

TM210

Working with Automation Studio



Prerequisites and requirements

Training modules	–
Software	Automation Studio 4.2 LTS Automation Runtime 4.25
Hardware	X20 control and X20 I/O modules

Table of contents

1	Introduction.....	4
1.1	Learning objectives.....	4
1.2	Safety notices and symbols.....	5
2	Installation and registration.....	6
2.1	Installation wizard.....	6
2.2	Automation Studio licensing.....	8
3	Your first project.....	9
4	Automation Studio.....	10
4.1	Basic concept of Automation Studio.....	10
4.2	"Coffee machine" sample project.....	11
4.3	Automation Studio help system.....	13
4.4	The workspace.....	15
5	Software, hardware and configurations.....	21
5.1	Software management in the Logical View.....	21
5.2	Hardware organization - Physical View and System Designer.....	21
5.3	Organizing configurations in the Configuration View.....	22
5.4	Assignment of programs - Software configuration.....	23
6	Configuring the hardware.....	26
6.1	Testing in simulation mode.....	28
6.2	Adding I/O modules from the Hardware Catalog.....	30
6.3	Mapping process variables to I/O channels.....	31
6.4	Editing an I/O configuration.....	32
6.5	Configuring network interfaces.....	33
6.6	Build the project.....	34
7	Initial installation of the controller.....	35
7.1	Connecting to the target system.....	36
7.2	Transfer Automation Runtime.....	39
7.3	Project installation.....	41
7.4	Offline installation via data storage device.....	43
8	Programming.....	47
8.1	Programming languages.....	47
8.2	Initialization and cyclic subroutines.....	47
8.3	Variables and data types.....	48
8.4	Creating an application.....	50
8.5	Import, export and team functions.....	51
9	Updates and licenses.....	53
9.1	Hardware and software upgrade.....	53
9.2	Technology Guarding.....	54
10	Summary.....	55

Introduction

1 Introduction

Automation Studio is the project development environment used specifically for B&R automation components. This includes controllers, motion control components, safety modules and HMI applications. Clearly organized project structuring options and the ability to manage multiple configurations ensure that teams can work together efficiently and all machine variants can be represented in a single project.

Users are able to choose from a wide range of programming languages, diagnostic tools and editors to assist them at every stage of engineering. Standard libraries provided by B&R and the integrated IEC programming languages allow for a highly efficient workflow. Extensive simulation options enable applications to be configured and tested independently of the hardware.

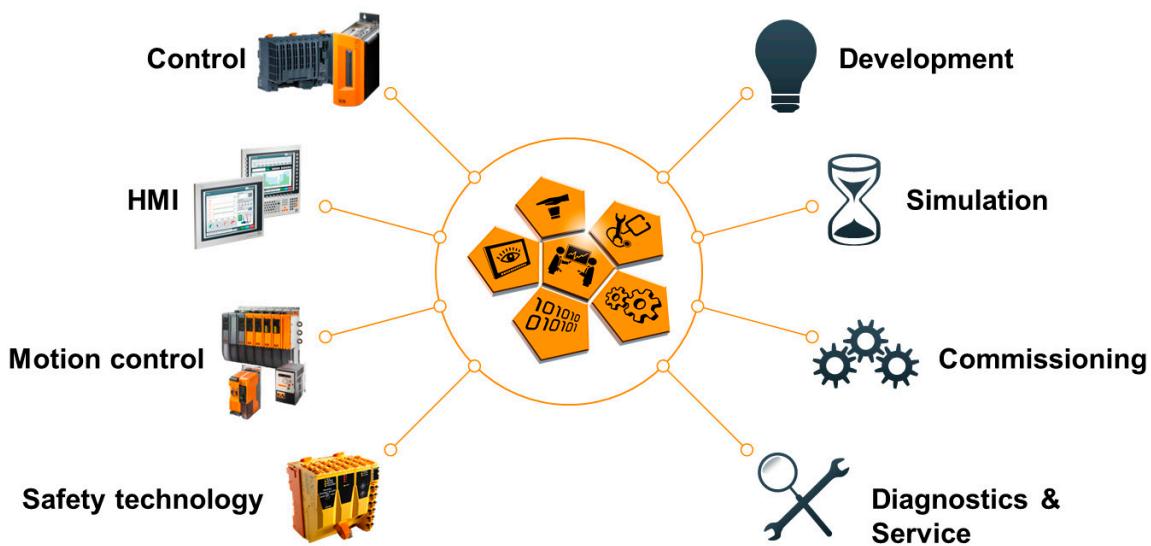


Figure 1: Automation Studio: one engineering tool for the machine's entire lifecycle

1.1 Learning objectives

This training module uses simple exercises to demonstrate the wide range of tools available in Automation Studio. In addition, it will frequently refer to the extensive Automation Studio help system, an invaluable reference for completing the exercises in this training module.

- You will learn how to create and configure projects in Automation Studio.
- You will learn how to develop small programs and declare process variables.
- You will learn how to navigate the Automation Studio help system.
- You will learn how to set up hardware configurations and use the simulation features in Automation Studio.
- You will learn the steps to follow in order to commission a B&R controller.
- You will learn how to use the Automation Studio user interface and the various editors available.
- You will learn about the options for configuring modules with Automation Studio's hardware management features.

1.2 Safety notices and symbols

Safety notices in this manual are organized as follows:



Danger: Disregarding these safety guidelines and notices can result in severe injury, death or substantial damage to equipment.



Warning: Disregarding these safety guidelines and notices can result in severe injury or substantial damage to equipment.



Caution: Disregarding these safety guidelines and notices can result in injury or damage to equipment. This information is important for preventing errors.

Additional notices and information in this manual are organized as follows:



Note: Provides important tips and additional information



Help: References additional documentation (help system, data sheets, user's manuals)



Example: Indicates an example to further illustrate a topic



Results: A brief summary of the results of an exercise

Organization of safety notices in external manuals

This manual contains references to other manuals. How safety notices are organized in external manuals is listed in the respective manual.

Exercise: Tasks and exercises

Sections marked with an orange stripe on the left side contain information about exercises as well as the associated actions to be taken. Exercises are intended to provide a deeper understanding of the information provided.

Installation and registration

2 Installation and registration

Automation Studio is installed automatically when the Automation Studio DVD is inserted into the computer's DVD drive. It is also possible to start the installation manually by running the **Install.exe** file. Once Automation Studio has been installed, it needs to be registered.

2.1 Installation wizard

The installation wizard guides the user through the installation process. After selecting the language for installation, a number of installation options are available. These include viewing version information, calling system requirements, etc.

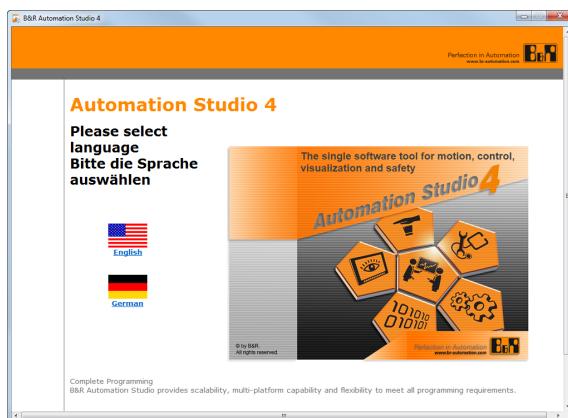


Figure 2: Automation Studio installation: Selecting the installation language

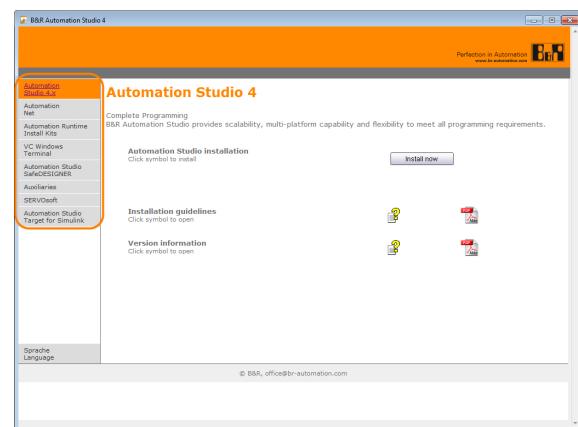


Figure 3: Automation Studio installation: Installation options and version information

Two different installation methods are available:

- Standard installation
This is a normal installation and will install Automation Studio on the computer's hard drive.
- Portable installation
This option can be selected to install Automation Studio on a removable storage device.

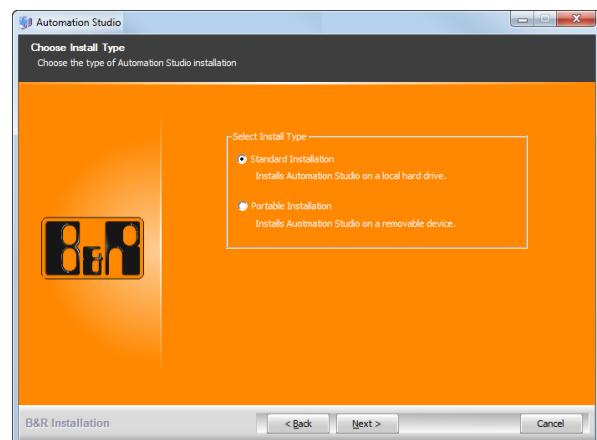


Figure 4: Dialog box for selecting a standard or portable installation



Automation software \ Software installation \ Automation Studio

2.1.1 Starting Automation Studio

The Automation Studio installation automatically adds the necessary shortcuts to the Start menu and desktop in Windows. It is then possible to launch Automation Studio from the Windows Start menu or using the shortcut on the desktop.



Figure 5: Automation Studio desktop icon

When Automation Studio is started for the first time, the Automation Studio start page is displayed.

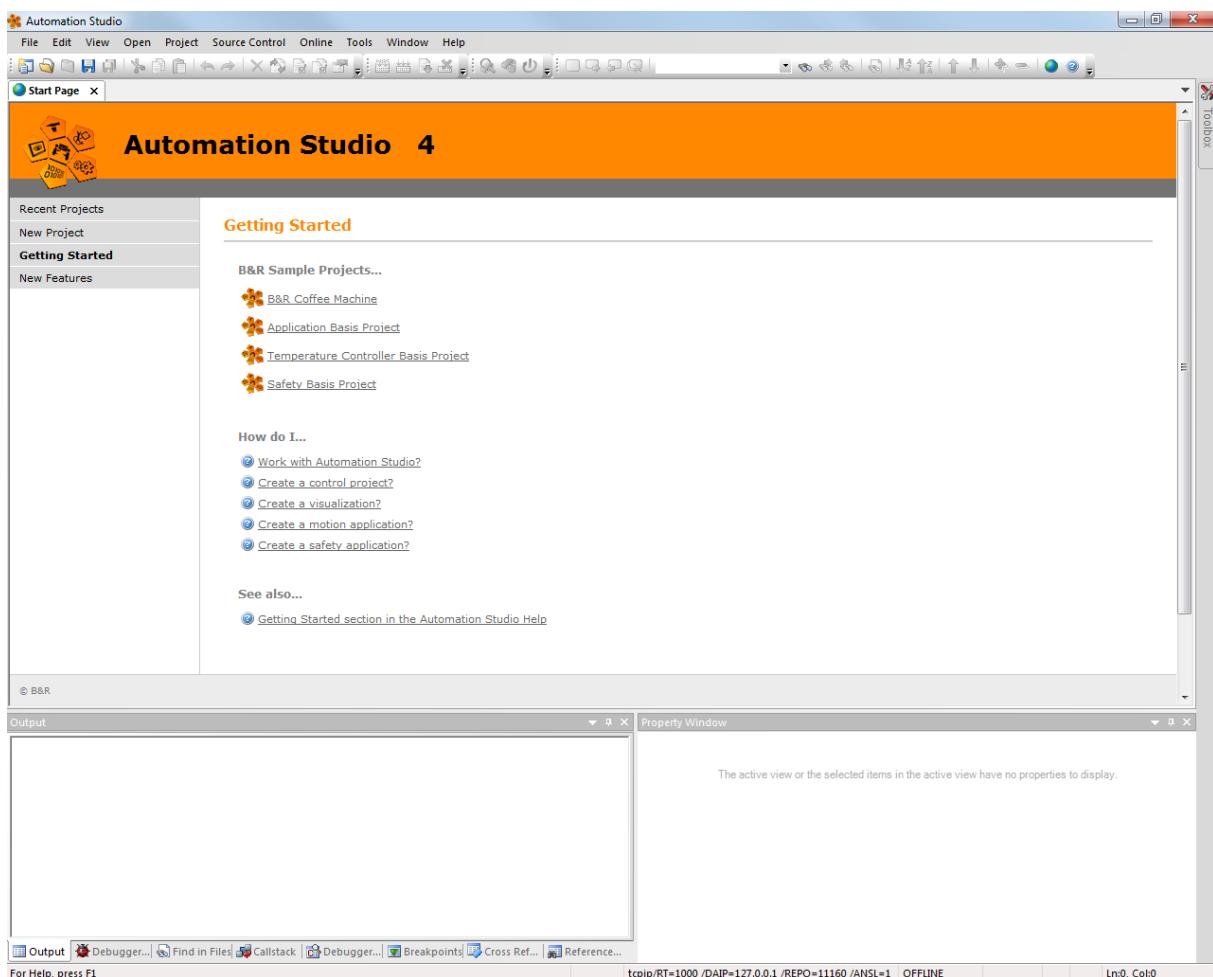


Figure 6: The Automation Studio start page

From the start page, you can create a new project or open an existing one.

