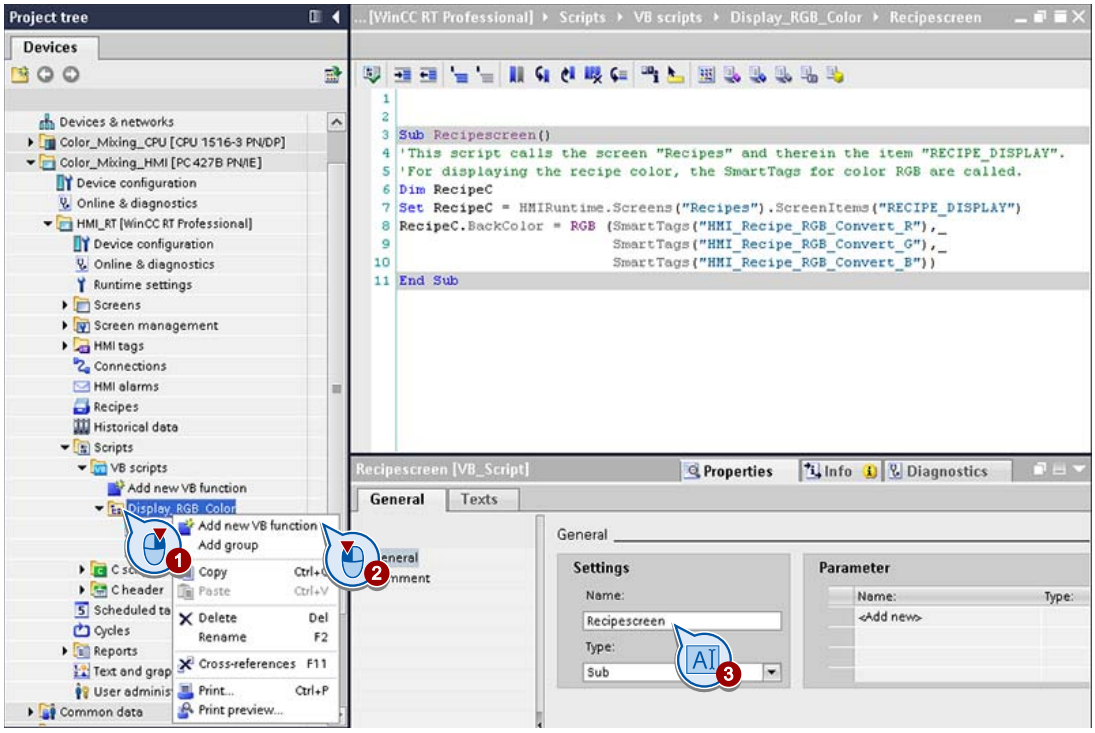
Requirement

You have created the controller tags.

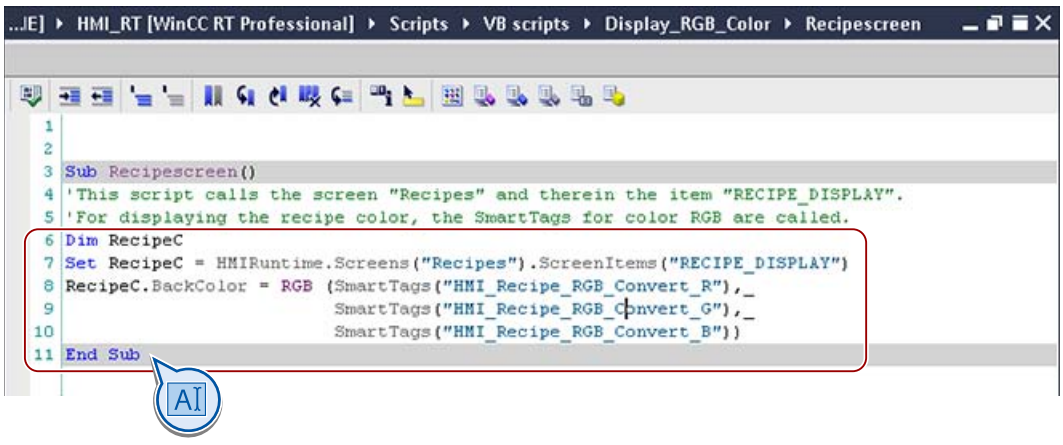
Creating the script

The "Recipescreen" script has already been created in the sample project. The next two steps show the basic procedure.

1. Under "Scripts", create a new user-defined function with the name "Recipescreen".



2. Write the following code:



The "Scripts" editor assists you in entering code:

- You can insert code templates for frequently used instructions with the "Instructions" task card.
- Autocomplete allows fast, error-free writing.
- Syntax highlighting improves orientation in the code.

- Use the keyboard shortcut <Ctrl+J> to call the object list to, for example, supply values to system functions, methods, and properties. Select screens, screen objects, tags or colors, for example. You transfer a selected object from the object list into the code by double-clicking it.
- You can use the keyboard shortcut <Ctrl+Space> to call a list with constants and functions of the programming language used in the "Scripts" editor.

Note

What does the user-defined function do?

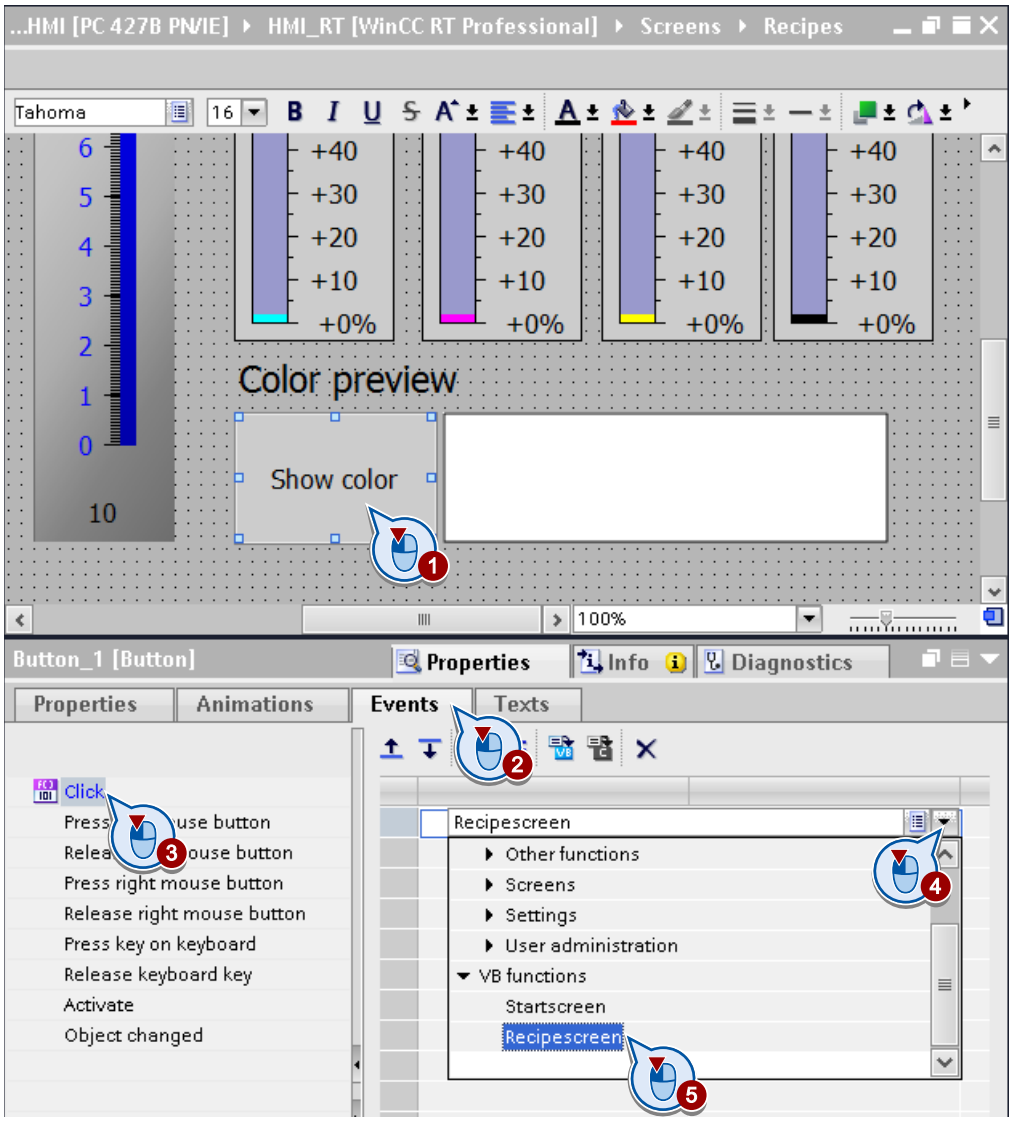
The script uses "SmartTags" objects to read the Red, Green and Blue elements from the corresponding process tags and transfer these to the VBS object "RGB".

"RGB" converts the triple comprised of the red, green and blue elements into a single number that corresponds to a screen color.

The result of this conversion is set as background color for the rectangle.

Integrating a user-defined function into the screen

1. Go to the "Recipes" screen.
2. Configure the execution of the user-defined function "Recipescreen" at the "Show color" button.



Result

The operator can display a preview of the selected color on the screen.

Configuring user administration 9

9.1 Introduction

Basics of user administration

WinCC allows you to restrict security-relevant operations to specific user groups and thereby protect runtime data and functions against unauthorized access.

- Set up user groups and assign authorizations to these groups. You then configure the authorizations required for operation of safety-related objects.
- Each user is a member of a user group and inherits the authorizations from this user group.
- In runtime, users must log on with their user name and password. The requested action is executed provided the user is authorized accordingly.

Sample project

The following requirements are valid for the sample project:

- Users, user groups and authorizations are configured in the user administration of the HMI device and transferred along with the project to the HMI device.
- All users may change their personalized passwords in runtime.
- Administrators are allowed to create new users in runtime and to change the password of other users, for example, of a user who has forgotten the password.
- Field service technicians may partially operate and monitor the plant. They are not granted access to recipes and may not administer users.
- Administrators can identify themselves and log on using a chip card.

Tasks

- Creating user groups for administrators, field service technicians and users, and assigning group authorizations
- Creating a user for an individual and assigning this user to a user group
- Configuring the user view to allow user data to be modified in runtime
- Disabling access of field service technicians to recipe management
- Configuring logon of a user depending on a tag value

9.2 Creating user groups

Introduction

The following two user groups are already configured when you create a project:

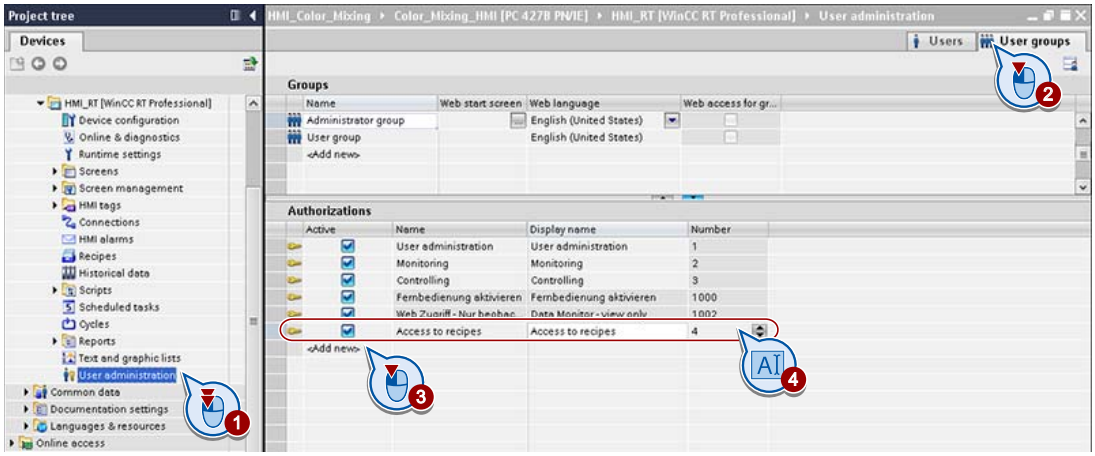
- Administrator group: The members of this group have access to all data and functions. These include the user administration and the operator control and monitoring of the plant.
- User group: The members of this group have the authorization to operate the plant.

You only need to create a new user group for the field service technicians in the sample project. Service technicians are authorized to operate and monitor parts of the color mixing system:

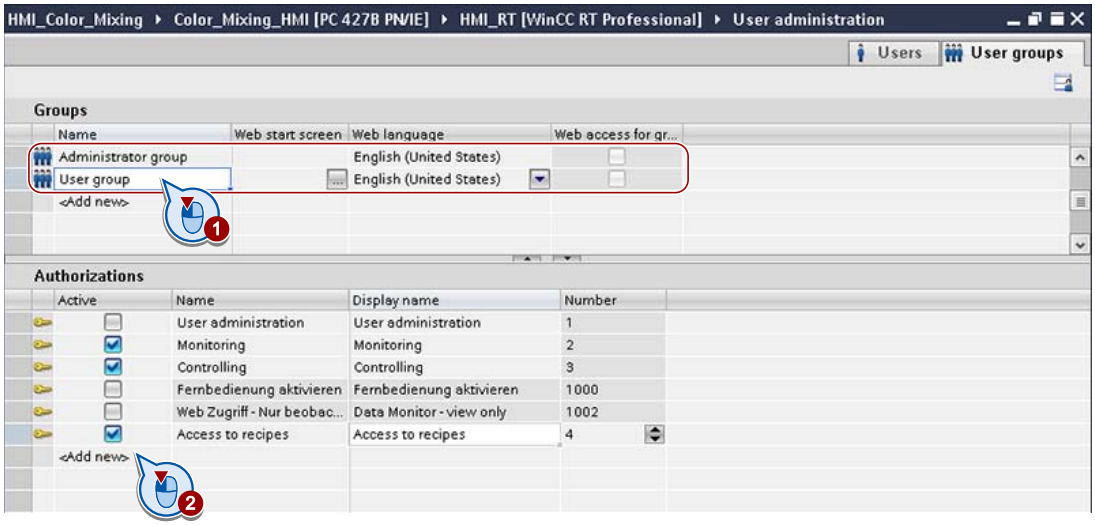
- Service technicians may change their password in runtime, but not administer users.
- Service technicians are denied access to the recipes.