It also includes the "Getting started" section for users to learn more about working with Automation Studio. The start page can be opened at any time by selecting <Help> / <Show start page> from the main menu; it is also displayed after a project is closed.

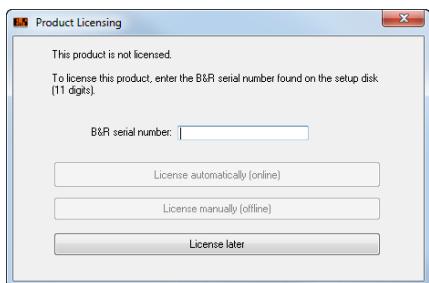
Installation and registration

2.2 Automation Studio licensing

Once Automation Studio has been installed, it needs to be licensed.

This process is started by entering the 11-digit serial number of the software in the licensing window. The serial number (beginning with 8986 or A555) can be found in the software's license agreement.

The licensing window is displayed each time Automation Studio is started (until registered, of course) and can also be opened from the "About Automation Studio" window.



Once a correct serial number has been entered in the licensing window, the license can be activated online if the workstation is connected to the Internet.

If a direct connection to the Internet is not available, the process can be completed manually by visiting the B&R website (www.br-automation.com in the area **Service → Software Registration**) from another computer with Internet access.

Figure 7: The Automation Studio licensing window



The full range of functions are available in Automation Studio for 30 business days after installation. After this time, the program is locked until the software is registered.

The licensing window also appears when you open SafeDESIGNER. Here you should enter the serial number from the Automation Studio license agreement.



Automation software \ Software installation \ Automation Studio \ Licensing

3 Your first project

In this section you will use the Automation Studio help system as a reference to create a new project, transfer it to Automation Runtime Simulation and test it using Automation Studio.

Exercise: Create a new project using the help system as a reference

The corresponding section of the help system can be opened directly from the start page in Automation Studio.

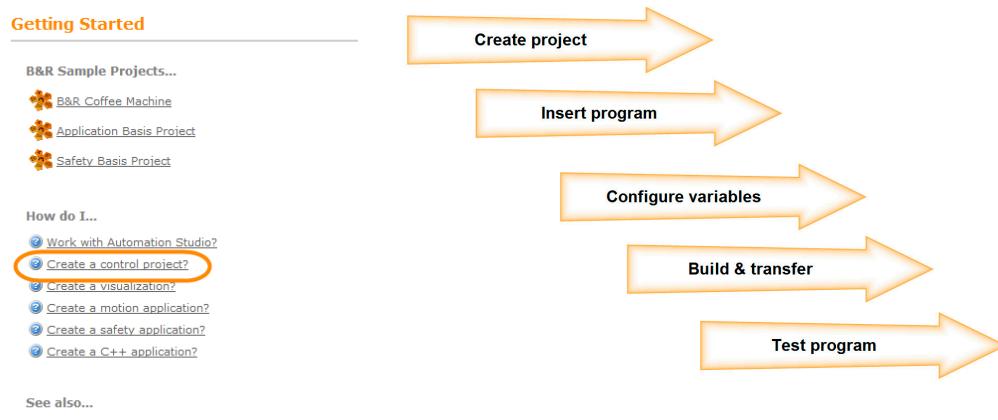


Figure 9: Steps for creating your first project

Figure 8: Automation Desktop home screen

- 1) Click on the **How do I create a control project?** link on the start page.
- 2) Select the first sub-section "First Automation Studio project with Automation Runtime Simulation"
- 3) Work through each of the steps.



The help system is a useful aid for creating a new project, designing a program and then transferring the project to the Automation Runtime Simulation where it can be tested. Some of the elements of Automation Studio have already been introduced during the course of this first project. The next few sections will provide additional information about the structure of an Automation Studio project with the help of a sample project.



The language of the help system can be set by selecting **<Tools> / <Options>** from the main menu (English or German).



Automation software \ Getting started \ Creating programs in Automation Studio \ ARsim example project

Automation Studio

4 Automation Studio

This section will use an sample project to explain a bit more about Automation Studio and the structure of projects.

4.1 Basic concept of Automation Studio

In an Automation Studio project, the software is structured in the Logical View. The machine structure and machine functions determine what the structure looks like. This direct relationship between the application and the machine makes the structure of the software intuitive and easy to navigate.

Completed software components are assigned to their respective configurations in the Configuration View. For each configuration, there is a corresponding hardware configuration that is managed in the Physical View.

Automation Studio allows automation projects to be structured modularly to support distributed development approaches and make working in project teams more efficient.

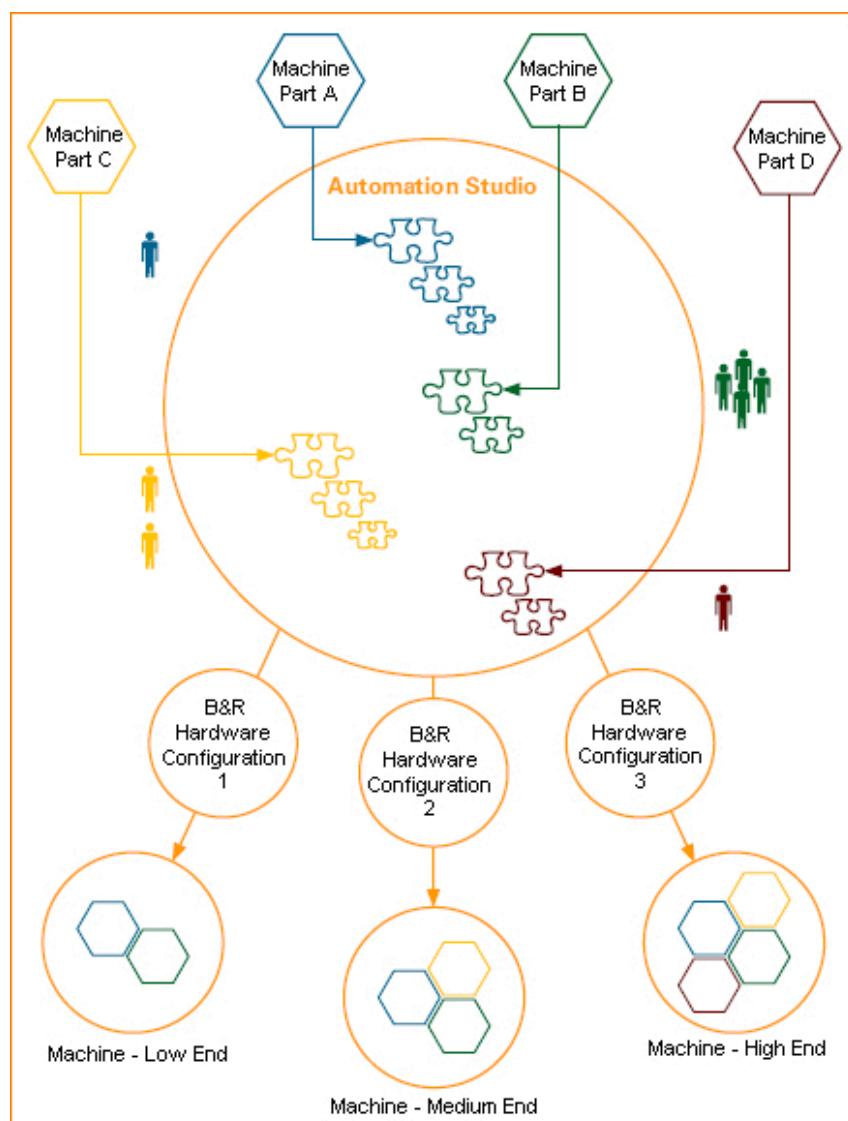


Figure 10: The structure and basic principle of Automation Studio

4.2 "Coffee machine" sample project

Automation Studio comes with a number of sample projects. This training module will use the "CoffeeMachine" sample project to explain the Automation Studio environment and some of the many different functions available. The "CoffeeMachine" sample project can be used to illustrate the path from an initial design concept to a full representation of the software and hardware components in Automation Studio.

Exercise: Open the "CoffeeMachine" sample project in Automation Studio

This sample project can be opened from the start page.

- 1) Open the Automation Studio sample project from the start page.



Figure 11: Select the sample project.

- 2) Confirm the location where the sample project will be unzipped.



After following these steps, the "CoffeeMachine" sample project will be unzipped and opened in Automation Studio.

4.2.1 Overview of the machine

The software is structured in Automation Studio in a way that corresponds directly to the structure of the machine and its functional units.

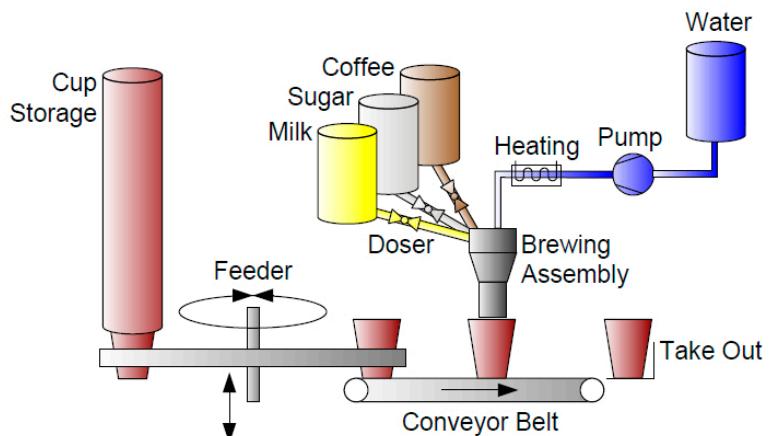


Figure 12: Overview of the system

Automation Studio

The basic process can be outlined as follows:

- The cup extractor takes a cup and places it on the conveyor belt.
- The cup is moved to a defined filling position.
- A dosing unit determines the amount of sugar, milk and coffee needed for the specified type of coffee.
- The water is heated and pumped into the cup.
- The conveyor belt moves the cup to the dispensing position.



Automation software \ Sample programs \ CoffeeMachine

Programming \ Structured software development

4.2.2 Description of function

The software structure is structured according to the coffee preparation process. Each individual function is represented in Automation Studio by clearly structuring the design and programming aspects of the project.

This image shows the Logical View. Each folder (called a package in Automation Studio) represents a logical section of the software structure.

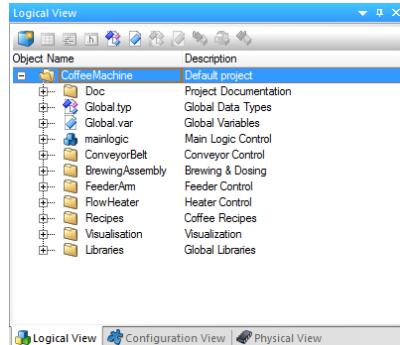


Figure 13: Software management in Automation Studio

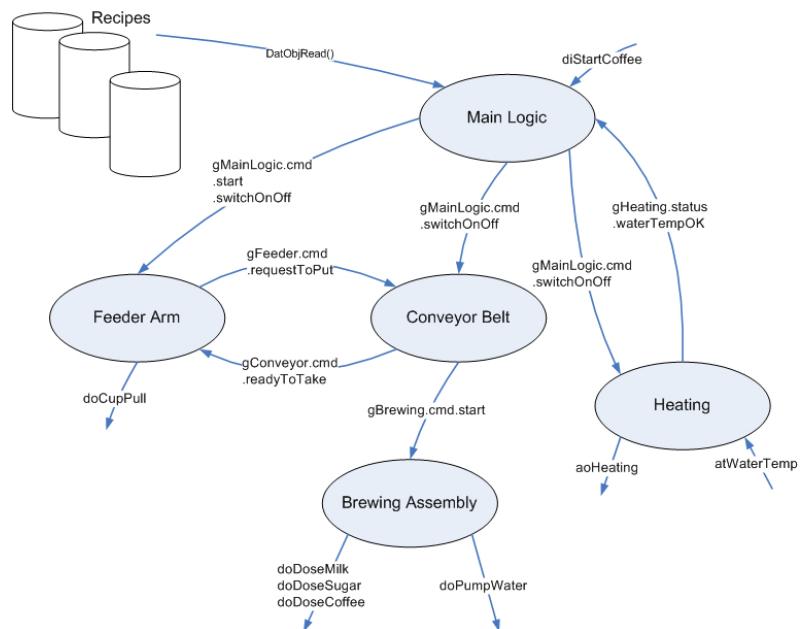


Figure 14: Functional description of the program

4.3 Automation Studio help system

The Automation Studio help system is an invaluable resource throughout the development, configuration and commissioning of a project. It serves as a reference guide for using Automation Studio and its editors, creating programs or HMI applications, configuring drives and much more. In addition, the Automation Studio help system contains access to all B&R hardware documentation. When you install new hardware and software upgrades, the Automation Studio help system is expanded to include the necessary documentation (see "Hardware and software upgrade" on page 53)

Exercise: Use the Automation Studio help system

Press <F1> in Automation Studio to open information in the help system relevant to a selected element. There is also a search function to look for information about a specific topic. The functions of the help system are described in the section "**Automation Software – How do I use the help system**".

Use the Automation Studio help search function to find the technical data for the X20CP1586 controller. Power consumption information can be taken from the technical data for the selected controller.