Procedure

1. Create the "Access to Recipes" authorization in the user administration dialog.

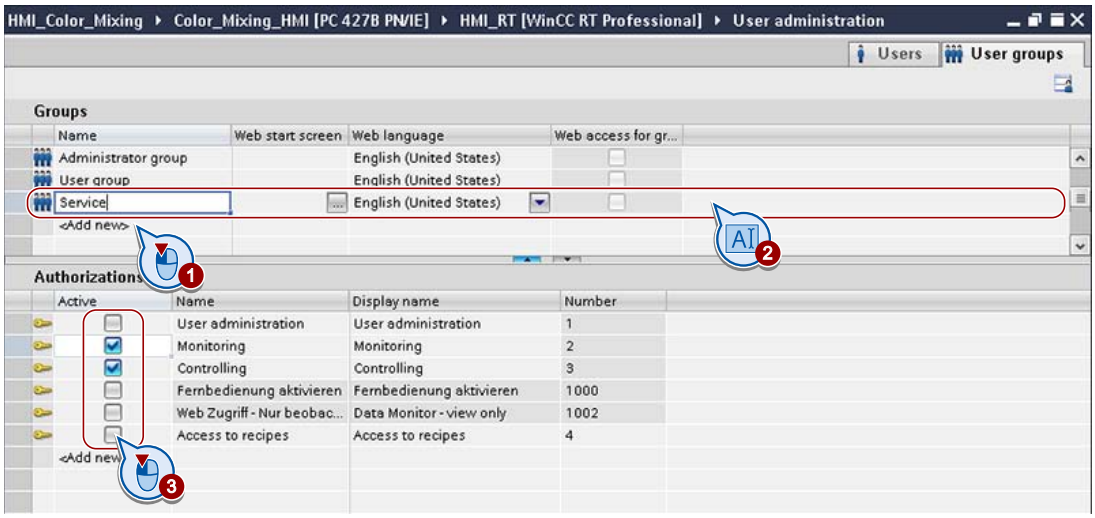


2. Grant both existing user groups the additional authorization for "Access to Recipes".



3. Create a new "Service" user group and activate the default "Monitoring" and "Controlling" authorizations.

   You can leave the "User administration" and "Access to Recipes" authorizations disabled.



Result

A separate user group is created for service technicians. All user groups are equipped with the required authorizations.

9.3 Creating users

Introduction

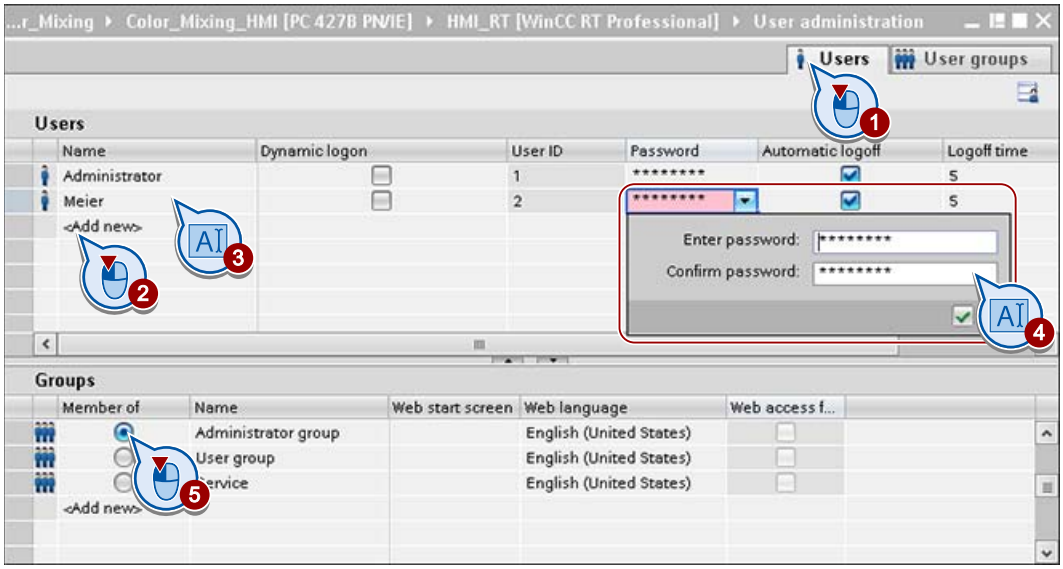
The "Administrator" user is created automatically during the creation of a project. Four more users are to be granted access to the HMI device:

- Administrator Mr. Meier
- Service technician Ms. Lopez
- Operators Ms. Greenwood and Mr. Peters

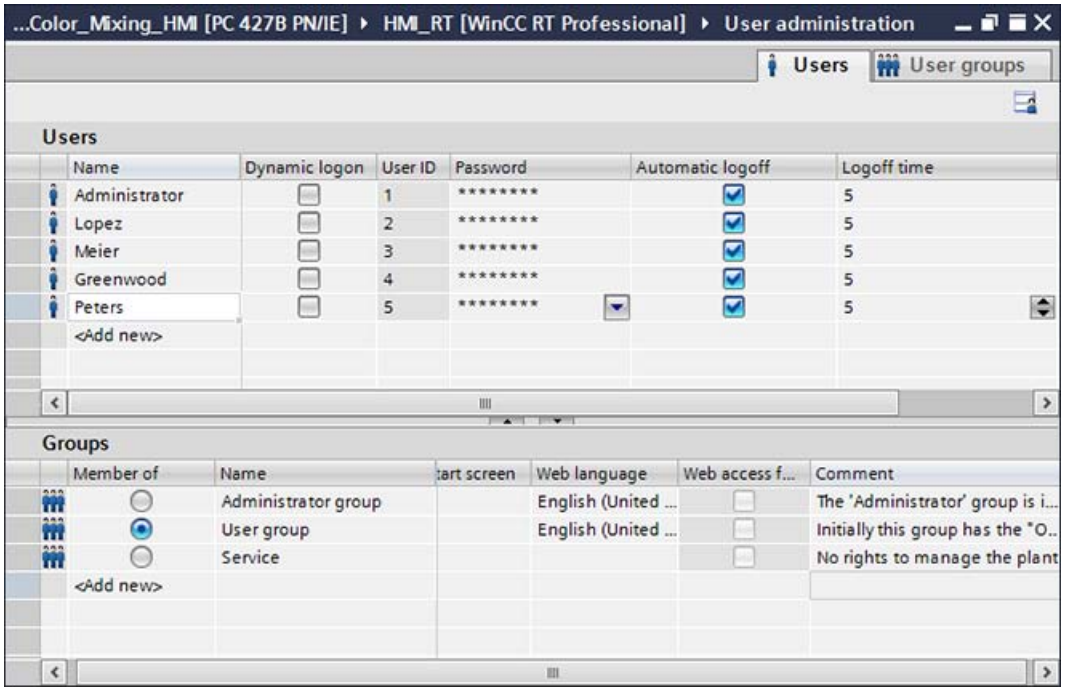
Procedure

1. Change to the user administration dialog, "User" tab (1).
2. Add Mr. Meier as new user (2+3).
3. Enter a provisional, temporary password for Mr. Meier (4).

4. Assign Mr. Meier to the group of administrators (5).



5. Add the other users in the same way. Assign each person to the user group that corresponds to their tasks.



Result

All users of the HMI device have been created and are members of a user group. If configured appropriately, access to data and functions in runtime is restricted to authorized users.

Note

Additional security settings

Select "Runtime settings > User administration" to configure additional security settings for the users and user passwords. Here, you specify that users must change their password after a certain number of days have elapsed, for example.

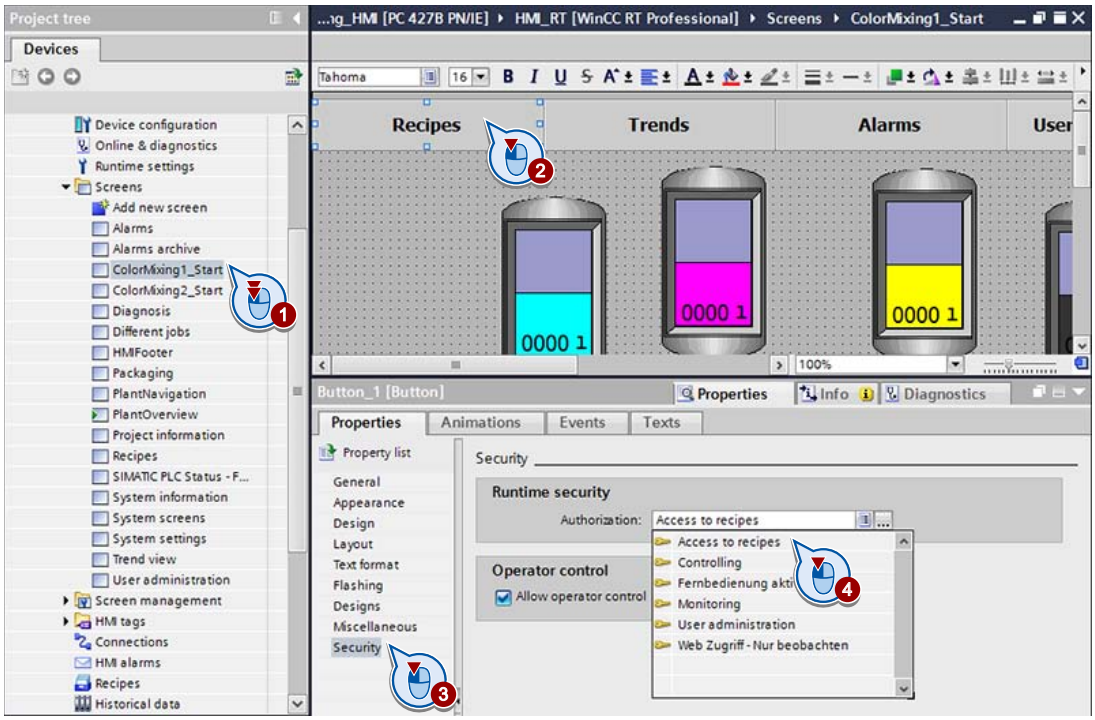
9.4 Configuring a button with access protection

Introduction

Service technicians are not to be permitted to access recipes. To ensure that this access is denied, set a corresponding access security function for the button that is used to call the recipe screen.

Procedure

1. In the start screen, select the button that opens the "Recipes" screen (1 + 2).
2. Under "Security" in the inspector window, specify that this screen is only displayed to users who have been granted the "Access to Recipes" authorization (3 + 4).



Result

Access to the "Recipes" screen is blocked for the "Service" user group in runtime.

9.5 Creating a screen for user administration

Introduction

User and password settings are downloaded along with the project to the HMI device.

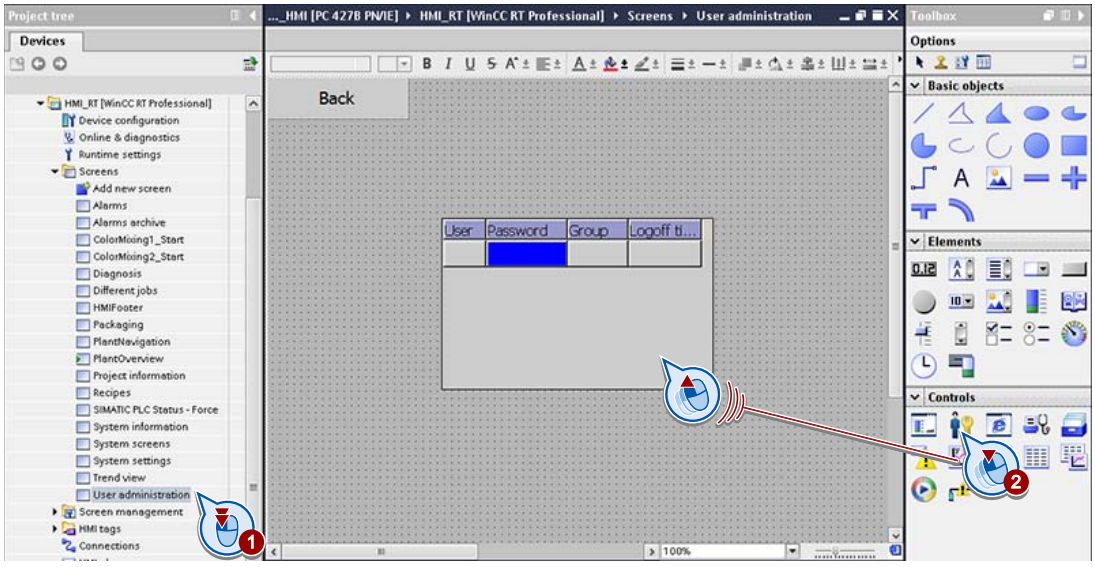
The "User view" object also allows you to manage users and passwords on the HMI device.

- The full functional scope of the user view is available to users authorized for user administration. They may create users, or delete and change their own password or that of other users.
- Users not granted this authorization change their password in the user view.

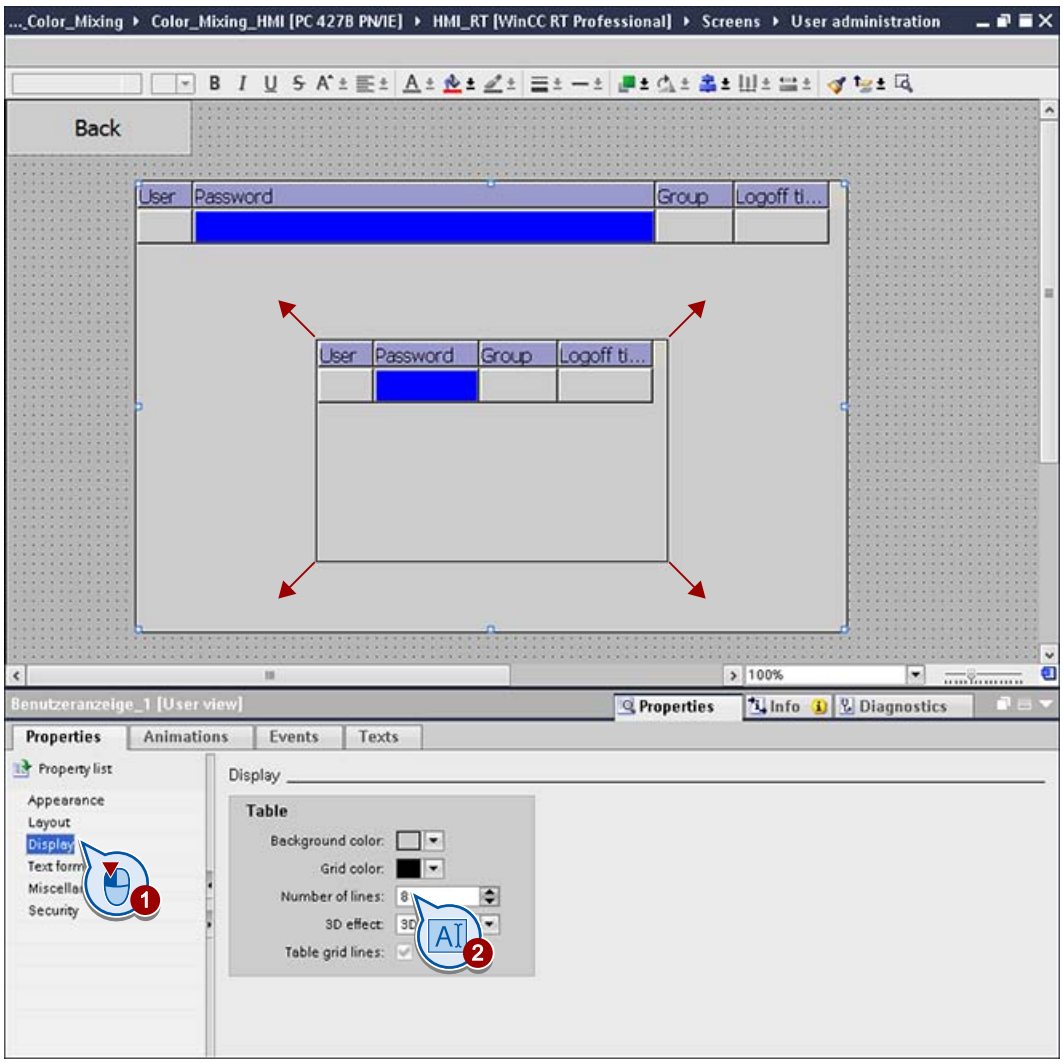
In our sample project, only the administrators are authorized to manage users in runtime.