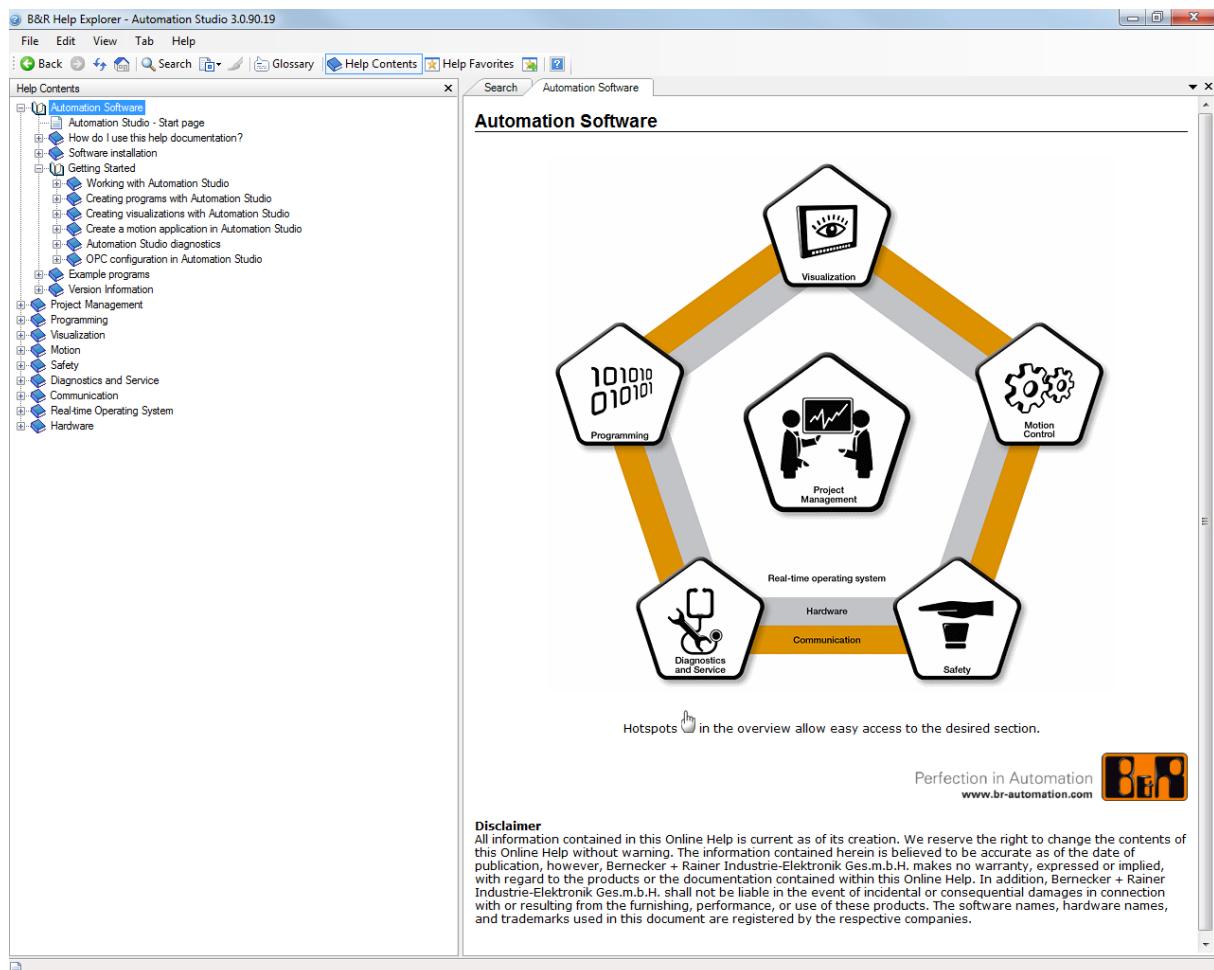


Figure 15: The Automation Studio help system

Automation Studio

4.3.1 Managing help favorites

"Help favorites" make it possible to manage and store helpful pages within an personalized structure. You can save the help favorites to a separate file. In this way, help favorites can be managed as needed and even passed on to other people.

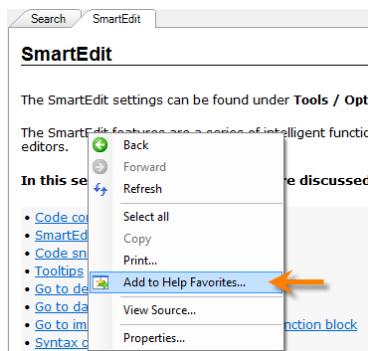


Figure 16: Add a help page using the shortcut menu

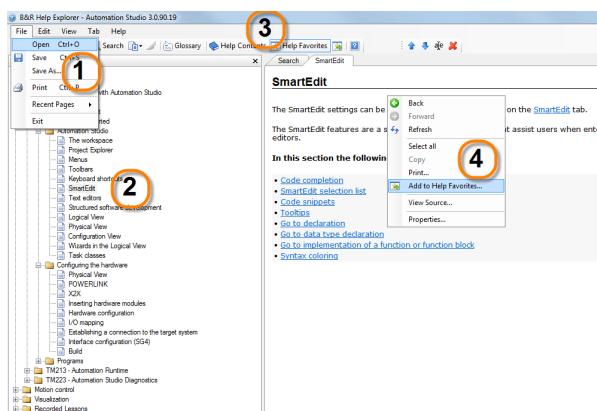


Figure 17: Managing help favorites

Adding help favorites

Add the currently open help page to your help favorites by right-clicking on the document and then selecting "Add to help favorites". This will open a window for selecting where the page should be ordered in the list of favorites.

Managing favorites

1 Saving and opening

Help favorites can be saved to a file by selecting "Save" or "Save as" from the File menu. Any newly added entries are saved automatically.

2 List of help favorites

The help favorites are managed in a tree structure. Entries can be moved and renamed. Clicking on one of the entries will open the corresponding help page.

3 Toolbar

The "Help contents" and "Help favorites" buttons in the toolbar switch between the Favorites view and the navigation tree in the Automation Studio help system. The navigation tree remains synchronized if a favorite is selected in the Favorites view.

Automation software \ How do I use the help system? \ Help favorites

4.4 The workspace

Automation Studio is divided into several different areas, each with a specific function.

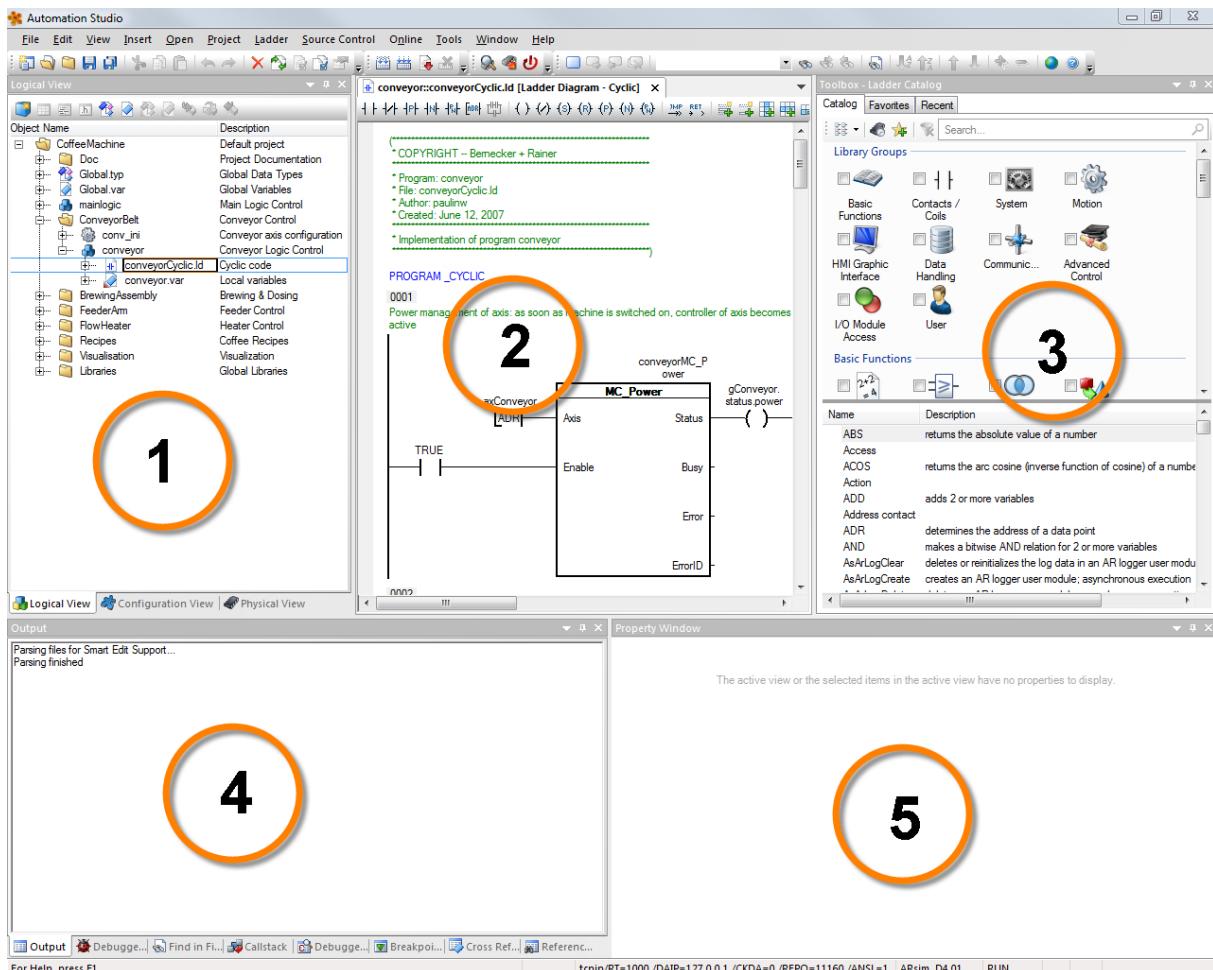


Figure 18: The Automation Studio workspace

- 1 The Project Explorer is located on the left side of the window. This is used to manage and edit software and configuration objects in a project.
- 2 The center of the screen is where open documents are worked on. This is where program code is edited, for example.
- 3 The Toolbox window is located on the right side of the window. Depending on what object you are currently working on, the Toolbox window allows you to select and add hardware modules, program functions or software objects.
- 4 The output window is located on the lower left side of the main window. It is used to display information such as messages that are generated when a project is being built.
- 5 The Properties window is located on the bottom right. This window displays configuration options for whichever object or hardware module is currently selected. It can also be used to edit the properties of selected object.



Project management \ Workspace

Automation Studio

Exercise: Work with Automation Studio

The aim of this exercise is to introduce participants to the Automation Studio workspace.

- 1) Identify the various sections of the workspace.
- 2) Use the context-sensitive help (**<F1>**) to learn about each section.

4.4.1 Reposition the windows

When a project is opened in Automation Studio, the various section windows are all docked to the main program window. Clicking on a window's title bar and dragging it away from the main window will undock it so that it can be moved to another location.

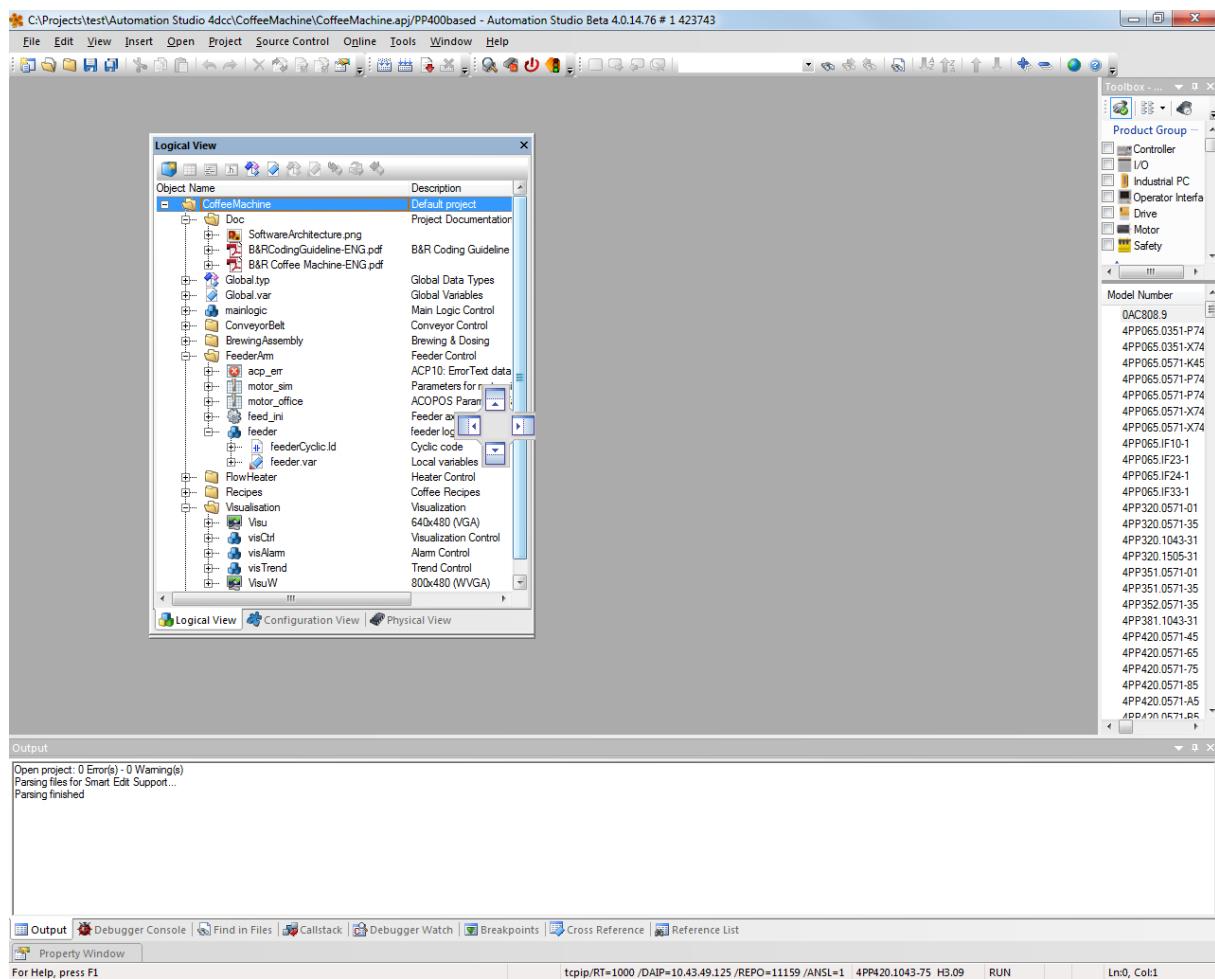


Figure 19: Docking and undocking windows

Project management \ Workspace \ Project Explorer

4.4.2 Auto-hiding windows

To create more space on the screen, for example when working with visual programming editors, the Project Explorer and all other dockable windows can be hidden automatically. Clicking on the pin icon in the title bar of the Project Explorer or any of the other dockable windows will toggle the auto-hide function.



Figure 20: Use the pin symbol to dock or hide windows

 The default window layout in Automation Studio can be restored by selecting <**Window / Reset window layout**> from the main menu.

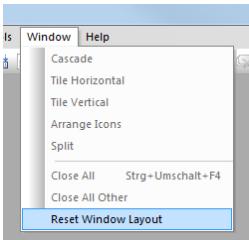


Figure 21: Restoring the default window layout in Automation Studio

 Project management \ Workspace \ Project Explorer

4.4.3 Workbooks

Workbook mode presents a clear way of displaying open windows and a convenient way to switch between them. Windows can be overlapping or arranged above or next to one another.



Figure 22: Workbook

If several editors are open at the same time, each one is displayed in a separate workbook. For an overview of all open workbooks, click on the drop-down symbol on the right side of the title bar.

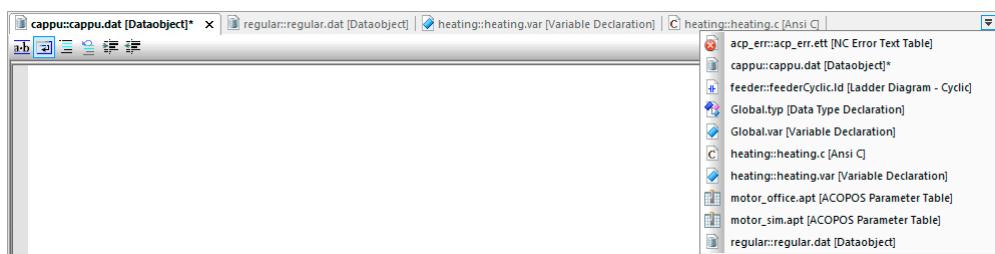


Figure 23: Overview of open workbooks

 Project management \ Workspace

- Workbook mode
- Workbook mode \ Workbook configuration

Automation Studio

4.4.4 Menus and toolbox

All of the functions in Automation Studio can be accessed via menus.

Depending on the context (active editor or window), however, individual menu items may be shown/hidden or enabled/disabled. In this way, only the functions possible within the current context are available for selection.

Toolbox window

The Toolbox window provides access to program and configuration elements, programming functions or hardware modules depending on the editor that is selected.

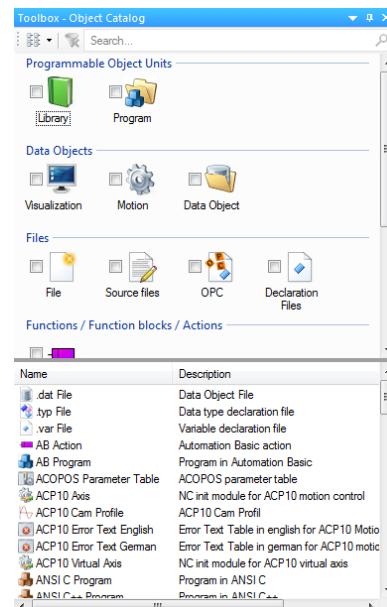
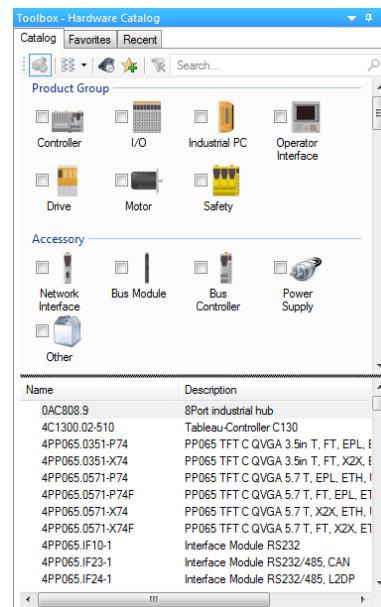
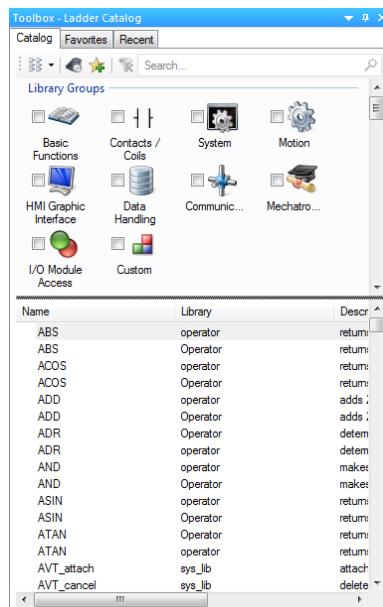


Figure 24: Toolbox - Ladder Diagram: Programming functions, commands, functions and function blocks

Figure 25: Toolbox - Hardware Catalog: Hardware modules and infrastructure components

Figure 26: Toolbox - Object Catalog: Programs and configuration files

Project management \ Workspace \

- Catalogs
- Menus
- Toolbars
- Keyboard shortcuts

Project management \ Hardware management \ Physical View \ Editing operations

4.4.5 The convenience of SmartEdit

The SmartEdit feature combines a range of intelligent functions that provide additional support in Automation Studio editors.

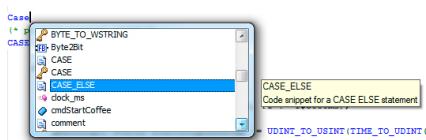


Figure 27: SmartEdit - Code completion

Auto-complete

Pressing <TAB> will automatically complete the entry of some terms. Characters already entered serve as a filter.

This feature is supported by the following elements:

- Variable names, structure elements, constants and enumerators
- Function names and function blocks
- Language constructs (IF THEN, CASE, FOR)

Automatic declaration

Depending on preference, automatic declaration of new variables can be helpful. In SmartEdit settings, you can make numerous custom adjustments to the editors. Open the configuration window by selecting <Tools> / <Options> from the menu. On the "SmartEdit" tab, activate the setting for automatically declaring variables.

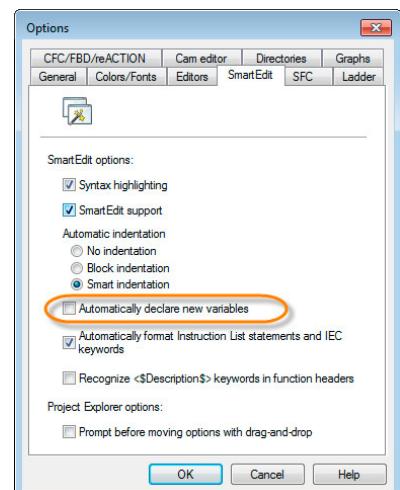


Figure 28: Setting for automatic declaration of new variables

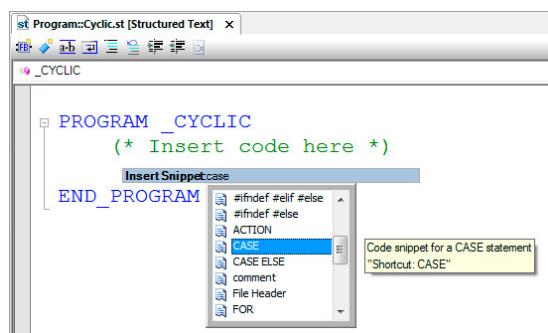


Figure 29: Adding a code snippet

Code snippets

The <CTRL> + <q>, <k> keyboard shortcut can be used to insert code snippets into a program. Code snippets are ready-made bits of source code that can be managed by the user in the code snippet manager.

Additional editor functions

The editors in Automation Studio offer additional editor functions which can help improve your overview of the program code.

- Tooltips
- Syntax coloring
- Coloring for modified lines of code
- Collapse or expand code segments
- Display corresponding parentheses pairs
- Automatic indentation
- Open variable declarations and function implementations directly from the program code

Function	Keyboard shortcut
Complete a code snippet	<TAB>
Auto-complete a variable or function name	<CTRL> + <SPACE>
Open the code snippet selection window	<CTRL> + <q>, <k>
Collapse or expand outlining for all code segments	<CTRL> + <q>, <l>
Collapse or expand outlining for the current code segment	<CTRL> + <q>, <m>
Jump to a variable declaration	<CTRL> + <d>
Jump to the declaration of a variable's data type	<CTRL> + <t>
Jump to a function or function block implementation	<CTRL> + <i>

Table 1: Useful SmartEdit keyboard shortcuts

 Programming \ Editors

- General operation \ SmartEdit
- General operation \ SmartEdit \ Code snippets
- Text editors
- Table editors
- General operation \ Dialog boxes for input support \ Automatic declaration of variables

Project management \ Workspace \ Keyboard shortcuts

5 Software, hardware and configurations

The Project Explorer is a central element of the Automation Studio interface. The Project Explorer contains the Logical View for organizing software, the Physical View for organizing hardware and the Configuration View for managing configurations to represent multiple machine variants in a single project.

5.1 Software management in the Logical View

POUs (program organization units) are arranged in the **Logical View** in a tree structure.

Each of these units is organized into packages, which are comparable to folders or directories. Each package might represent the software elements required for a function unit or system component as well as the associated documentation.

In this view, there is no direct relationship between the software and the actual hardware being used. It only serves to organize and manage the different POU s.

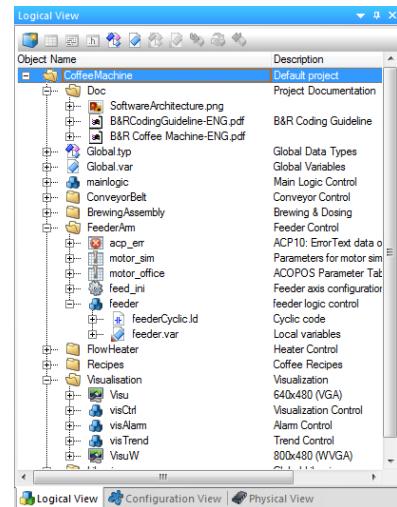


Figure 30: Logical View

5.2 Hardware organization - Physical View and System Designer

The hardware required for the machine can be managed either in a hierarchical view or in a graphical view.

Automation Studio's hardware management features are used to perform the following tasks:

- Insert and configure hardware modules
 - Map variables to I/O data points
 - Configure fieldbus modules and interfaces

Physical View

The Physical View is a hierarchical topology overview of the configured hardware.

When you create a project or a new system configuration (see [6 "Configuring the hardware" on page 26](#)) you select a CPU. Then the required hardware modules are added into the topology from the Toolbox.

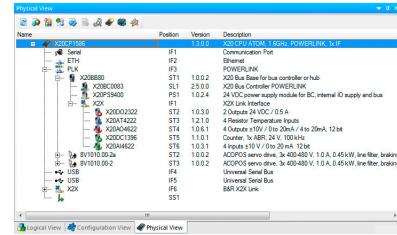


Figure 31: The X20CP1486 CPU in the Physical View

The configuration of individual hardware components and interfaces is opened in Physical View via the shortcut menu. This allows you to make module-specific settings, network configurations and configuration settings for system behavior.

Software, hardware and configurations

System Designer

System Designer provides a visual representation of the Physical View. In System Designer, the control components can be arranged just as they would be used in the actual machine. I/O modules are added and removed in the same way as in the Physical View (see [6.2 "Adding I/O modules from the Hardware Catalog"](#)). The two views are synchronized with each other.

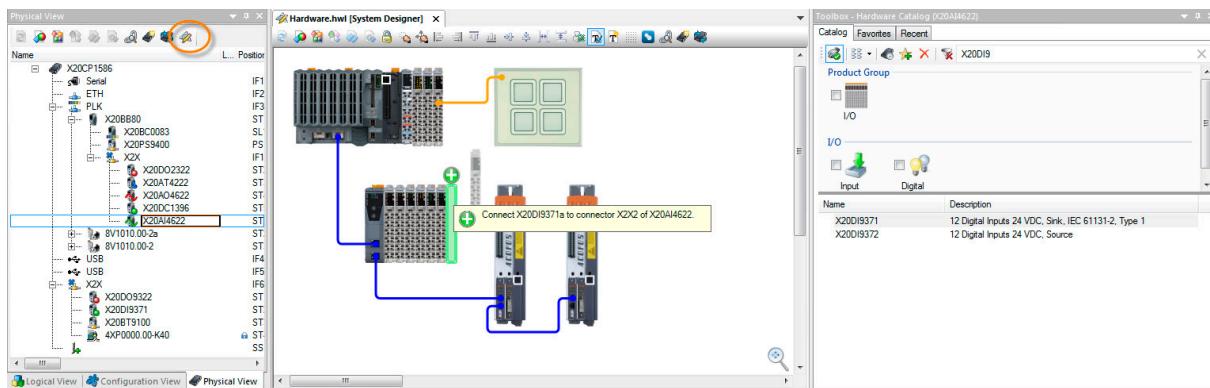


Figure 32: Open System Designer via the toolbar of the Physical View. Add hardware modules from the Toolbox.