Procedure

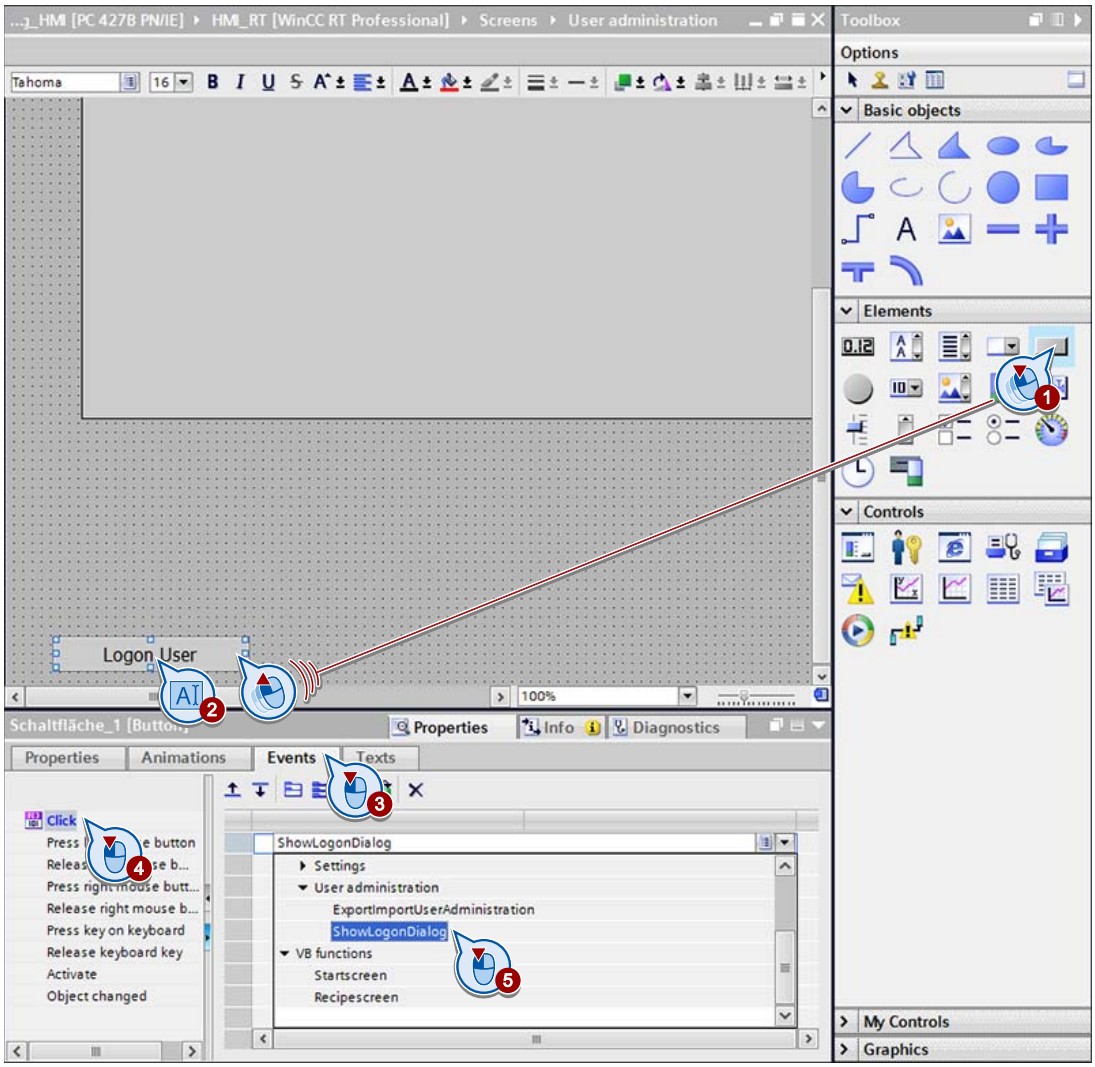
1. Open the "User Administration" screen and insert the "User view" object in the screen.



2. Use the mouse to zoom the object as required and configure the properties in the inspector window:
   - Specify the maximum number of users to be listed, for example.



3. Complete the process by inserting a button to open the logon dialog:



Result

Users and passwords can be managed from the HMI device.

Testing

The configuration should be tested in runtime following each extensive editing step. This way, inconsistencies and display errors can be detected and removed in due time.

A detailed description is available in the section "Testing and loading a project" of this Getting Started.

9.6 Configuring user logon per chip card

Introduction

In WinCC RT Professional, you can configure the logon of a user depending on a tag value.

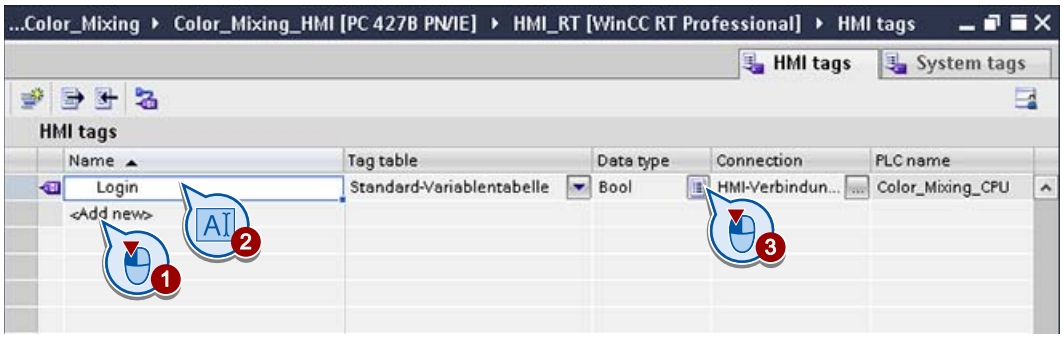
This will permit you to allow individual users to log on via a separate button, for example. You can also allow the logon procedure to start via a tag occupied by a device, for example using a chip card.

In the sample project, a user is to be able to log on to the HMI device using a chip card in order to be able to respond quickly in an emergency.

Additional information is available in the WinCC online help in the TIA Portal under the keyword "Dynamic logon".

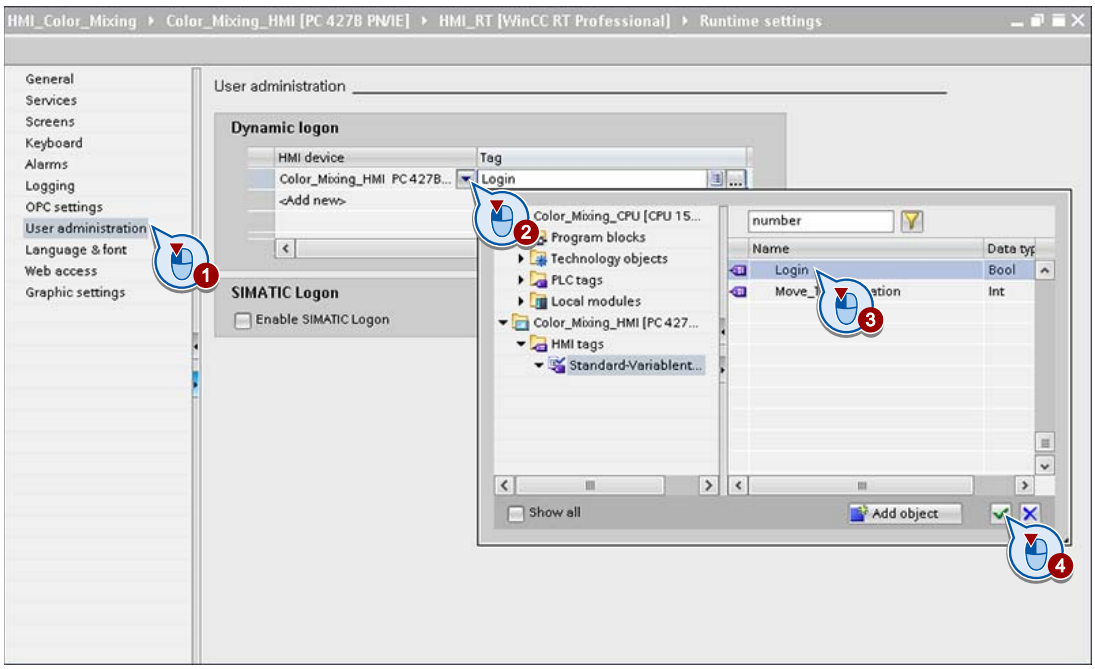
Procedure

1. Create the Boolean variable "Login" in the tag editor.

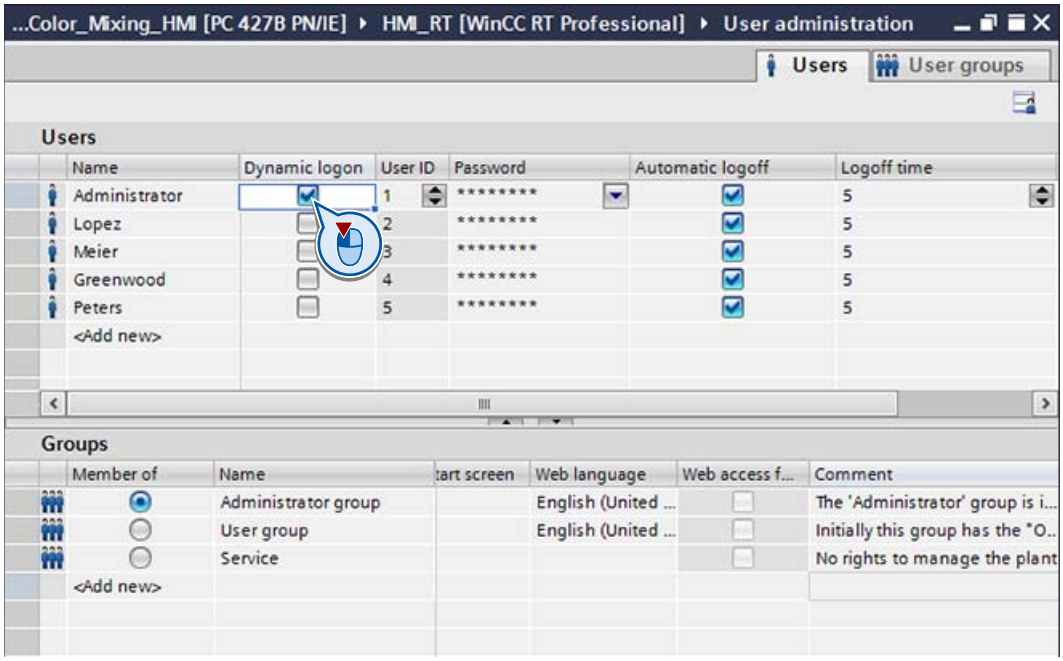


   The tag is occupied by the controller: If the chip card is recognized as valid, its value is "1"; otherwise, its value is "0".

2. Open the "User administration" in the "Runtime settings", add the HMI device to the "Dynamic logon" area, and select the configured tag "Login".



3. Open the user administration and activate the "Dynamic logon" option for the administrator.



Result

The administrator can log on to the HMI device using a chip card in runtime.

Configuring multilingual user interfaces 10

10.1 Introduction

Basics

WinCC supports multilingual user interfaces.

The configured languages are available in runtime under "Runtime settings > Language & font". With the appropriate configuration, the user can change the language directly in runtime.

All texts of a project can be translated in WinCC:

- User texts created manually in the configuration phase.

  These texts must be translated into all other available languages.
- System texts that WinCC generates automatically when you configure the project.

  These texts are generated initially in the user interface language of WinCC. You can change these texts if required.

Text examples:

- Button labels
- Display names of recipes
- Alarm texts
- Comments in tables
- Text lists

Note

Controller alarms

User-defined controller alarms are only displayed in the HMI. You must translate these alarms in the controller using STEP 7.

Tasks

The color mixing system is operated in a new subsidiary that is located in Russia. A Russian user interface must be provided for maintenance and service technicians. To this end, you add another language to the project. Once the export is completed and the texts have been translated in Russia, you import the texts and proofread the translation.

Note

Language switching in runtime

You need to configure a corresponding operating object to switch the language in runtime, e.g. a button for which you configure the "SetLangauge" system function at the "Press" event in the inspector window.

Configuring the language switch is not covered in detail in this Getting Started.

10.2 Adding languages

Introduction

The sample project is to be made available also with a Russian user interface.