The System Designer editor allows objects to be laid out automatically. The ability to add text boxes makes it possible to group hardware components and assign them to a control cabinet, for example.



Project management \ Hardware management

- Physical View
- System Designer
 - View settings \ Automatic layout
 - Editing operations \ Text boxes

5.3 Organizing configurations in the Configuration View

All the different variants of a given machine are managed in the **Configuration View**.

The configurations typically differ in terms of software scope and the exact hardware used.

You can add a new configuration using the toolbox.

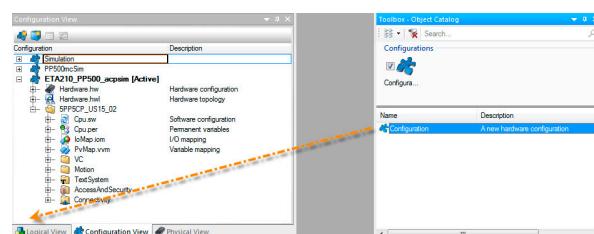
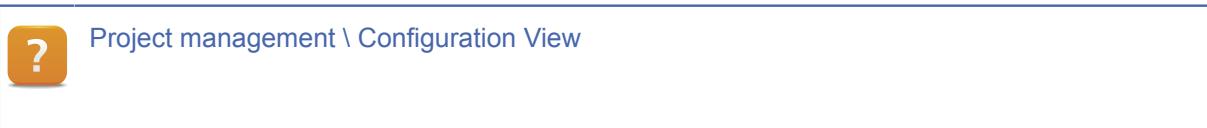


Figure 33: Configuration View with active configuration [Active]. Add a new configuration via drag-and-drop.

When you double-click on a configuration to activate it, the corresponding hardware assigned is displayed in the Physical View as well as in the System Designer. Only one configuration can be **[Active]** at a time.

The following configuration elements are managed, among others, in the Configuration View.

- Software configuration and I/O mapping
- Configuration of user role system, certificates, text system and OPC UA server.
- Configuration of mapp Technology components
- Configuration of motion control



5.4 Assignment of programs - Software configuration

The software elements to be transferred to the target system for the active configuration are assigned in the software configuration.

There are several different ways to assign software to the software configuration:

- ["Automatic assignment when creating a program" on page 24](#)
- ["Adding existing programs manually" on page 24](#)

To open the software configuration in the Physical View, double-click on the controller or select **Software** from the shortcut menu of the controller.

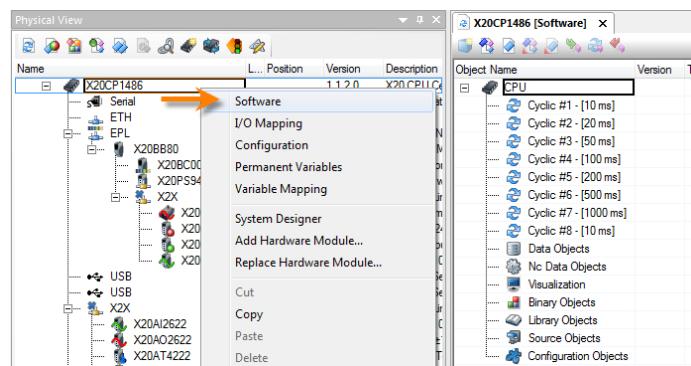
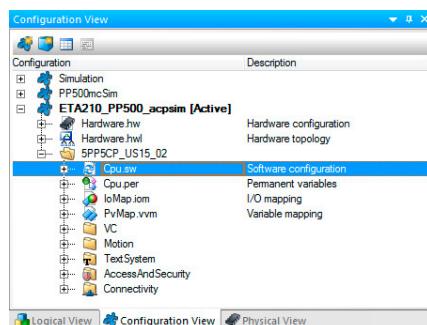
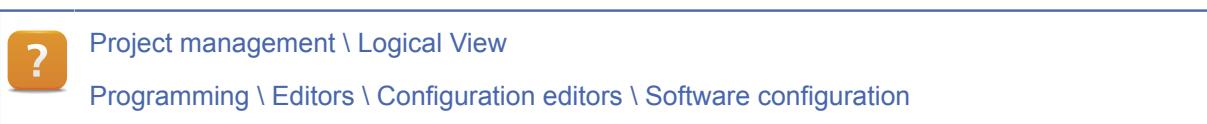


Figure 34: Open the software configuration from the controller's shortcut menu.



Alternatively, the file "CPU.sw" in the Configuration View can be used to open the software configuration.

Figure 35: Open the software configuration by selecting "CPU.sw" in the Configuration View.



Software, hardware and configurations

5.4.1 Automatic assignment when creating a program

If a program is added to the Logical View via the toolbox, then it is automatically included in the software configuration for the active configuration.

Task class #4 is used as the default task class. The default task class is changed in the settings of the active configuration.

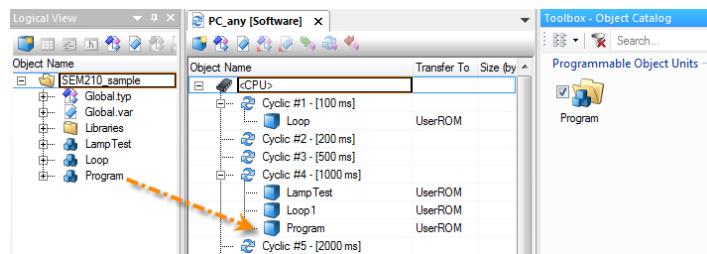


Figure 36: When added, the software object is automatically assigned to the software configuration of the active configuration.



The execution order and priority of programs can be changed by moving the corresponding objects as necessary directly in the software configuration.



Real-time operating system \ Method of operation \ Runtime performance \ Task classes
Project management \ Configuration View \ Properties of the objects in the Configuration View

5.4.2 Adding existing programs manually

If a new hardware configuration is added to the Configuration View, the software elements must be manually assigned to the software configuration.

Open the software configuration, then select the Logical View in Project Explorer. An object can then be assigned by moving it from the Logical View to the desired position in the software configuration.

This procedure makes it easy to assign individually selected programs to a particular configuration.

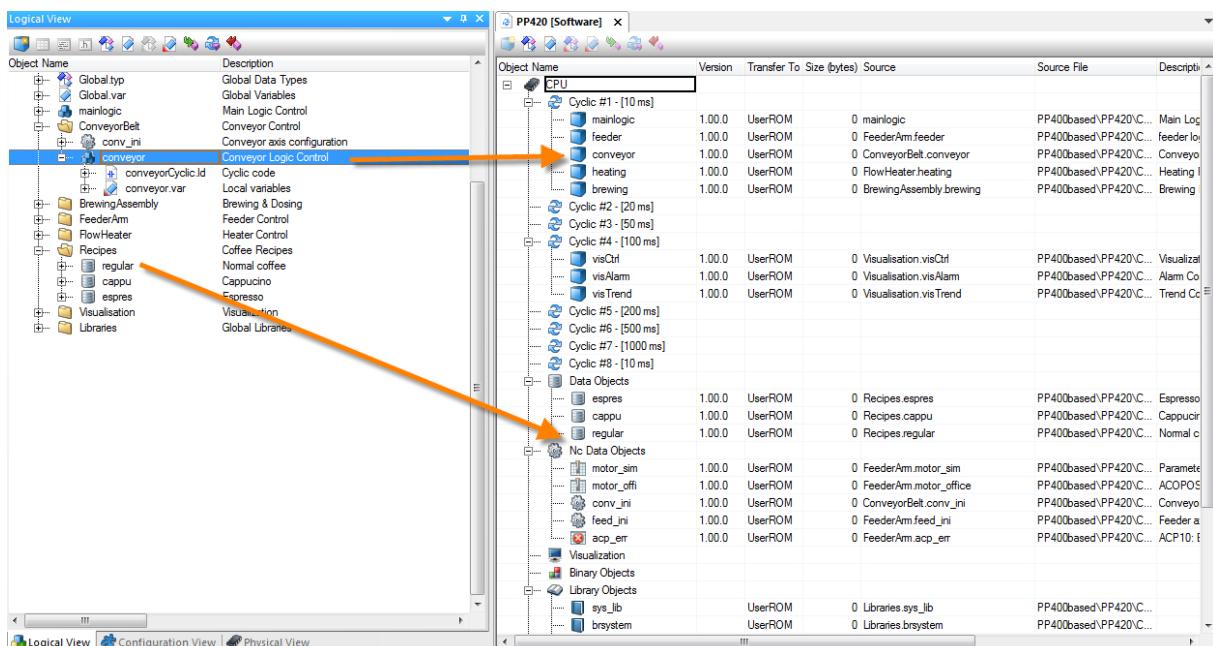


Figure 37: Assigning software elements to the software configuration



Entire packages from the Logical View can be added to the software configuration, even if they contain a mixture of programs and other object types. To do this, drag the selected package to the highest position in the software configuration.



Double-click on an entry in the software configuration to open the Logical View with the corresponding entry automatically selected.

5.4.3 Creating variants by assigning software packages

The ability to manage multiple configurations in one project allows you to work with the different variants of a given machine type all in the same project. They may differ in terms of software scope and/or the exact hardware used.

The Logical View is shown in the image on the left. This view is used to manage libraries and programs. The image on the right shows a schematic representation of the different variants of a machine. In Automation Studio, each variant is represented by a corresponding configuration. The configurations differ from each other in terms of the hardware used and the software packages that are assigned to them.

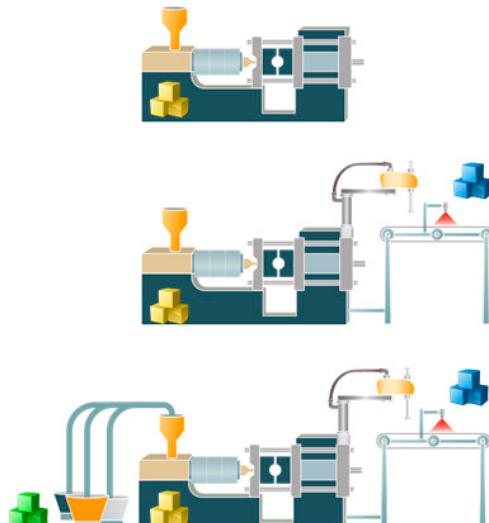
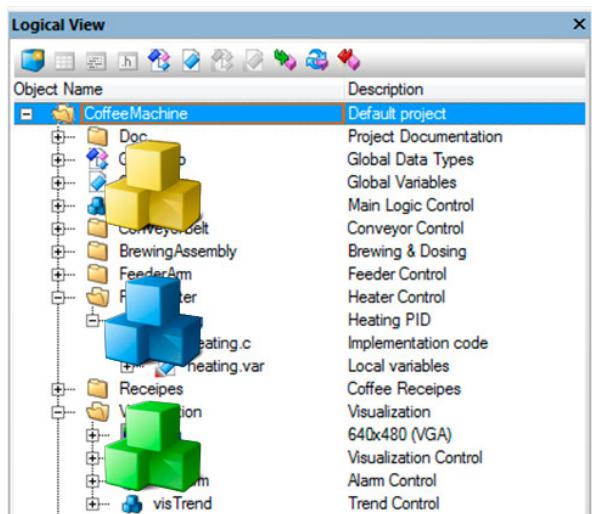


Table 2: Different machine variants can be created by assigning software packages to them.



Project management

Configuring the hardware

6 Configuring the hardware

This section is based on the project created in 3 "Your first project" on page 9.

This project already contains a configuration based on the "Automation Runtime Simulation" and a Ladder Diagram program with two variables.

Exercise: Configure the hardware and commission the system

Close the active project by selecting <File> / <Close project> from the main menu. This will display the start page, where you can open the previous project from the list of recent projects.

Assign the "LampTest" program to the Cyclic#1 task class.

- 1) Create a new configuration.
- 2) Assign the "LampTest" program to the software configuration.
["Adding existing programs manually" on page 24](#)
- 3) ["Testing in simulation mode" on page 28](#)
- 4) ["Adding I/O modules from the Hardware Catalog" on page 30](#)
- 5) ["Mapping process variables to I/O channels" on page 31](#)
- 6) ["Configuring network interfaces" on page 33](#)
- 7) ["Build the project." on page 34](#)
- 8) ["Connection using "Browse for target systems"" on page 37](#)
- 9) ["Transfer Automation Runtime" on page 39](#)
- 10) ["Project installation" on page 41](#)



This task consists of multiple subtasks that will be explained in detail in the next few sections.

Add a new configuration using the toolbox. To do this, open the Configuration View and insert a new configuration via drag-and-drop or by double-clicking. Then select the desired controller in the wizard.

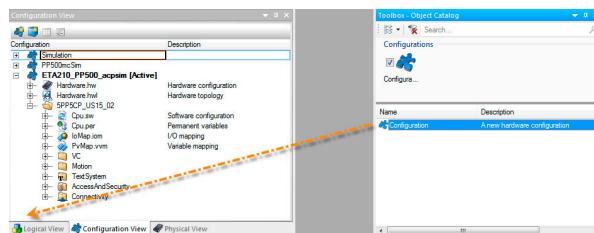


Figure 38: Add a new configuration in the Configuration View using the toolbox

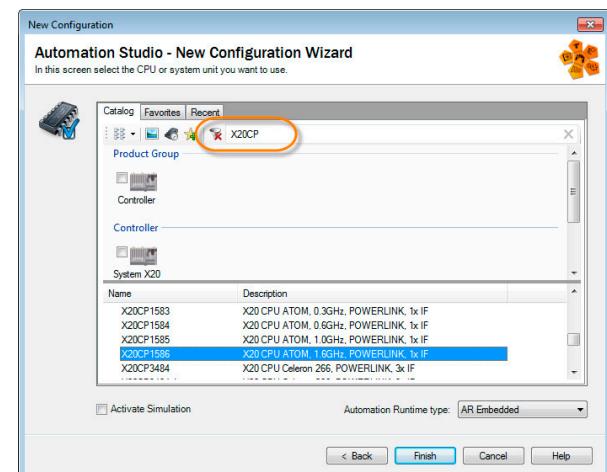


Figure 39: Select the desired controller using the filter in the wizard.

Exercise: Create a new configuration and assign programs

In this exercise, we will add a new configuration with an X20 controller (e.g. X20CP1586) to our project.

- 1) Add a new configuration via the toolbox.
- 2) Select the desired controller in the wizard.
- 3) Rename the configuration if necessary.
- 4) Open the software configuration.
- 5) Assign the "LampTest" program.

We have already created the "LampTest" program using the Getting Started section in the Automation Studio help system.

Configuring the hardware

6.1 Testing in simulation mode

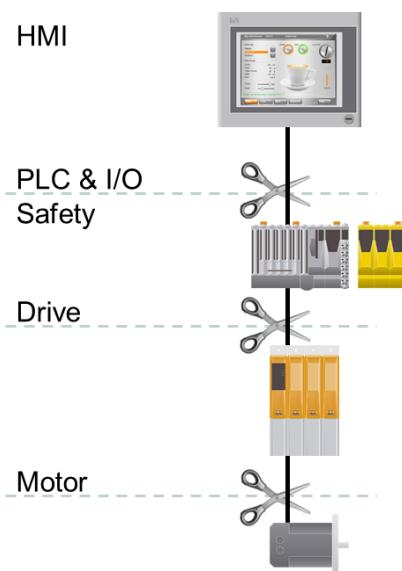


Figure 40: Complete simulation at every level

Automation Studio provides extensive simulation options for the controller, HMI, drive controller and motors. In essence, all components of an integrated automation solution from B&R can be simulated.

If it is not possible or desirable to operate the actual motor on the machine, it can instead be simulated. Motion profiles can be carried out on the controller or PC, even if the entire drive system is not available.

The platform-independent Automation Runtime system allows control programs to be created and tested directly on the PC. This function is also available for the safety application. Control applications can be executed in slow motion or time lapse in order to hone in on different phases in the machine's lifecycle.

Integrated VNC and web server functionality makes it possible to operate HMI applications not just remotely, but also directly on the PC.

The integrated WinIO interface makes it possible to fully simulate I/O points.

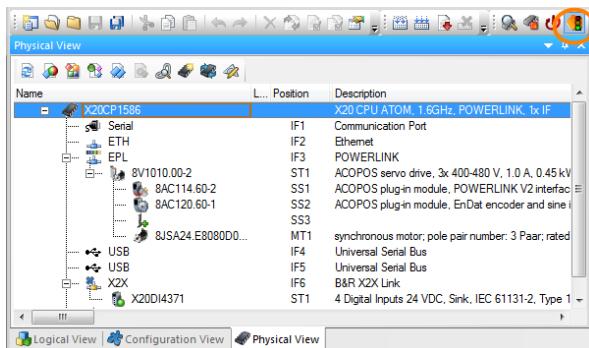


Figure 41: Enabling controller simulation from the Automation Studio toolbar

Simulation of a controller can be started by selecting the simulation icon in Automation Studio. All control programs run directly on the PC. This means that all of the software functions in the control application can be configured and tested independently of the hardware. When you switch to simulation mode, the project is rebuilt, the simulation environment is automatically started and an online connection to Automation Runtime Simulation is established.

The Automation Studio status bar indicates when a CPU simulation is running.

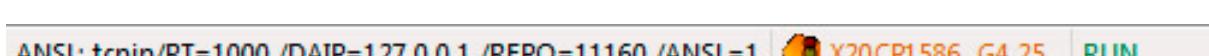
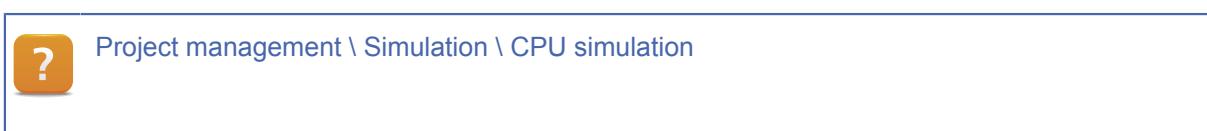


Figure 42: Automation Studio status bar - Simulation running



Exercise: Activate a simulation of the control system

Automation Studio offers comprehensive simulation functions for hardware and software. The goal of this exercise is to put the active configuration into simulation mode and test it.

- 1) Activate simulation mode via the toolbar.
- 2) Transfer the configuration - or alternatively perform offline installation.

See also:

- ° ["Project installation" on page 41](#)
- ° ["Offline installation" on page 44](#)

- 3) Testing the "LampTest" program



After the configuration has been transferred to ARsim, all the diagnostics tools¹ become available. Program sequences can be tested and further developed without having to connect to the machine. To transfer a project to the real controller, simulation mode is ended via the menu. You can tell whether the configuration is running in simulation mode using the traffic light icon in the status bar.

ANSL: tcpip/RT=1000 /DAIP=127.0.0.1 /REPO=11160 /ANSL=1 |  X20CP1586 G4.25 | **RUN**

Figure 43: Automation Studio status bar - Simulation mode active

ANSL: tcpip/RT=1000 /SDT=5 /DAIP=10.43.15.31 /REPO=11159 /ANSL=1 /PT=1... |  X20CP1586 G4.25 | **RUN**

Figure 44: Automation Studio status bar - Connection to the real hardware

¹ Further information on the topic of diagnostics can be found in the training module "TM223 - Automation Studio Diagnostics" and in the Automation Studio help system.

Configuring the hardware

6.2 Adding I/O modules from the Hardware Catalog

When using an X20 system, I/O modules can be added directly to the X2X Link interface.

I/O modules can be added either in the Physical View or in the System Designer. I/O modules can be dragged and dropped from the Hardware Catalog to the desired position.

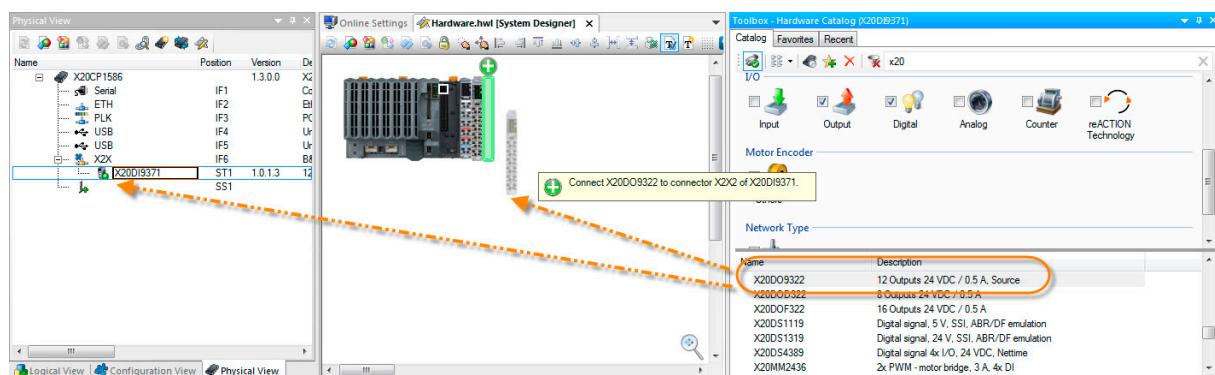


Figure 45: Assigning an I/O module via drag-and-drop



Selecting an interface (X2X, POWERLINK, etc.) in the control system filters the Hardware Catalog to show only the ones that match this interface. The results list can be filtered by setting further filter categories.

It is also possible to search for a particular module by entering all or part of its model number.

Exercise: Add a digital input and output module

In this exercise, two X20 modules will be added in the Physical View using the "Getting started" section of the Automation Studio help system as a guide. The first is a digital input module (X20DI9371); the second is a digital output module (X20DO9322).

- 1) Open the help section:



Automation software \ Getting started \ Creating programs in Automation Studio \ Mapping variables to I/O channels

- 2) Select the X2X Link interface in the Physical View.
- 3) Drag and drop the module to the X2X Link interface.



Communication \ X2X

Communication \ POWERLINK

6.3 Mapping process variables to I/O channels

I/O mapping is another way of referring to the assignment of variables being used in the control program to a module's I/O channels.

Variables are allocated to an I/O channel by selecting **<Open> / <I/O Mapping>** in the I/O mapping editor that is opened using the shortcut menu in the relevant module, or by double-clicking on the relevant I/O module.

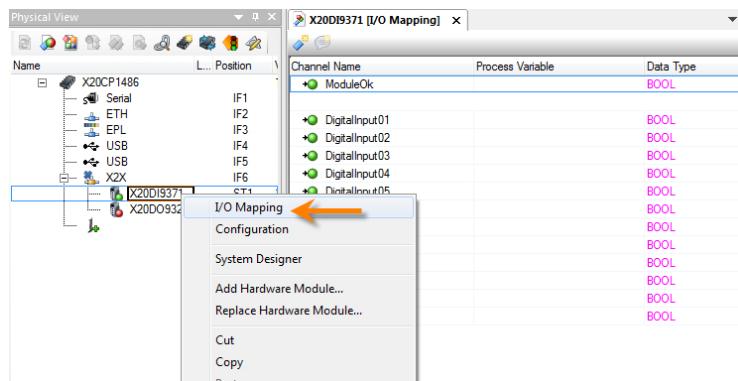


Figure 46: Open the I/O mapping for a module

Exercise: Map process variables to I/O channels

In this exercise, two variables will be mapped (one to a digital input channel and one to a digital output channel) using the "Getting started" section of the Automation Studio help system as a guide.

- 1) Open the corresponding section in the help system:

?

Automation software \ Getting started \ Creating programs in Automation Studio \ Mapping variables to I/O channels

- 2) Open the I/O mapping.
 - 3) Assign the "Switch" variable to a digital input channel.
 - 4) Assign the "Lamp" variable to a digital output channel.

Programming \ I/O configuration \ I/O mapping

Configuring the hardware

6.4 Editing an I/O configuration

All B&R I/O modules have configuration options. The I/O configuration can be used to configure I/O modules without having to do any programming.

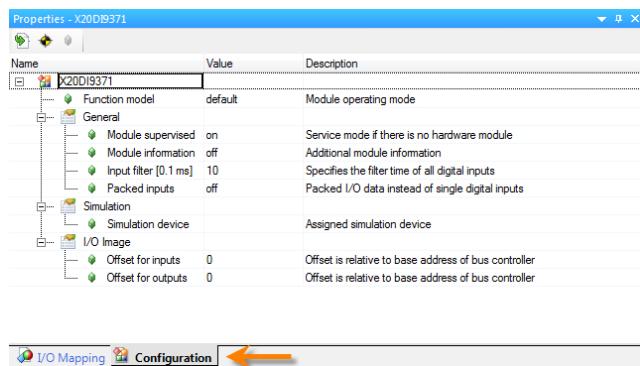


Figure 47: The Properties window is updated immediately when you select one or more modules.

The I/O configuration can only be opened by selecting **<Open>** / **<I/O configuration>** from the main menu or **<I/O configuration>** from the respective I/O module's shortcut menu.

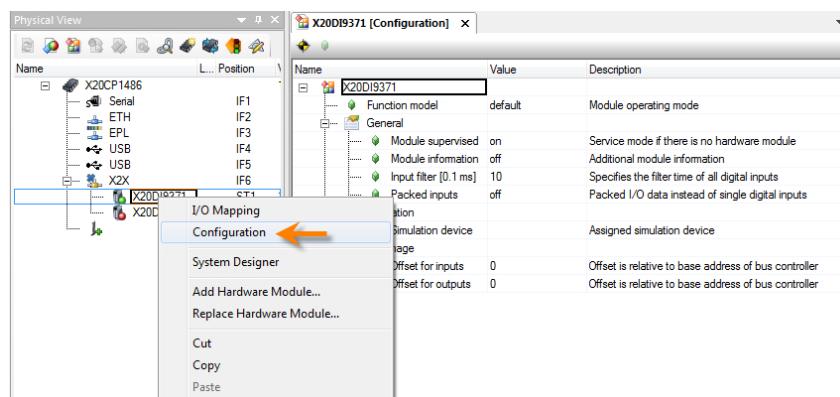


Figure 48: Opening a module's I/O configuration



The data sheet for the selected I/O module can be accessed directly from the Physical View. To do this, select the I/O module in the Physical View. Press the **<F1>** key to open the Automation Studio help system. The data sheet for the selected module appears. The data sheet contains technical information, configuration options, connection examples and details of the LED status indicators.



Programming \ I/O configuration

Hardware \ X20 system \ X20 modules

6.5 Configuring network interfaces

Automation Studio requires a network connection in order to communicate with the controller. PC and controller must therefore be on the same subnet.



If the PC is already running on a network, the settings should be noted before changing them.

Please contact your network administrator for detailed information about the integration of devices into an existing network.

The network properties can be opened using the shortcut menu for the desired Ethernet interface for the controller. It is then possible to configure the parameters for the network interface as needed.