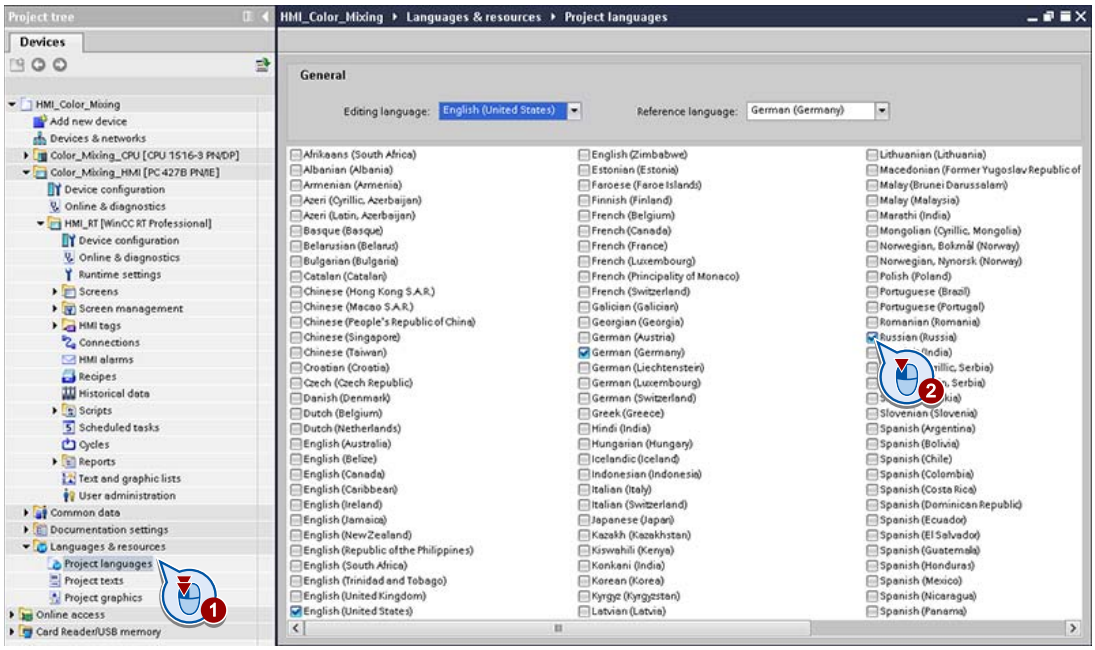
Note

Language of the operating system

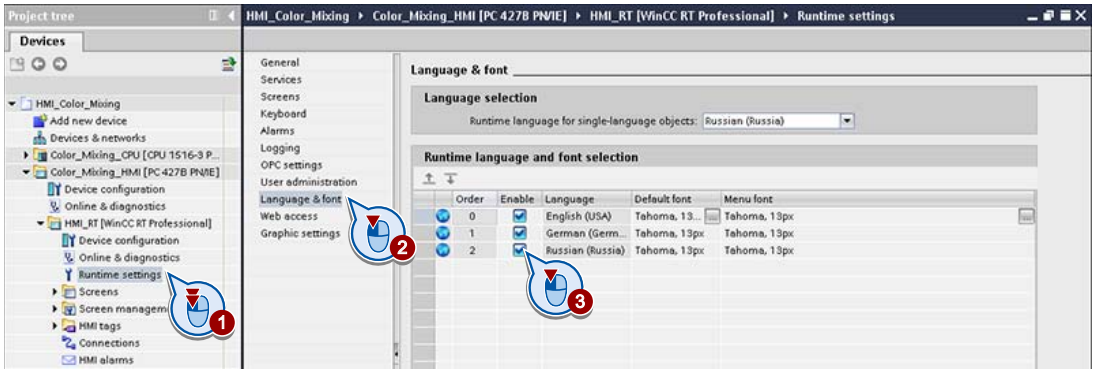
To use the language "Russian" on the HMI device, you must set the operating systems of the HMI device and the configuration PC to "Russian".

Procedure

1. Add "Russian" as new project language.



2. Activate the language "Russian" in the runtime settings to set up the Russian user interface.



Note

Order

The first time runtime starts, the project will be displayed in the language with the lowest number. When the language is switched, it switches in the order of the assigned numbers if no language was specifically set using the "SetLanguage" function.

A button for switching the language has already been configured in the "HMI_Footer" screen.

3. Runtime is to start up with the Russian user interface. For this reason, move the "Russian" language to the top position.

Result

Once you have translated the project and downloaded it to the HMI device, runtime will start with Russian user interface.

10.3 Translating user interfaces

All texts of a WinCC project can be translated. WinCC provides you with the following options for doing this:

- You can enter and edit translated texts in the set languages directly in the properties of an object.
- You can edit texts in the project languages in the "Project texts" editor.
- You can export the texts for external translation and then re-import them.
- You can change the editing language and enter the texts in this language directly in the object.

Translating texts at the object

In the "Texts" tab of an object, you can edit all texts of the selected objects in the set project languages.

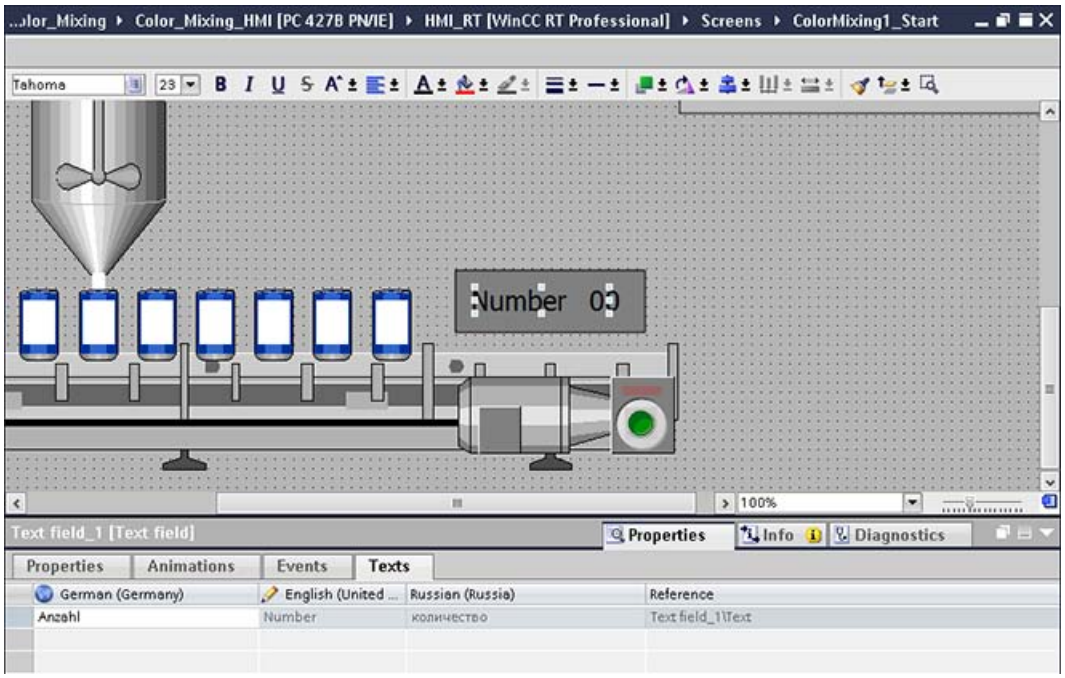
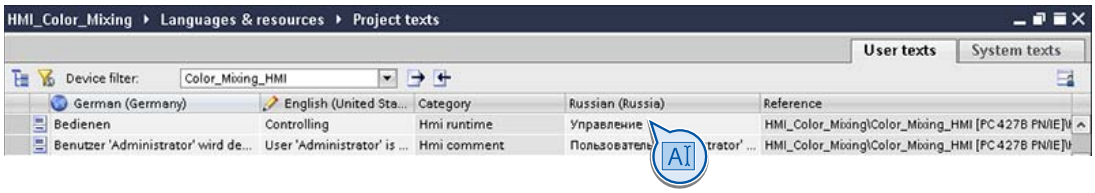


Figure 10-1 MultilingualUI_03

Translating in the "Project texts" editor

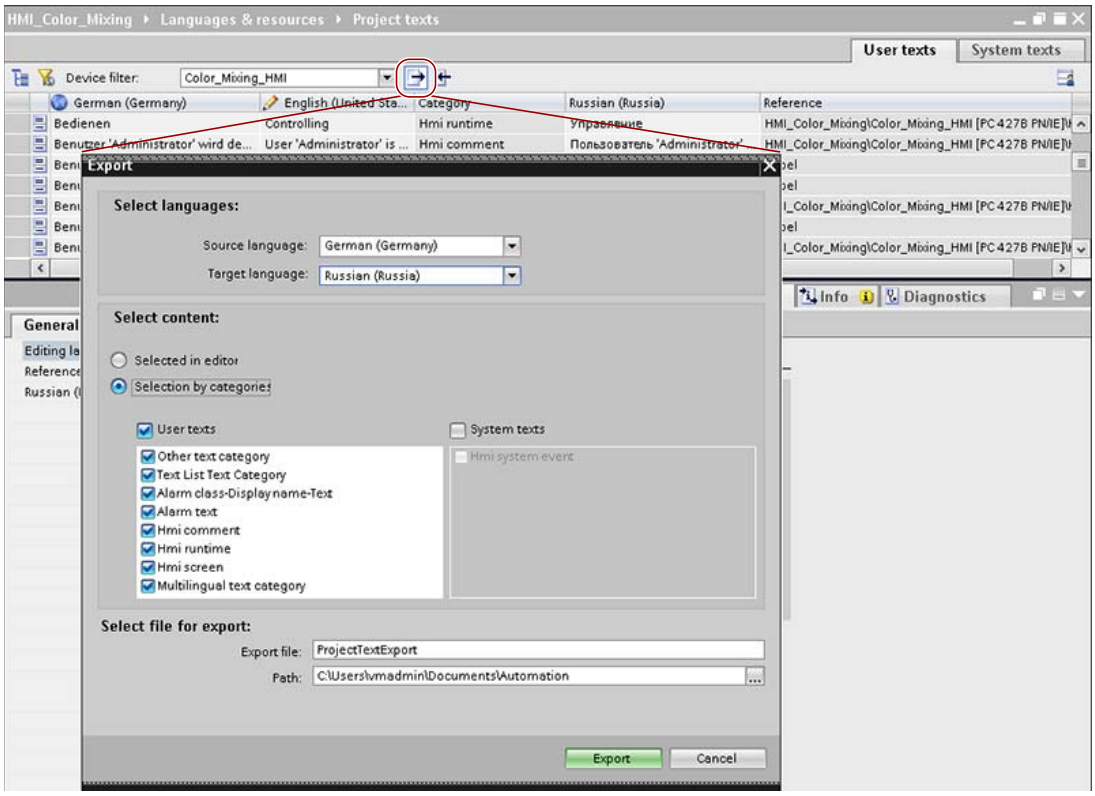
Use the "Project texts" editor to enter the translations in a clear tabular format.



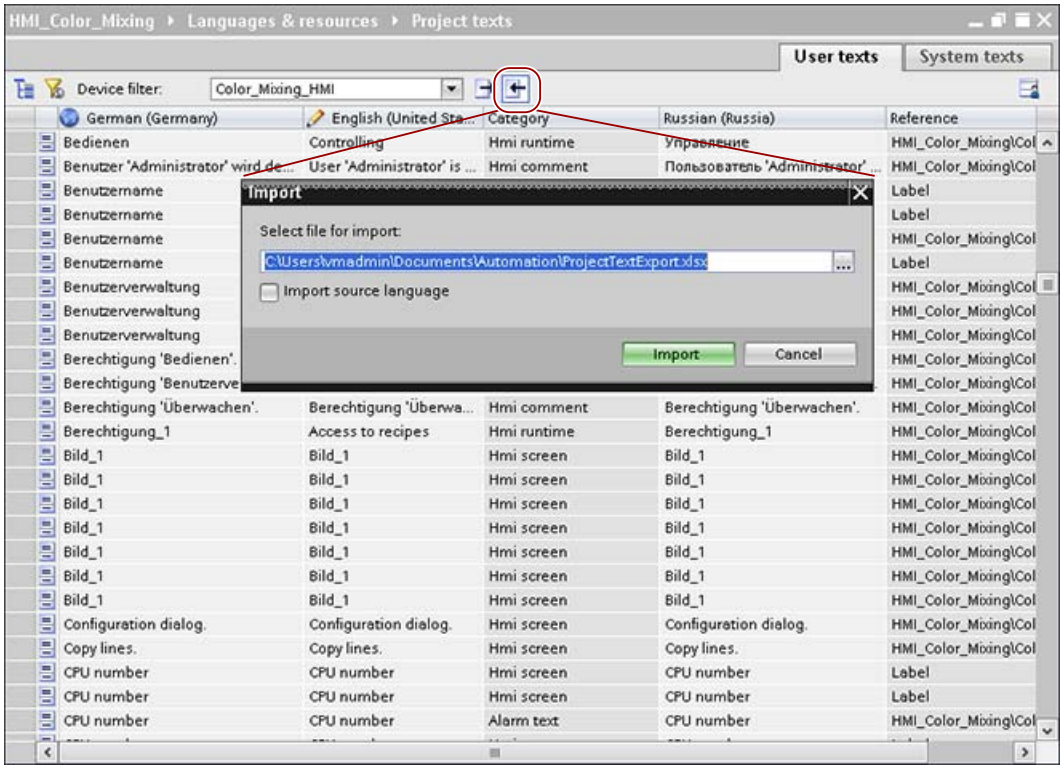
With this procedure, you translate the texts contained in the screens separately from the screens; for this reason, other language labels may no longer fit into the button and are truncated. Therefore, it is highly recommended that you check the screens following translation.

Translation with export and import

You export all selected texts at once in the "Project texts" editor.

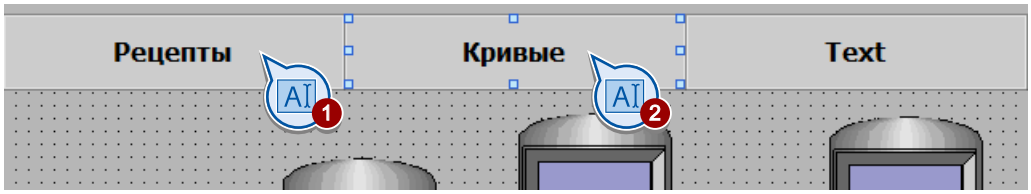


Following translation, you import the translated texts into the "Project texts" editor.



Entering translations in the editing language

The editing language determines the language used to process the project in WinCC. By changing the editing language in the "Project languages" editor, you can directly enter, proof and correct your translations in the HMI screens and WinCC editors.



Note

When using this procedure for the first translation into a new language, you run the risk of missing text elements, e.g. alarm texts. We therefore recommend this procedure to check the translated texts in screens and to translate or correct individual terms.