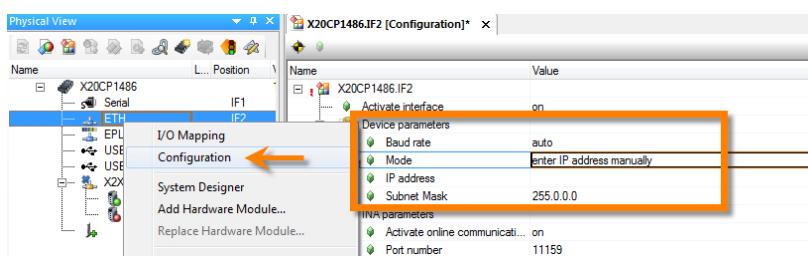


Figure 49: Opening and editing a controller's network settings

Exercise: Configure the network interface for the controller and PC

Configure the network interfaces on the PC and controller using the Automation Studio help system as a guide.



Automation software \ Getting started \ Creating programs in Automation Studio \ X20 CPU example project

- Configuring Ethernet settings on the target system
- Configuring Ethernet settings on the PC
- Configuring the online connection

Programming \ Building and transferring projects \ Establishing a connection to the target system

Communication \ Ethernet \ AR configuration \ Interface configuration

Configuring the hardware

6.6 Build the project.

Once a program has been completed, it must first be built before it can be transferred to the target system.

Build the configuration.

Building a project will recompile all of the changes made since the last build. This process is carried out using the toolbar or by pressing the <F7> key.

A successful build is indicated in the output window as follows:

Build: 0 error(s), 0 warning(s)

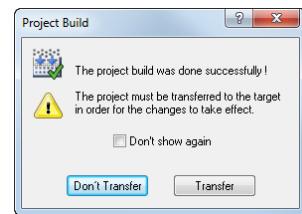


Figure 50: Dialog box indicating a successfully built configuration

Rebuild a configuration

When you recompile (rebuild) a configuration (by pressing <CTRL>+<F7>), all the software objects in the active configuration are created again, even if their source files have not changed since the last time they were compiled.



Programming \ Building and transferring projects \ Build

7 Initial installation of the controller

To install the controller, it is necessary to transfer Automation Runtime (operating system), the system components and the application project. Whether installation is performed online or offline depends on the target system. Systems with integrated flash memory are delivered with preinstalled default Automation Runtime. This guarantees that a connection with Automation Studio is established and online installation can be carried out. Offline installation is available for systems with CompactFlash and CFast cards.



Instructions on installation and commissioning can be found in the data sheet of the respective controller in section "System Flash programming".

Online installation

In BOOT mode, first the Automation Runtime version configured in the project is installed and then the Automation Studio project is transferred in its entirety.

The following steps are necessary:

- ["Connection using "Browse for target systems"" on page 37](#)
- ["Transfer Automation Runtime" on page 39](#)
- ["Project installation" on page 41](#)

Offline installation

For offline installation, Automation Runtime, system components and the application project are copied to the CompactFlash or CFast card. A USB card reader is required for this procedure.

The following steps are necessary:

- ["Offline installation" on page 44](#)
- ["Connection using "Browse for target systems"" on page 37](#)
- ["Project installation" on page 41](#)



If an operating system was already installed, then the step "Transfer Automation Runtime" is not necessary. The transfer of the project can be begun immediately once connection has been established.



Programming \ Building and transferring projects \ Establishing a connection to the target system \ Ethernet connections

Project management \ Project installation

Initial installation of the controller

7.1 Connecting to the target system

The previous section explained how to configure the network interface settings for the network controller. These settings only have to be configured once before a connection can be made. The "Browse for target systems" function allows you to assign temporary IP parameters for the controller.

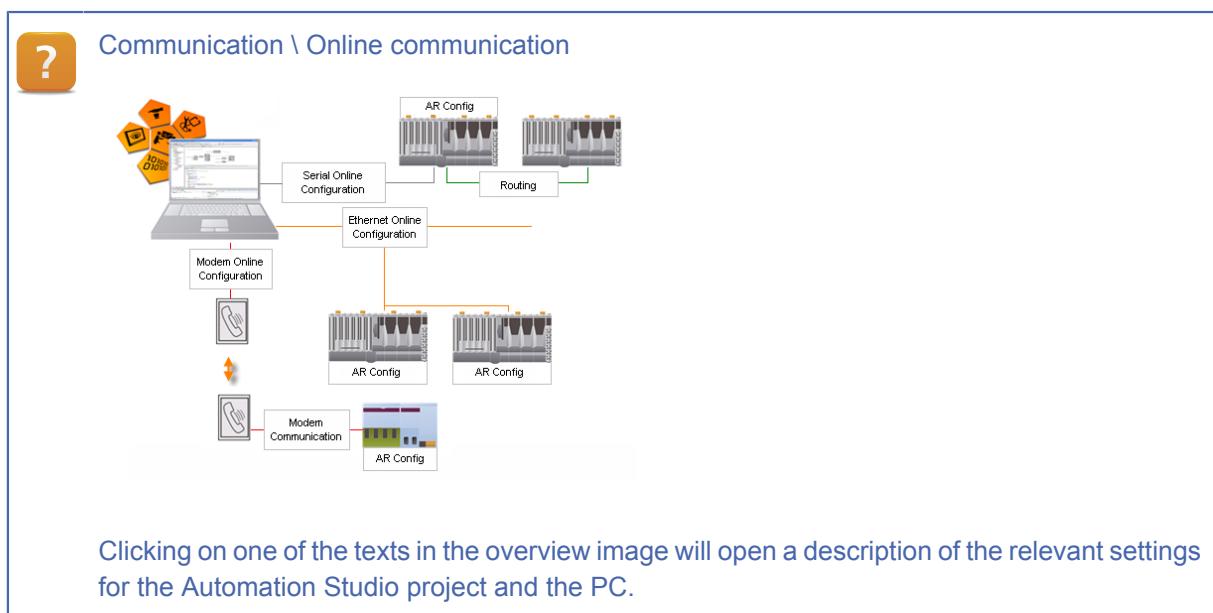
There are two ways to establish a connection:

- ["Connection using "Browse for target systems"" on page 37](#)
- ["Establish connection via manual configuration" on page 38](#)

The configuration dialog box for online connection is opened by selecting **<Online / Settings>** from the main menu.

Documentation regarding online connections and communication

For more information on the subject of communication, please refer to the Automation Studio help system. The help system contains details of the Automation Studio project settings for the communication interfaces and the required parameters for the PC network configuration.



Additional information can also be found in the following sections of the help system:



- [Programming \ Building and transferring projects \ Establishing a connection to the target system](#)
- [Communication \ Ethernet \ Network settings](#)
- [Communication \ Ethernet \ Online communication](#)
- [Communication \ Ethernet \ FAQ](#)

7.1.1 Connection using "Browse for target systems"

Clicking on the "Browse" icon in the toolbar activates the search function for the network. This will open up a second pane in the main window. The results of the network search will then appear a few seconds later. If the CPU could be identified on the network, it will be shown in the list of results. The connection is then established by right-clicking on the CPU and selecting "Connect" from its shortcut menu.

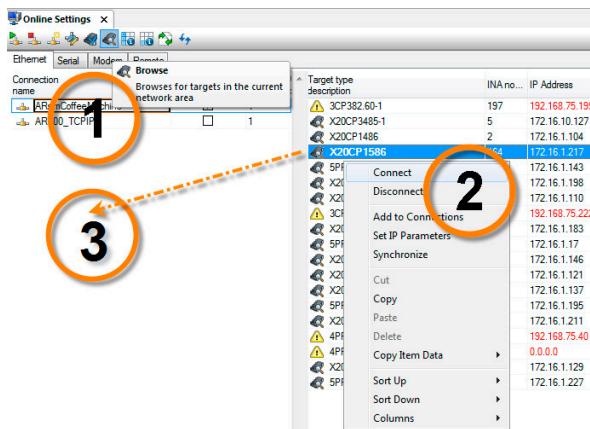


Figure 51: Browsing the network and connecting to the controller

Temporarily change IP parameters

If the network settings of the controller and PC don't match, the entry is highlighted in red in the search results.

IP parameters can be changed temporarily from an entry's shortcut menu. The temporarily changed settings can be added to the project configuration directly. If the changes are not transferred, the original settings will be used again after the controller is restarted.

- 1 Browse the network by selecting the "Browse" icon in the toolbar. Results are listed in a new pane on the right.
- 2 Select "Connect" from the CPU's shortcut menu.
- 3 The connection can be dragged from the result list to the left pane

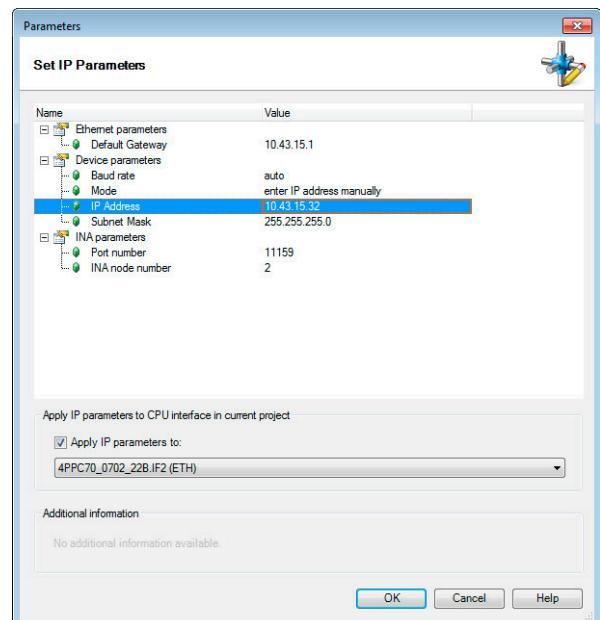


Figure 52: Temporary IP configuration



Programming \ Building and transferring projects \ Establishing a connection to the target system \ Ethernet connections

- Browse for targets
- Target system search \ Temporary IP configuration

Initial installation of the controller

Exercise: Connect to the controller

In this exercise, we will establish a connection to the controller. We will identify the controller on the network and then establish the online connection.

- 1) Open online settings
- 2) Enable the "Browse for target systems" function
- 3) Identify the target system in the results list
Target systems can be distinguished between based on the MAC address, node numbers, hostnames and serial numbers.
- 4) Establish the connection

7.1.2 Establish connection via manual configuration

Manual configuration

If it is not possible to browse the network, for example if SNMP broadcasts are blocked, it is possible to set up an online connection manually.

- 1 Add a new connection using the icon in the toolbar.
- 2 Specify the connection parameters (IP address, node number).
- 3 Enable the new connection from its shortcut menu.

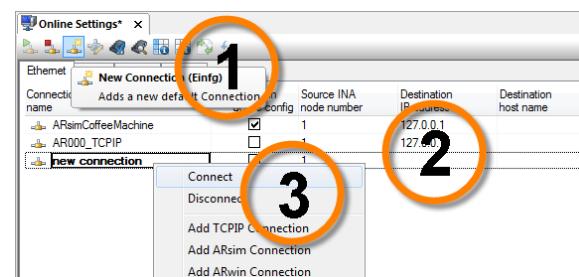


Figure 53: Add and configure a connection manually.



Programming \ Building and transferring projects \ Establishing a connection to the target system \ Ethernet connections



As soon as a connection exists between the PC and the controller, its status will be shown in the Automation Studio status bar.

- In BOOT mode, Automation Runtime can then be transferred to the controller.



Figure 54: An online connection to the controller has been established. The controller is in BOOT mode.

- In RUN or SERVICE mode, the entire project can be transferred to the controller. Automation Studio diagnostic tools are available when there is an active online connection.



Figure 55: An online connection to the controller has been established. The controller is in RUN mode.

7.2 Transfer Automation Runtime

When the target system is in BOOT mode, the transfer of Automation Runtime is started by selecting **<Online> / <Services> / <Transfer Automation Runtime>** from the menu. If the target system is in RUN or SERVICE mode, Automation Runtime can be transferred directly along with the project.

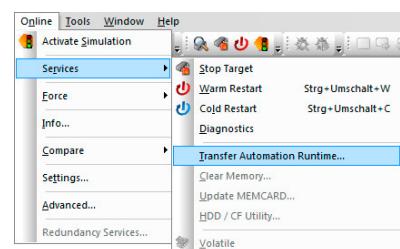


Figure 56: Transfer Automation Runtime

There is a wizard to assist you through the process of transferring Automation Runtime. It is recommended to select the option "Modules from the project scheme". This option includes the system settings from the project in addition to Automation Runtime. As a requirement for this, the project must be compiled in advance.

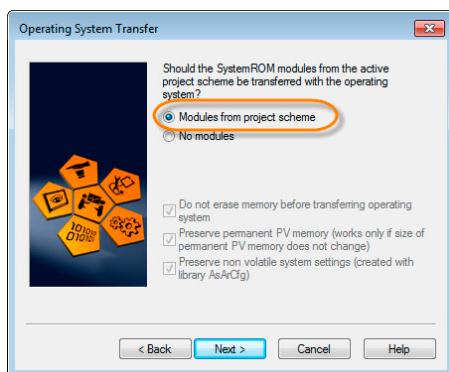


Figure 57: Transferring Automation Runtime: Transfer modules from the project scheme

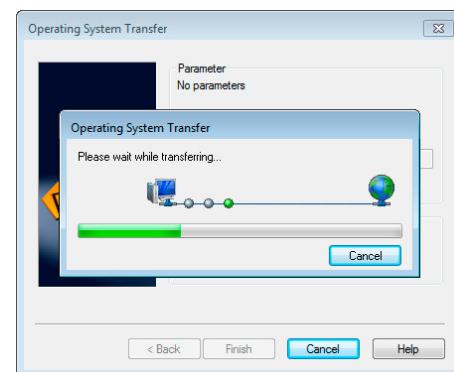


Figure 58: Transferring Automation Runtime: Progress indicator

Initial installation of the controller



All target systems have the option of deliberately starting in BOOT mode. Depending on the target system, this is done using a reset button, the mode selector switch or node number selector switch. The reset button can be used to restart the system and change the operating mode. The mode selected with a mode selector switch is applied after restarting. Further information about the possible operating modes and actions can be found in the data sheet of the respective target system.