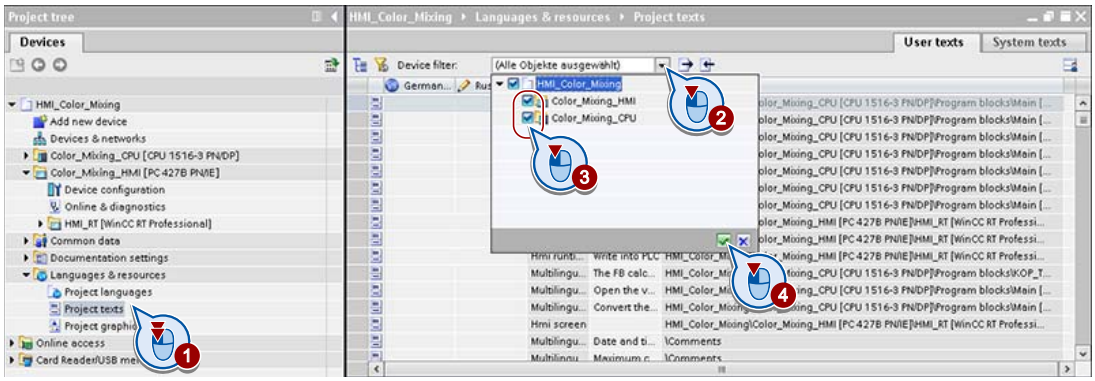
10.4 Entering Russian texts

Introduction

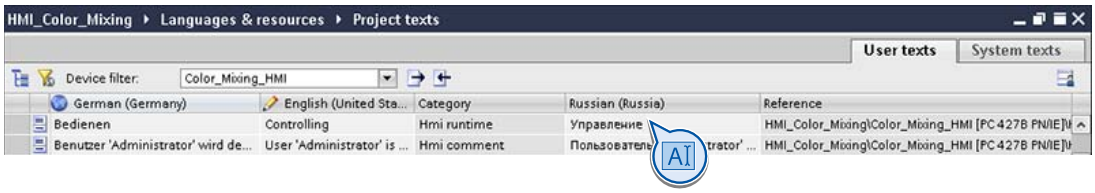
The translation of the HMI user interface contains incorrect texts. After having loaded the external translation, correct the incorrect text elements using the "Project texts" editor.

Procedure

1. Display the user texts of the project (1).
2. Set the "Device filter" to restrict the display to the texts belonging to the selected HMI device (2 - 5).



3. Correct the Russian user interface texts.



Result

The project texts are translated completely.

10.5 Proofing the translation

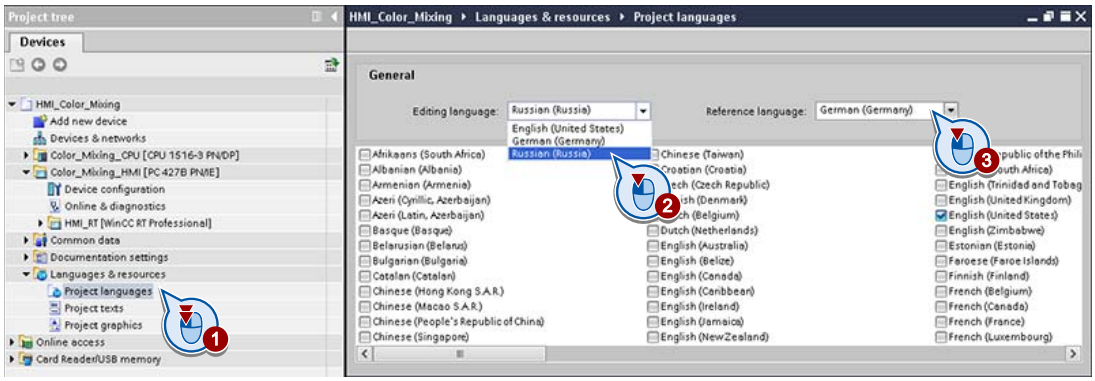
Introduction

WinCC supports you in proofing translations.

- After the editing language is changed, all user interface texts are displayed in the new set language.
- Untranslated texts are shown with the standard text "Text"; untranslated names with the corresponding standard object name.
- For proofing, the terms may also be displayed in the reference language.
- In the screen editor, all texts included in a selection are displayed in the "Texts" tab in the inspector window.

Procedure

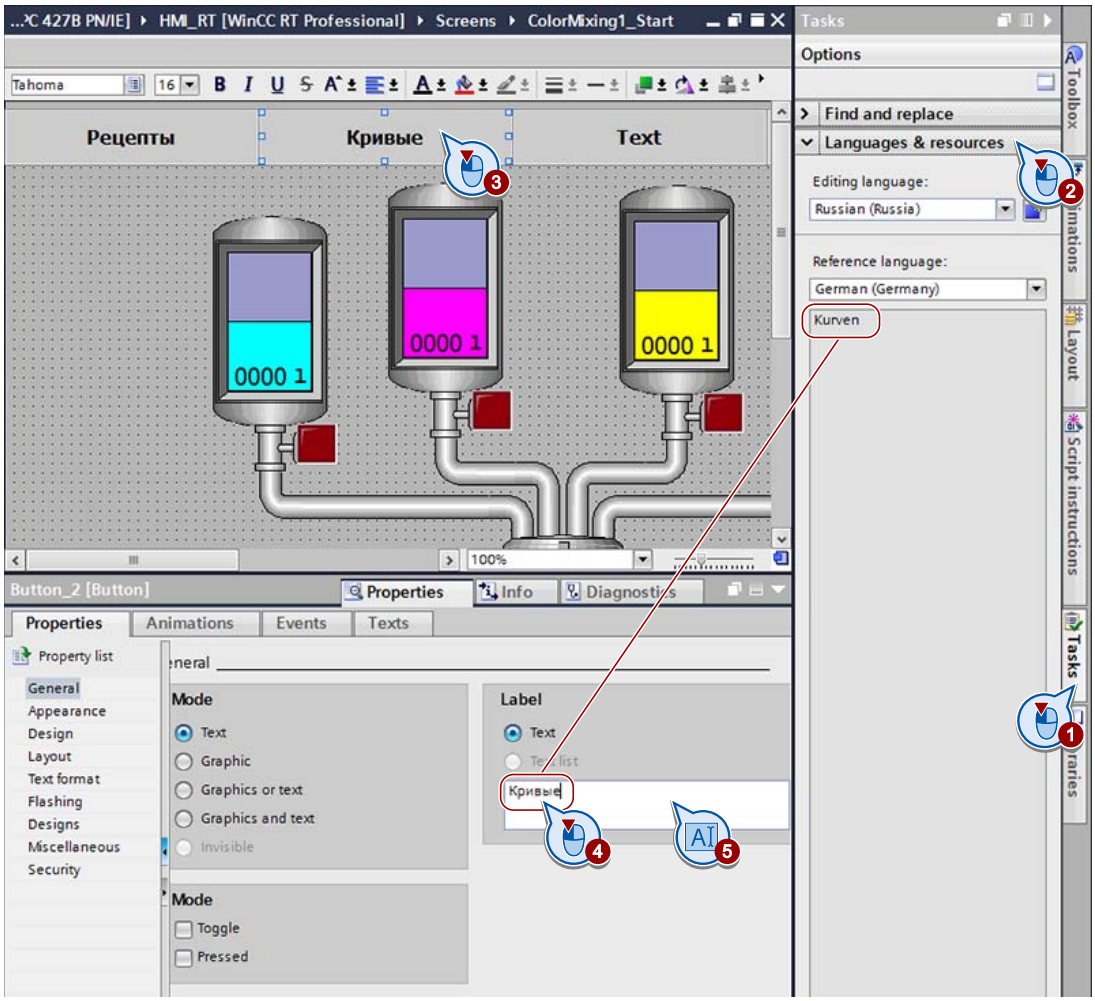
1. Select Russian as editing language and German as reference language.



2. Check the texts in the project.

   It is useful to display the texts in the reference language in the "Tasks" task card (1 - 4).

   Simply overwrite the texts you want to supplement or correct (5).



Note

Texts are displayed in the reference language only if the text was selected for editing. The text is not displayed in the reference language if only the object is selected.

Result

A Russian language version of the user interface is available.

Testing and loading a project

11

11.1 Introduction

Basics

A project can be tested for its current status in runtime at any time in the configuration phase. To this end, the WinCC runtime simulation is used.

Note

The configuration should be tested in runtime following each extensive editing step. This way, inconsistencies and display errors can be detected and removed in due time.

After completion of the configuration phase, the project is tested with controller connection. The complete control program for the color mixing station is contained in the sample project.

To use the HMI device during operation, load the tested project from the configuration PC to the HMI device.

Tasks

In the following chapter, you will learn how to simulate your project in runtime with and without a controller connection. Subsequently, you will load the project to the HMI device.

You will perform the following tasks:

- Checking the consistency of the project and compiling the project in runtime
- Testing the WinCC configuration in runtime

  In the simulation, various functions can be tested, e.g. display of the recipe data records and their visualization. A sample screen change is explained in this Getting Started.
- Compiling a control program in runtime and loading it to the controller from the configuration PC
- Operating the plant and simulating the program sequence in runtime

  The simulation is explained with a real controller in this Getting Started. The controller connection in WinCC can also be displayed using a tag simulator.
- Loading the project to the HMI device

  You have several options to load your project in WinCC. This Getting Started explains how to load a project directly from the configuration PC to the HMI device using a network path.

Ending runtime

The following button is available to stop the simulation in your sample project and return to WinCC:

11.2 Testing project without controller

11.2.1 Compiling a project

Introduction

In this section, you will compile the configured project into data which can be read by the HMI device. The consistency of the project is checked during compilation in WinCC. The project can be compiled and tested in runtime only under the condition that it contains consistent data. The project is re-compiled when runtime is started to make sure that all changes were compiled.

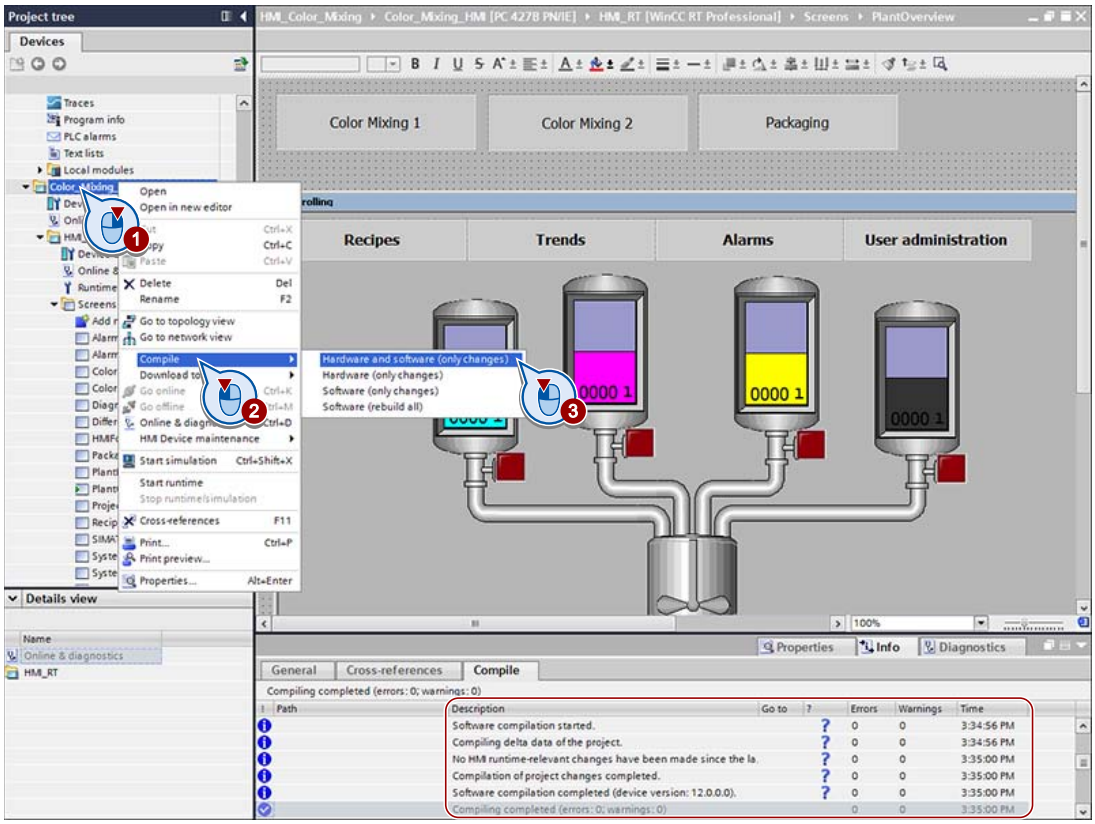
Procedure

1. Start the compilation of the project.
2. Check the compilation process in the inspector window.

   The inspector window indicates when the compilation has been completed without errors.

Note

Errors and warnings are shown in the inspector window during compilation. Click on the corresponding alarm to get to the error location. Correct the error and restart the compilation.



Result

The project has been compiled without errors and can be run in runtime. The project can now be tested in runtime.

11.2.2 Simulating a screen change

Introduction

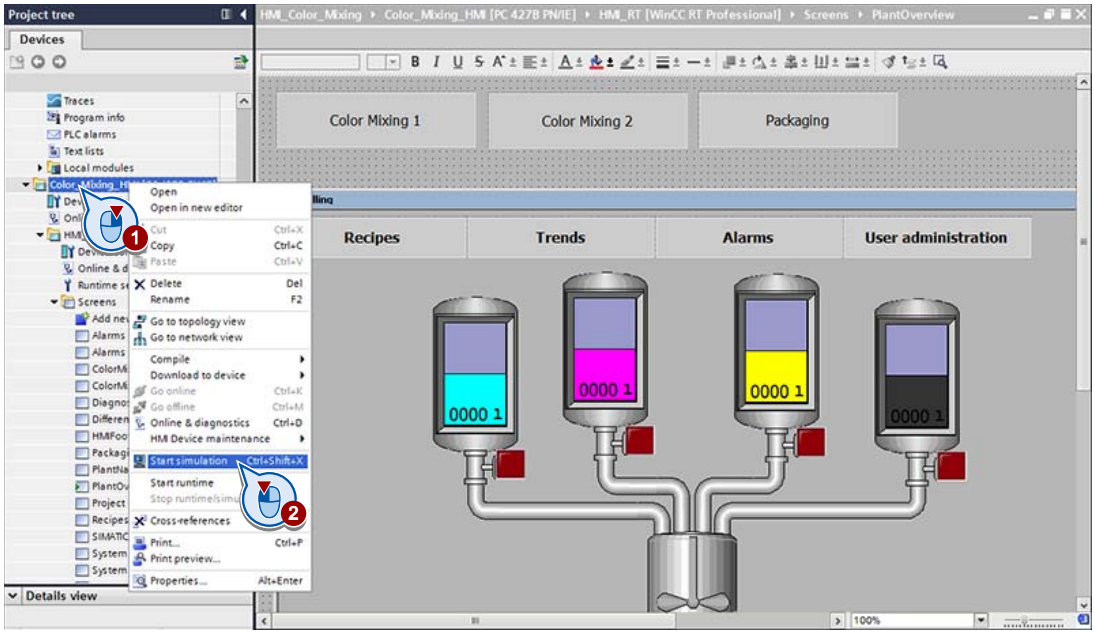
The project has been compiled successfully in runtime. Below, you will test your project by way of example in runtime simulation by calling an access-protected screen.

Requirement

- You know the password for the user "Lopez".

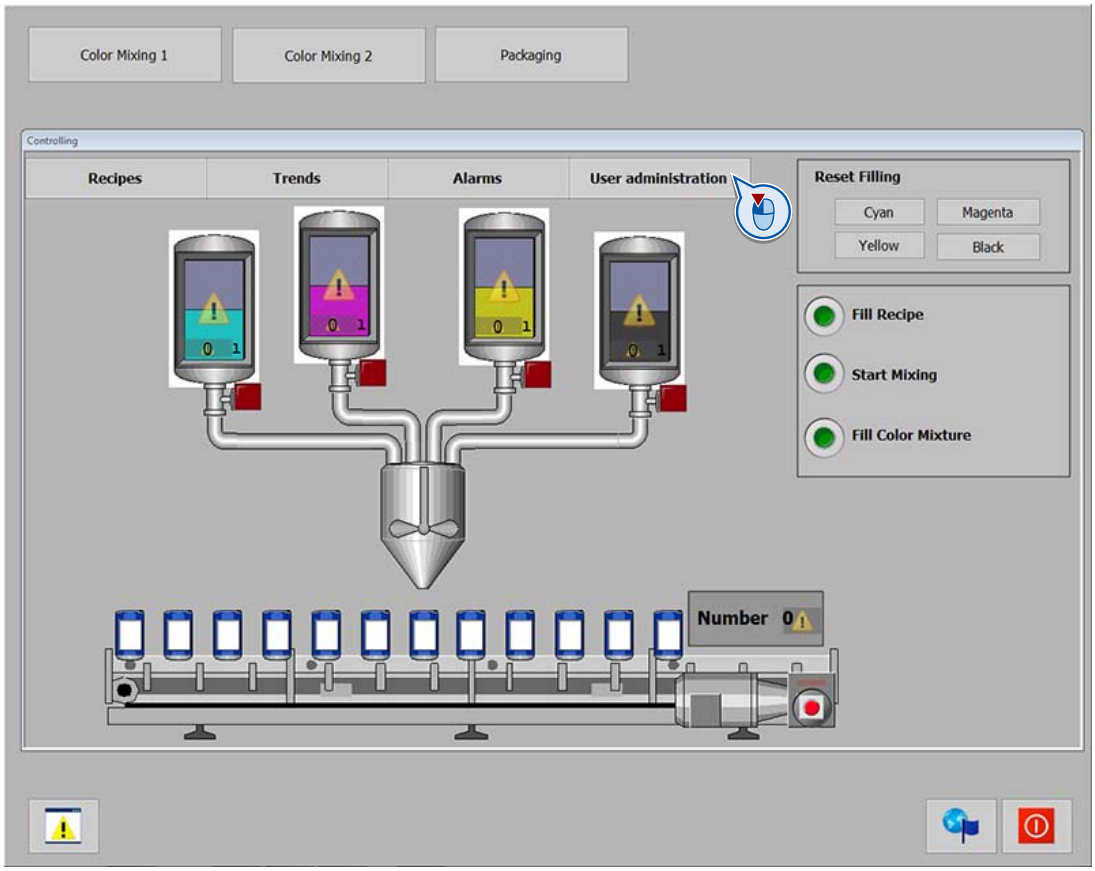
Procedure

1. Start the runtime simulation on the configuration PC.

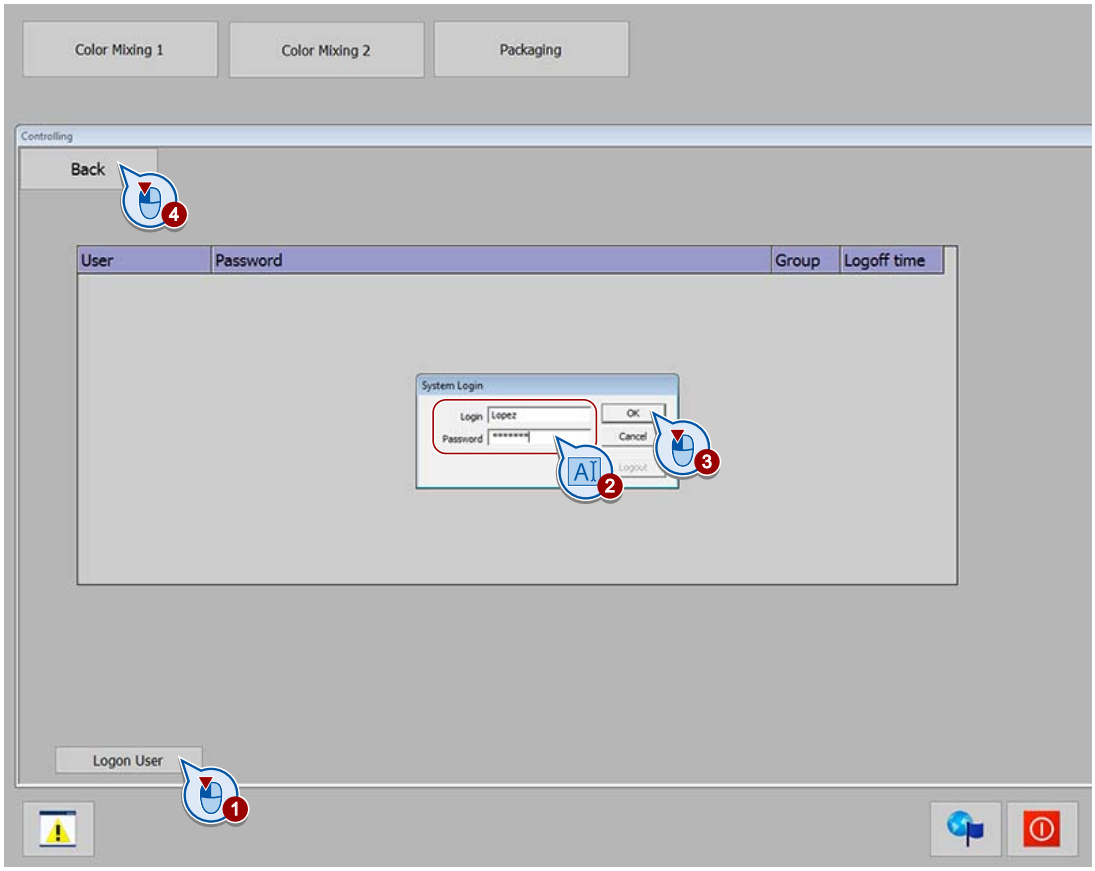


2. Close the alarm window.

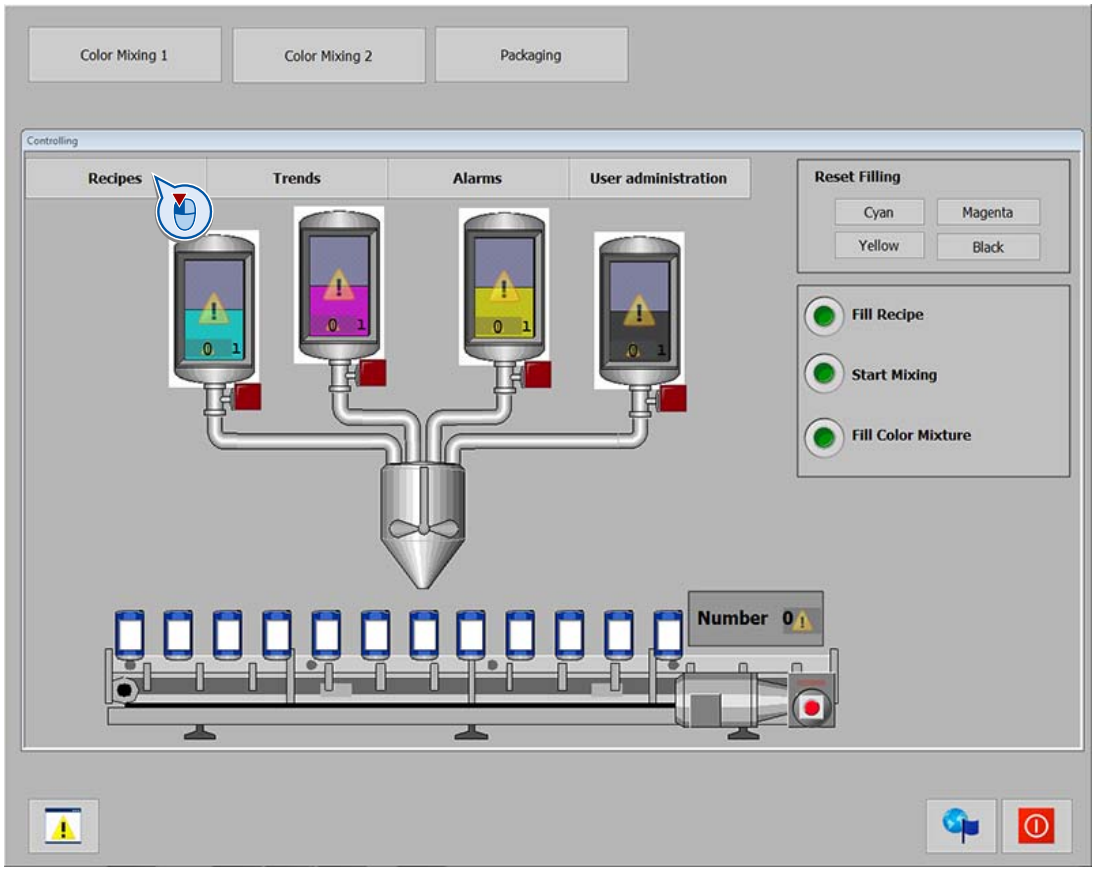
3. Check the start screen display and go to user administration.



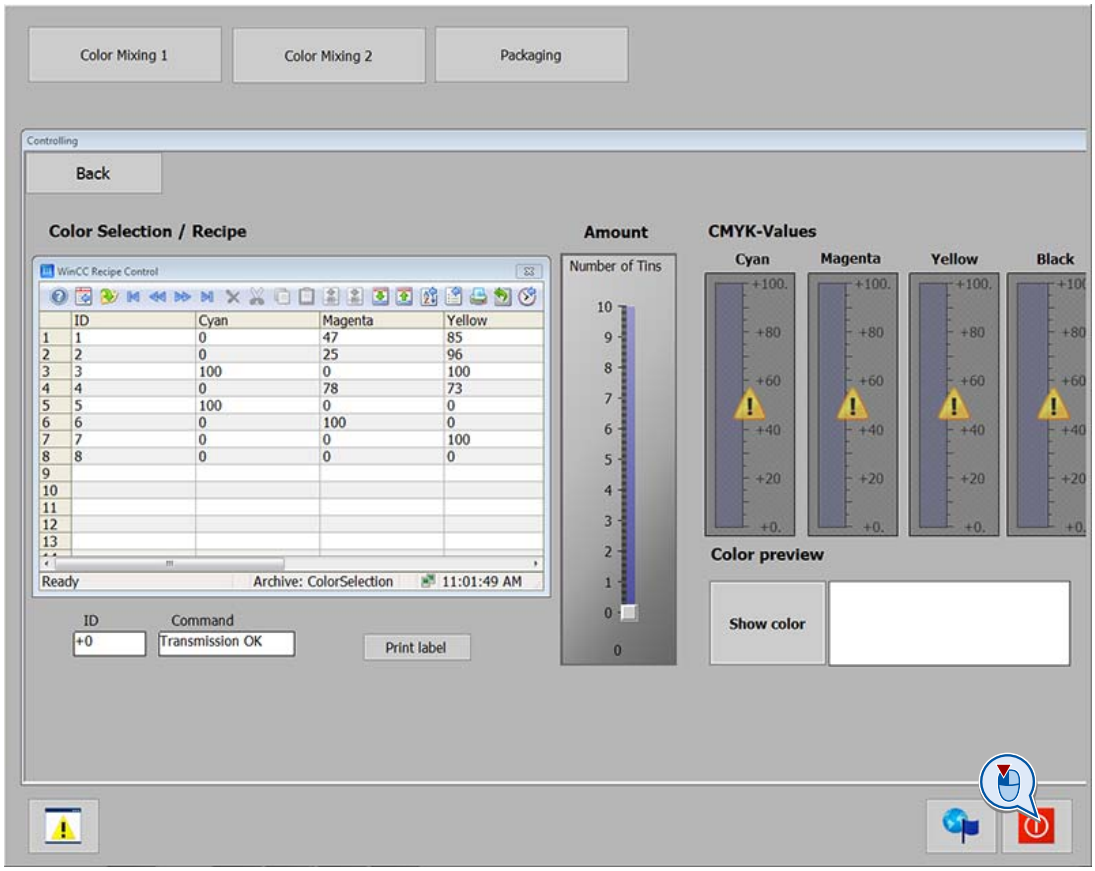
4. Open the logon dialog (1) and log on as user "Lopez" (2, 3). Return to the start screen (4).



5. Change to the recipe selection.



6. Check the recipe selection display and end runtime.



Result

The project has been configured without errors and can be run in runtime.

11.3 Testing the project with a controller

11.3.1 Compiling the control program

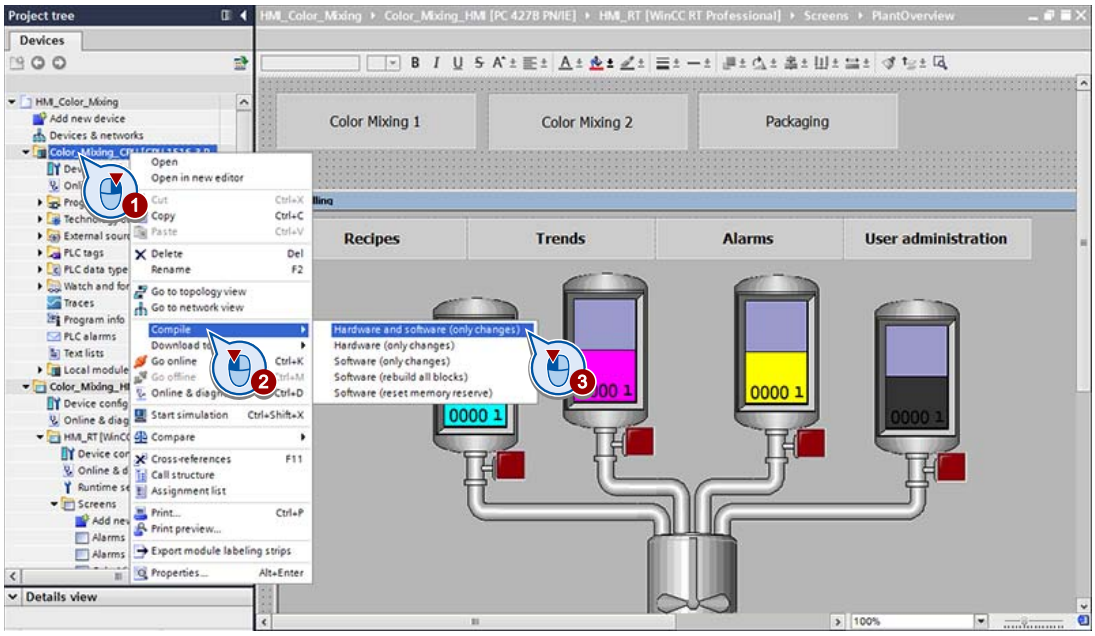
Introduction

To test the project with controller connection in runtime, load the control program to the controller and then start the runtime simulation.

The program is compiled in WinCC beforehand so that it can be read by the controller. The control program can be loaded only under the condition that the compilation has been completed successfully.

Procedure

1. Start the complete compilation of the control program.



Result

The compilation result is displayed in the inspector window.

The control program has been compiled without errors and can be loaded to the controller.

11.3.2 Loading the control program to the controller

Introduction

In this section, you will load the compiled control program to the controller. During the loading process, an online connection is established to the controller. You will specify the corresponding connection and interface parameters for this.

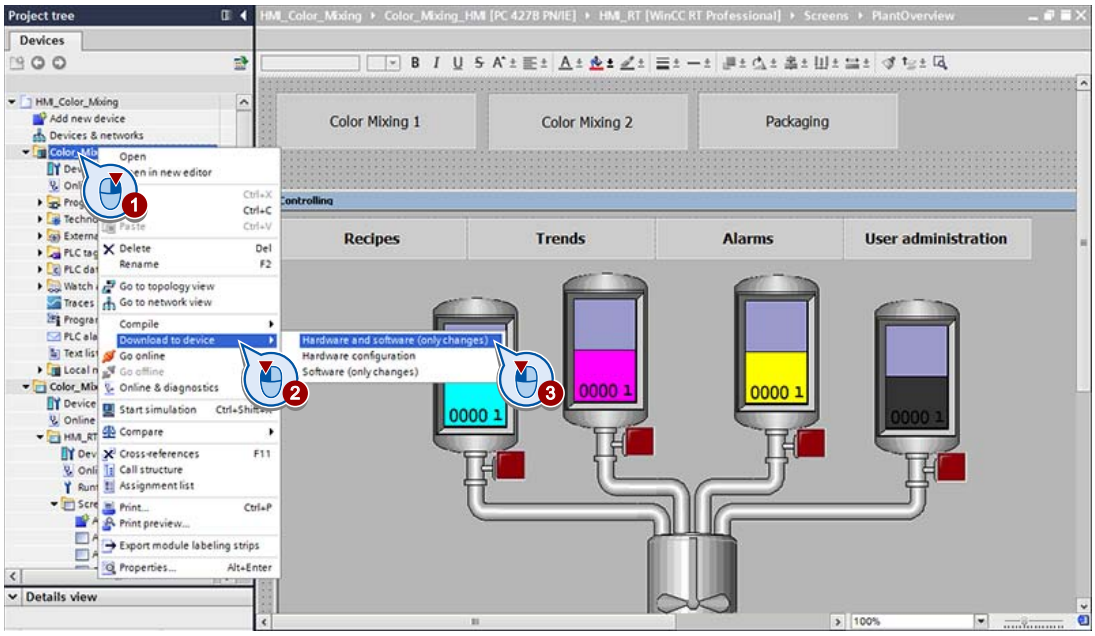
The connection and interface parameters of your plant are shown in the "Welcome" section of this Getting Started.

Requirement

- The controller is in "RUN" mode.

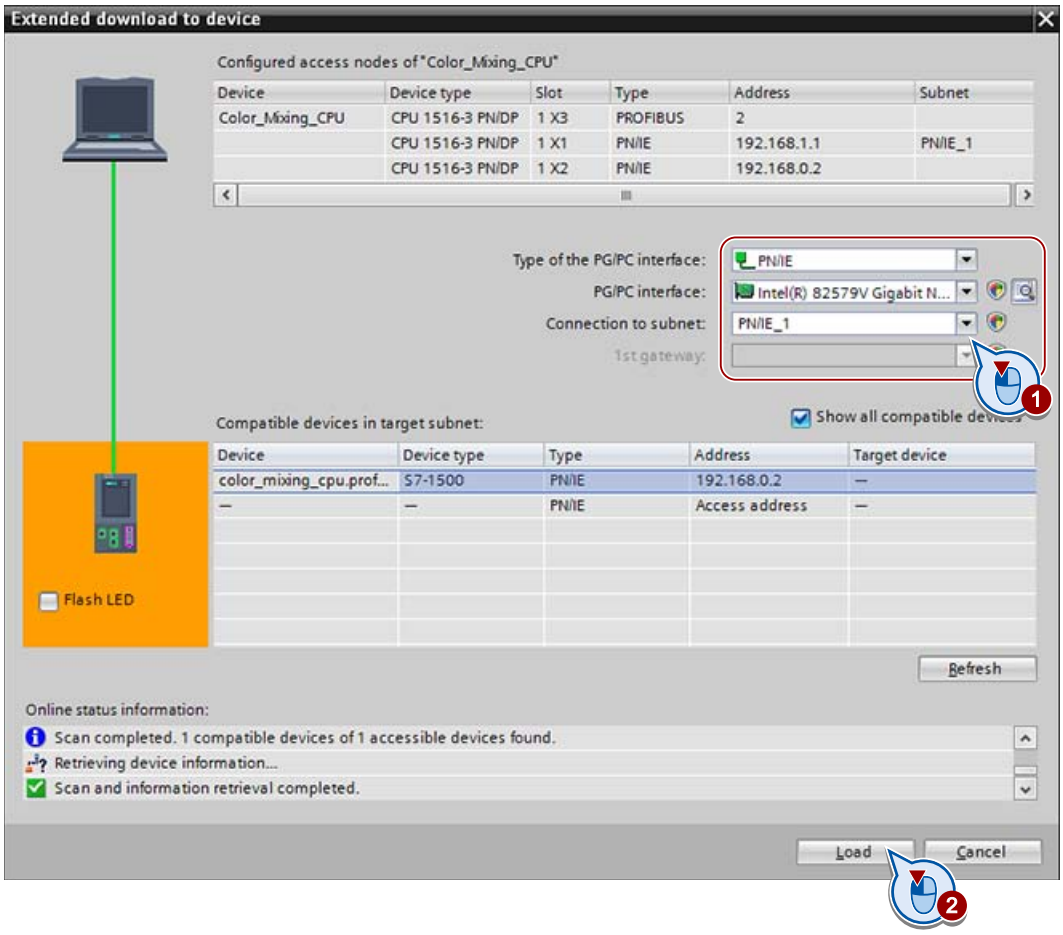
Procedure

1. Start the load process of the controller program.



2. Specify the connection parameters in the "Extended download" dialog.

   Once the online connection is set up, a solid green line is shown between the configuration PC and the controller in the dialog and the PLC is shown with a colored background. Only then can the load process be continued.



Note

The "Extended download" dialog is only opened automatically for the first load process. From the second time, the previous settings are taken over. The menu command "Online > Extended download to device..." can be used to access the "Extended download" dialog at any time.

3. Check your settings in the "Load preview" dialog and continue the load process.
4. Check in the subsequently displayed "Load results" dialog whether the load process was successful and complete the load process.

Result

The control program has been loaded to the controller.

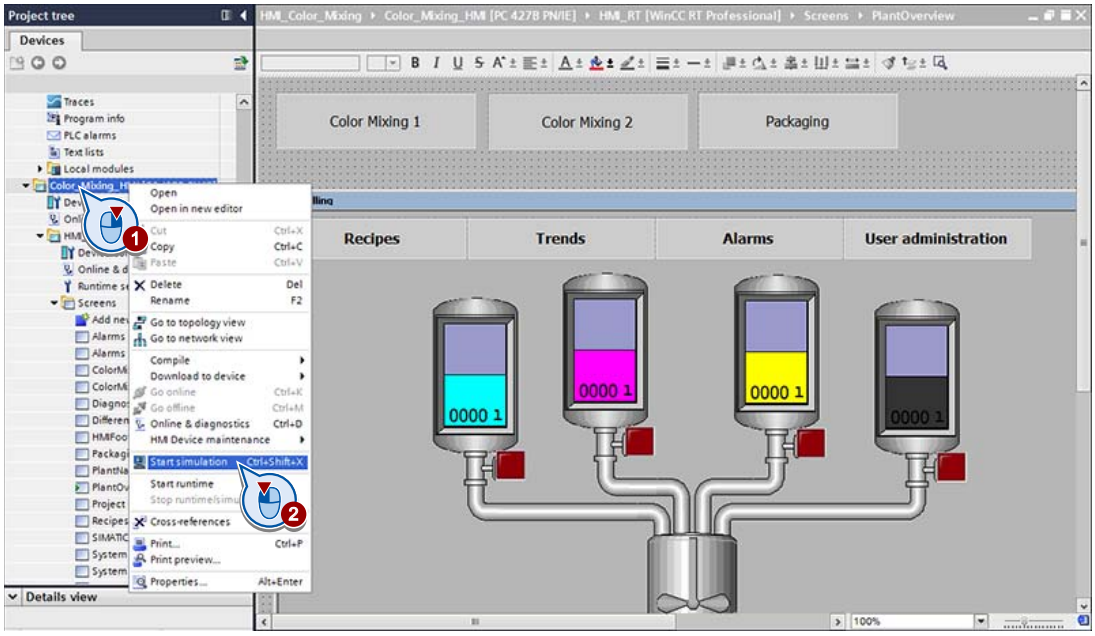
11.3.3 Testing a project

Introduction

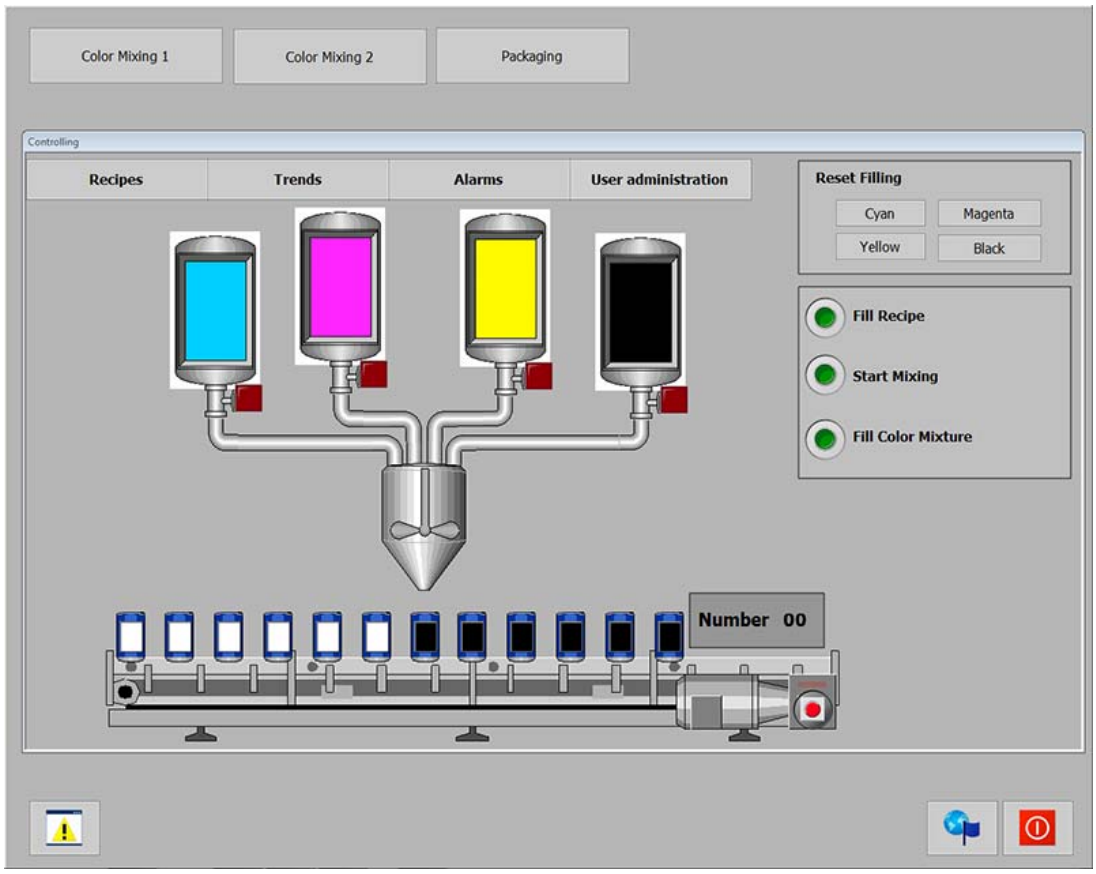
In this section, you will test the production and filling of a color mixture. To this end, create a recipe data record in runtime and initiate the corresponding actions in the start screen.

Procedure

1. Start the runtime simulation on the configuration PC.

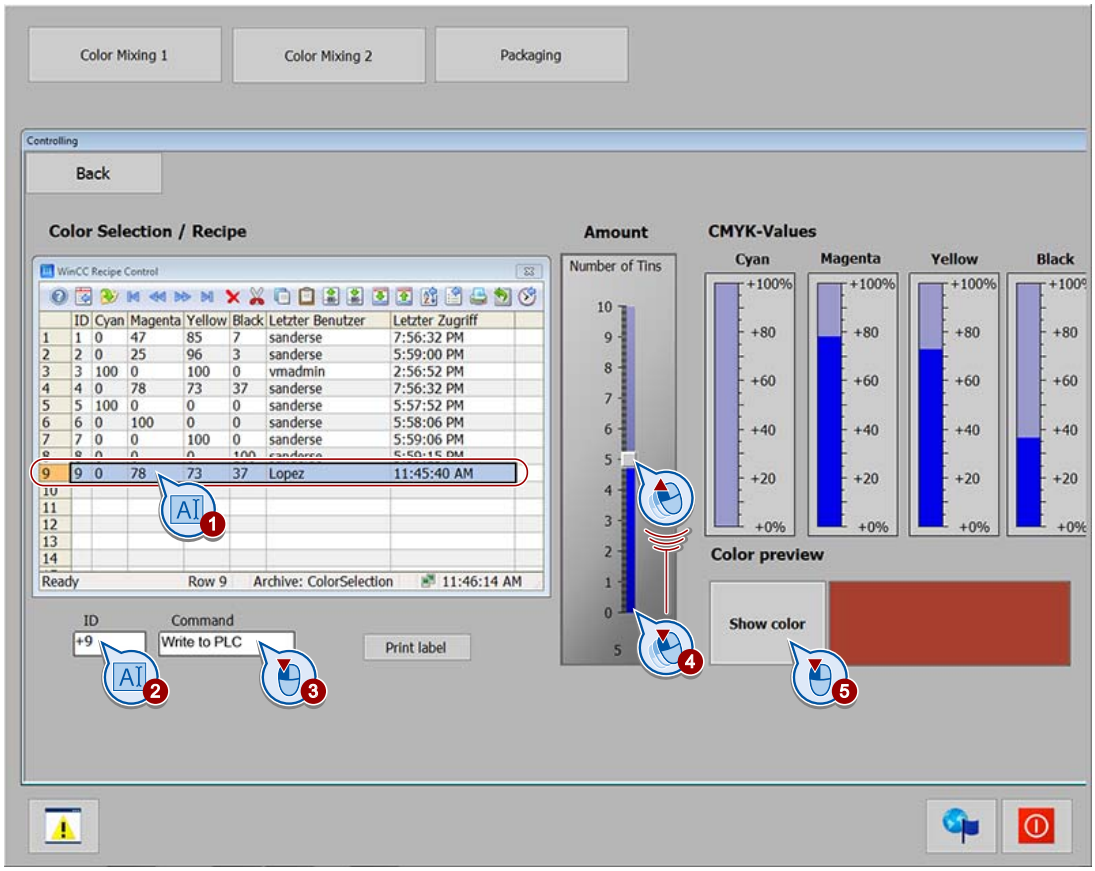


2. Move the alarm window to the background and go to the recipe selection.



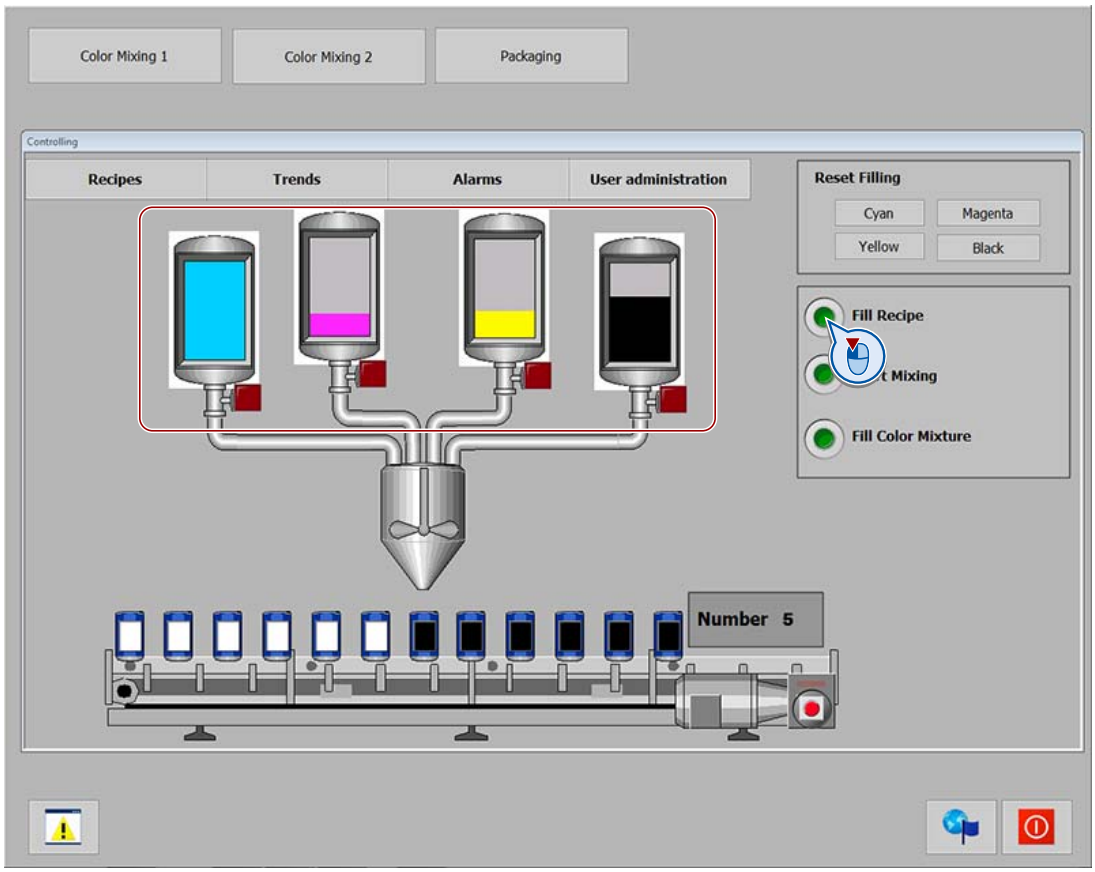
3. Enter a recipe data record (1) in the recipe view.
4. Select the new data record (2) and the command "Write to PLC" (3).

5. Specify the number of color cans to be filled (4) and view the selected color mixture so you can check it (5).



6. Test the production and filling of the new color mixture:
   – Initiate the filling of the color components into the filling tank.

     The fill levels of the reservoirs change.



   – Initiate the mixing of the color in the filling tank (1).

     The mixer rotates.
   – Initiate the filling of the color mixture (2).

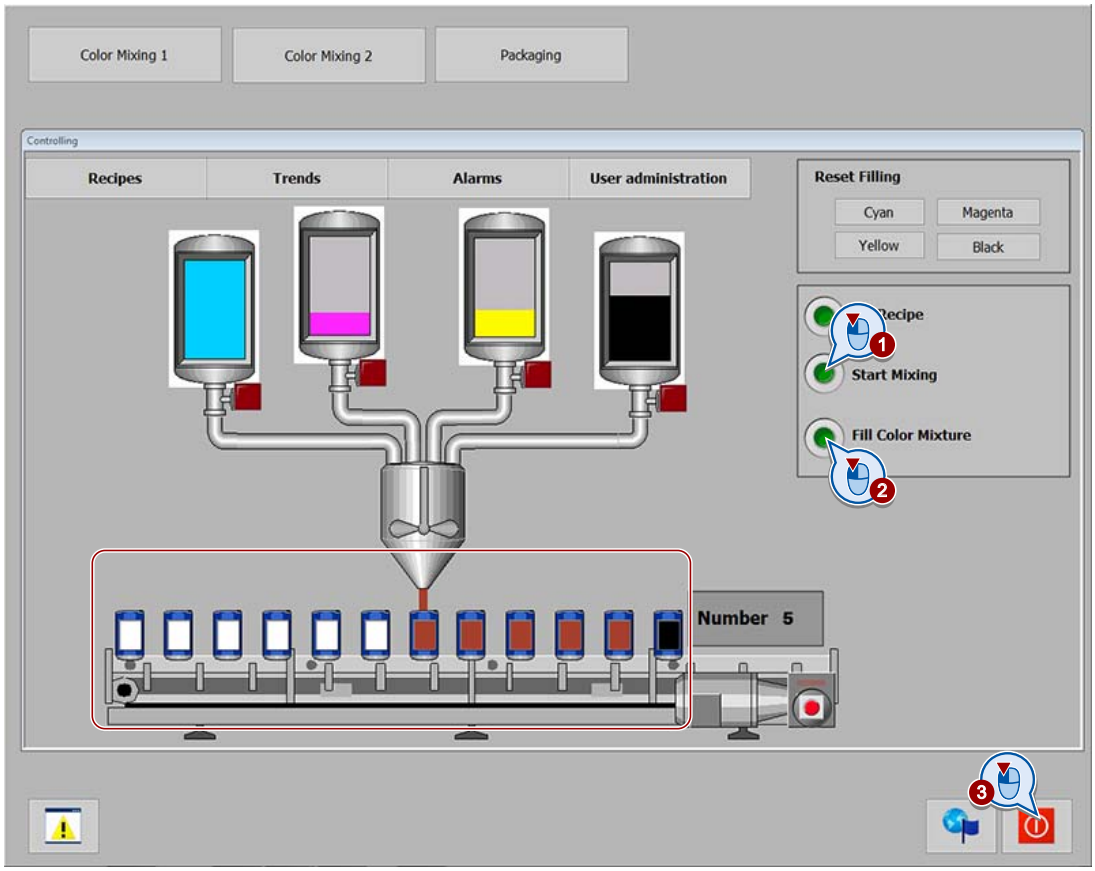
     The filling valve opens.

     A tin is filled and the valve closes again.

     The conveyor moves by one tin to the right.

     The filling valve opens again, and so on.

7. Check the program sequence and end runtime (3).



Result

The WinCC project is tested in connection with the controller. The project can now be loaded to the HMI device.

11.4 Loading the project to the HMI device

Introduction

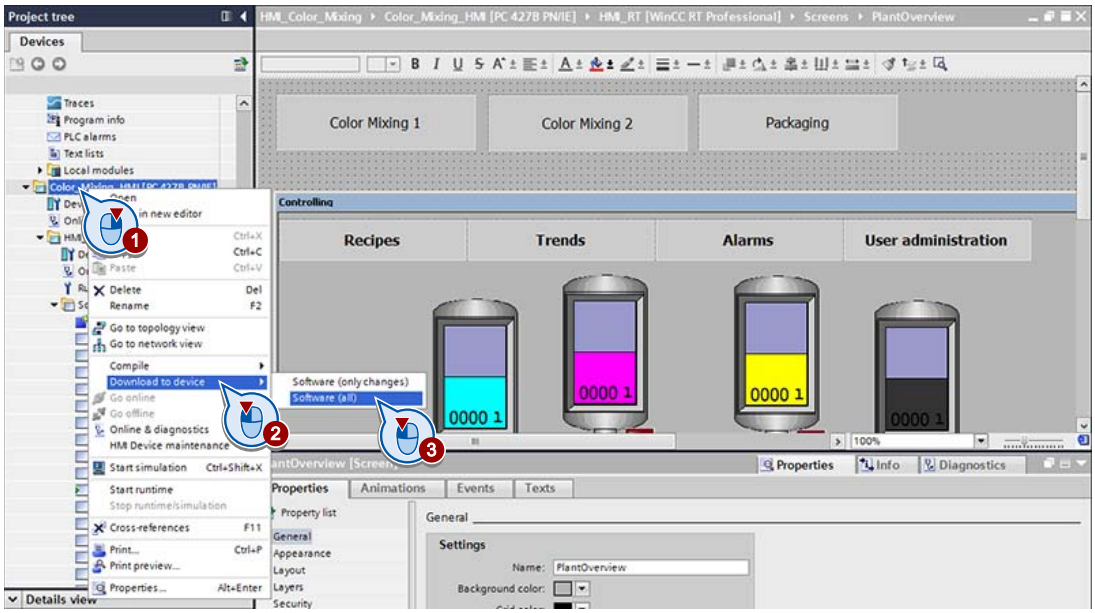
You have successfully tested the project with a connection to the controller on the configuration PC. In the following section, you will load the project from WinCC to the HMI device. You define the the connection parameters according to the plant configuration for loading.

The connection and interface parameters of your plant are shown in the "Welcome" section of this Getting Started.

No settings are required on the HMI device.

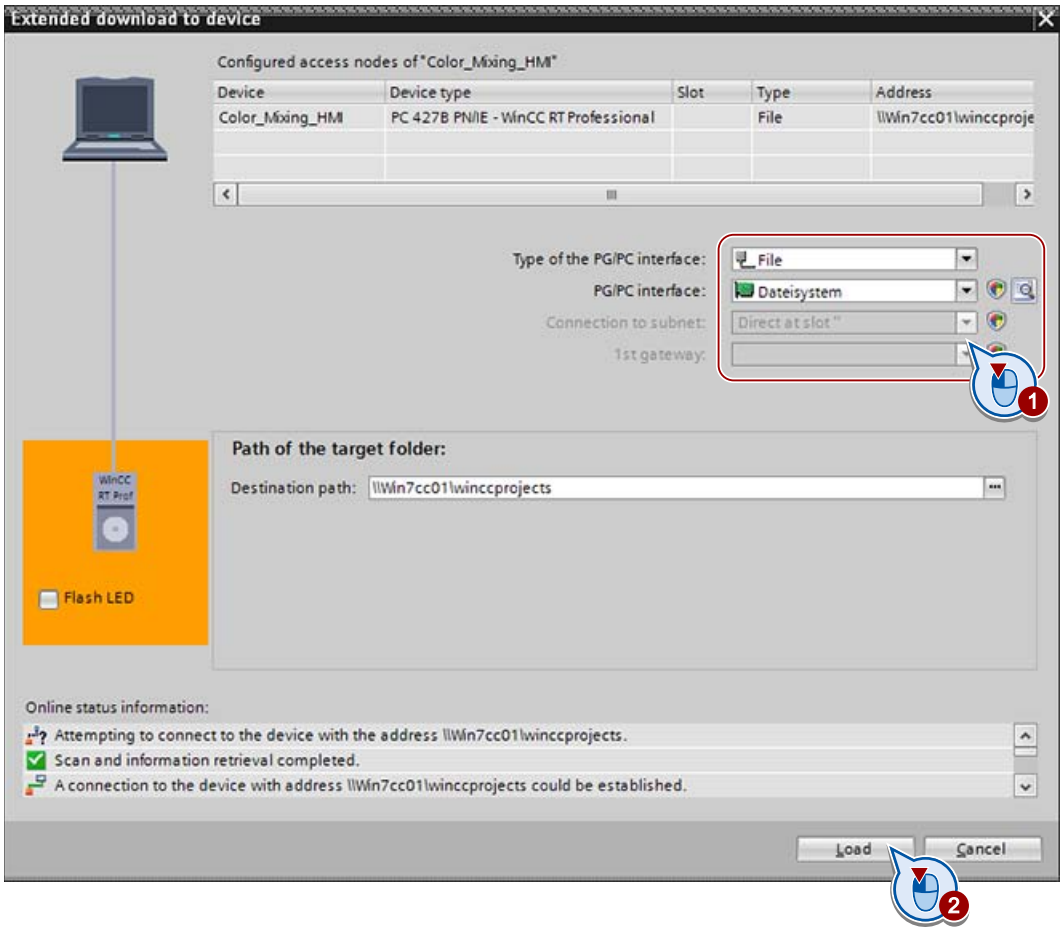
Procedure

1. Start the complete loading of the project.



2. Specify the connection parameters in the "Extended download" dialog.

   Once the connection is set up, a solid green line is shown between the configuration PC and the controller in the dialog and the HMI device is shown with a colored background. Only then can the load process be continued. The project data is re-compiled during the load process.



3. Check the defaults in the displayed "Load preview" dialog and complete the load process.

Note

The "Extended download" dialog is only opened automatically for the first load process. From the second time, the previous settings are taken over. The menu command "Online > Extended download to device..." can be used to access the "Extended download" dialog at any time.

All the data is loaded to the HMI device only for the first load process. During subsequent load processes, only changes are loaded.

Result

The project is loaded to the HMI device and can be run in runtime.

12 Summary

This Getting Started was an introduction to the configuration options of Runtime Professional.

You learned how to create screen objects you can use multiple times with faceplates and how to use the screen windows method to switch between plant parts.

You have configured alarms, logs and recipes and learned how to use scripts to dynamize the user interface in runtime.

An introduction to user administration and the creation of a multilingual user interface conclude this Getting Started with a successful test of the configuration.