Starting from default Runtime² Browsing for target systems via SNMP³ is also supported for SG4 systems with V3.06 and later.



[Programming \ Build & transfer \ Online services \ Transfer Automation Runtime](#)

[Hardware \ Power Panel \ Power Panel C-Series \ Commissioning \ Installation](#)

[Hardware \ X20 system \ X20 modules \ CPUs \ X20CP1301, X20CP1381 and X20CP1382 \ Programming the system flash](#)

Exercise: Transfer Automation Runtime

The objective of this task is to transfer Automation Runtime to the target system.

- 1) Establish a connection to the controller via "Browse for target systems" function.
- 2) [optional] Put the target system into BOOT mode via the reset button, the mode selector switch or node number selector switch.
- 3) Start Automation Runtime transfer via **<Online> / <Services> / <Transfer Automation Runtime>**
- 4) Select the option "Modules from the project scheme"
- 5) Close the wizard and the system to restart



After Automation Runtime has been transferred, the controller is in RUN mode. Now, the entire project can be transferred to the controller.

`ANSL:tcpip/RT=1000 /SDT=5 /DAIP=10.43.15.32 /REPO=11159 /ANSL=1 /PT=1... | 4PPC70.0702-20B G4.25 | RUN`

Figure 59: Automation Runtime transferred successfully. Connection established. The controller is in RUN mode.

- 2 The default Automation Runtime is a reduced variant of Automation Runtime which is preinstalled on all controllers. For example, the default Automation Runtime boots from flash memory.
- 3 The Simple Network Management Protocol is a network protocol used to monitor and control devices on a network from a central location (e.g. routers, servers, switches, printers, computers, etc.).

7.3 Project installation

If the project is compiled and Automation Runtime installed on the target system⁴, then the next step is to transfer the project.

The process is started via the transfer symbol in the Automation Studio toolbar. If need be, the latest changes are compiled in the project.



Figure 60: Transfer symbol in the Automation Studio toolbar

Initial transfer

An initial transfer is performed the first time a project is installed. During this, the target system is completely initialized. All data from any previous projects is removed. With an initial transfer, the entire project including Automation Runtime is always transferred to the target system. The memory is partitioned, formatted and the target system is restarted. RETAIN and permanent variables are deleted during an initial transfer.

An initial transfer is performed under the following conditions:

- The configuration ID of the target system differs from the one defined in the project. The configuration ID is established in the system configuration of the controller.
- The partitioning of the target system doesn't match the partitioning required for the project.
- The transfer takes place to a data storage device (see "[Offline installation via data storage device](#)" on page 43).
- The user forces an initial transfer in the transfer settings.

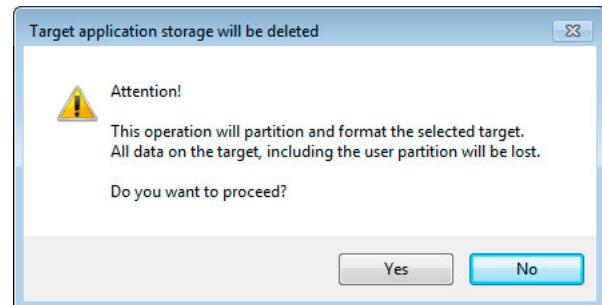
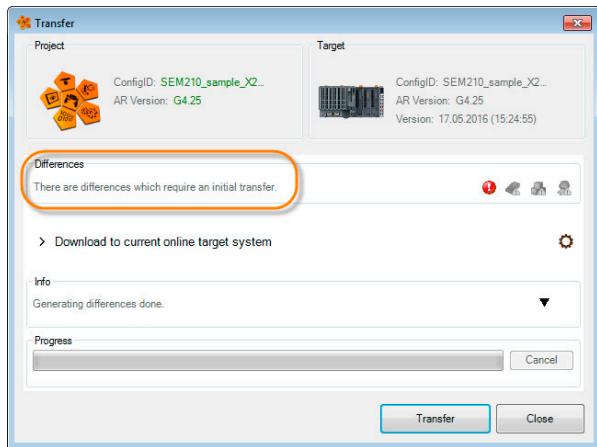


Figure 62: Note: The memory is formatted during an initial transfer

Figure 61: Transfer dialog box: Initial transfer required

⁴ With Automation Runtime C4.25 and later, the Automation Runtime version is transferred with the project, included in the transfer module and automatically installed.

Initial installation of the controller

Update transfer

Automation Studio will generally first attempt an update transfer on the target system. If an update transfer is not possible (the configuration ID has changed, for example), then an initial transfer is performed instead. During an update transfer, only data that has been changed is transferred to the target system. Depending on the transfer settings, the Init and Exit programs are executed or values are received from process variables.

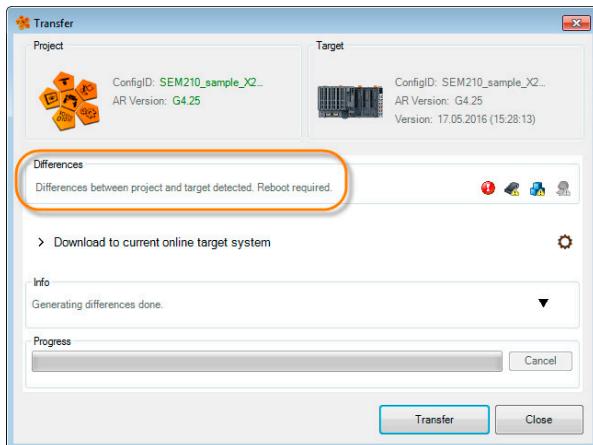


Figure 63: Transfer dialog box: Project differences require a restart

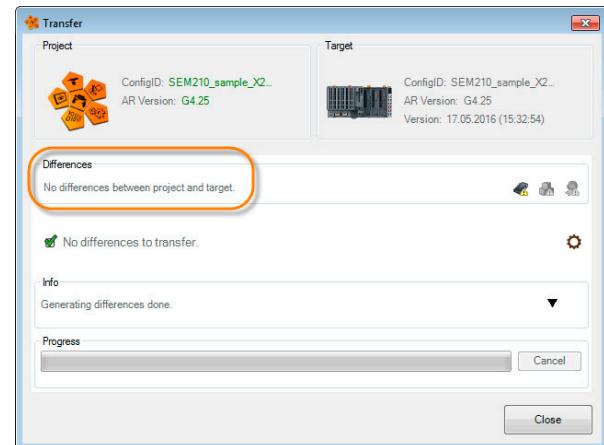


Figure 64: Transfer dialog box: no differences were identified between the project and the target system.



In addition to the textual notes, notification icons are shown in the transfer dialog box. These can be used to open tooltips with detailed information about the differences between the project and the target system configuration.



Figure 65: Additional information from notification icons in the transfer dialog box



Project management \ Project installation

- General \ Glossary
- Sequence \ Transfer type \ Initial transfer
- Sequence \ Transfer type \ Update transfer
- Operation \ Settings \ Settings during the transfer

Exercise: Carry out online project installation

After successful installation, the project must now be installed on the target system. Future project changes are loaded to the target system via online installation.

- 1) Build the project
- 2) Start the transfer
- 3) Keep an eye on instructions in the transfer dialog box.

- 4) Confirm the transfer.
- 5) Wait for the download and optional restarts.



The target system is now up-to-date. The function of the program "LampTest" can now be tested directly on the target system.

Exercise: Test the program

The "LampTest" program can now be tested on the target system. Setting the "Switch" digital input should set the "Lamp" output. Check the function in Automation Studio and also on the LED status indicators of the assigned I/O modules.

- 1) Enable monitor mode

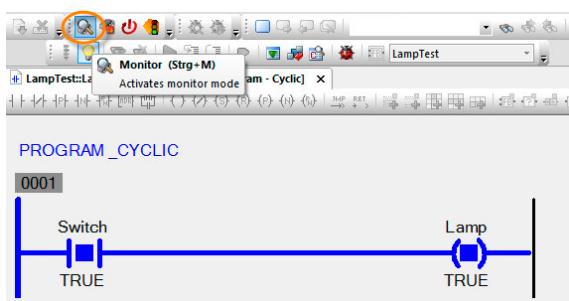


Figure 66: Ladder Diagram program "LampTest" in monitor mode with powerflow

- 2) Set the "Switch" input and observe the "Lamp" output
- 3) Compare the states in Automation Studio with the LED status indicators

7.4 Offline installation via data storage device

Alternatively to online project installation, offline installation (["Offline installation" on page 44](#)) or the USB remote install function (["Installation via remote install structure" on page 45](#)) can be used. In both cases, Automation Runtime and the Automation Studio project files are transferred.



Figure 67:
CompactFlash

Initial installation of the controller

7.4.1 Offline installation

During offline installation, the project is installed on a storage device. The transfer dialog box for doing this is slightly different than the one for online installation. The main difference is that a data storage device can be selected for the transfer. The dialog box for offline installation is opened by selecting <Tools> / <Offline Installation>.

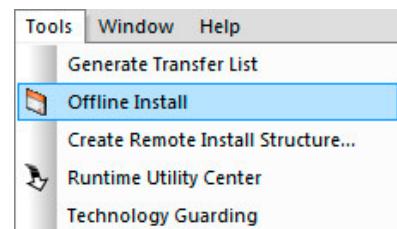


Figure 68: Open offline installation

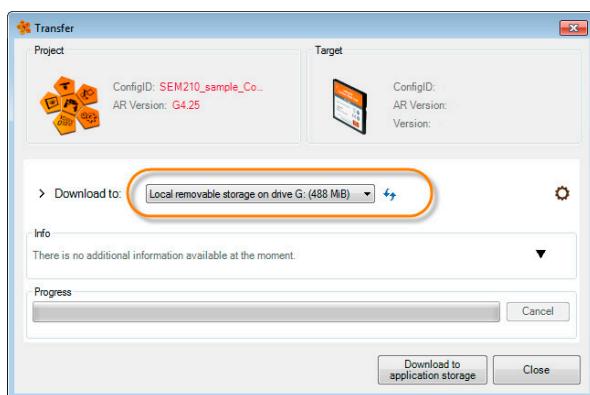


Figure 69: The dialog box for offline installation on a CompactFlash or CFast card

On CompactFlash or CFast card

Installation of the target system is carried out by creating an offline installation on a CompactFlash or CFast card. This process carries out partitioning, transfers the operating system and configures the necessary system settings.

Once the CompactFlash or CFast card has been inserted into the target system, the controller starts the installation project after the supply voltage has been turned on.

Automation Runtime Simulation

An installation package can be generated for the Automation Runtime Simulation (ArSim) at any time with offline installation. In the process, Automation Runtime files, project configuration and programs are copied to a local folder. Then the Automation Runtime Simulation is started automatically.



Project management \ Project installation \ Operation \ Transfer dialog box \ Offline installation

7.4.2 Installation via remote install structure

Automation Runtime and application software can be transferred to the target system by creating a remote install structure, either with a USB flash drive, a CompactFlash card or a DHCP server. The following section deals with remote installation using a USB flash drive.

In order to do that, the function "USB remote install" must have been enabled in the system settings of the controller.

First the Automation Studio project is compiled and a transfer list created.

The remote install structure is copied to a USB flash drive using the Runtime Utility Center, which is automatically installed when Automation Studio is installed.

The USB flash drive is inserted in the target system.

The next time the controller starts, it checks the versions of Automation Runtime and the application software and updates them if necessary.

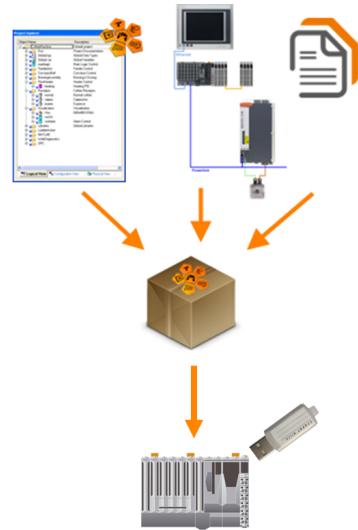


Figure 70: Update Automation Runtime and application via USB remote install

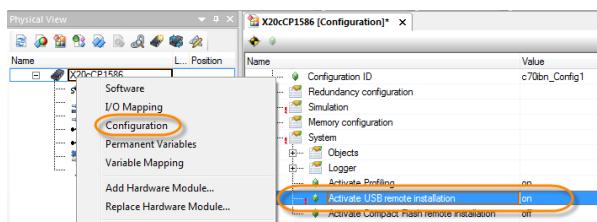


Figure 71: Enabling the USB remote install function in system properties

Enable USB remote install

In order for the control system⁵ to continue to support this function after the first update via USB remote install, you must ensure that this function is enabled in the system configuration. The configuration is opened in Physical View via the controller's shortcut menu. The required configuration entry is located in the "System" section.



Real-time operating system \ Target systems \ SG4 \ AR remote install

- Configuration
- Remote install structure \ USB

Creating a USB remote install structure

When you click on <Tools> / <Create remote install structure> in Automation Studio, a transfer list is generated. Then the Runtime Utility Center service tool is started.

⁵ This applies to control systems using the default Automation Runtime.

Initial installation of the controller

The USB flash drive can be selected in the configuration dialog box. It is also possible to decide if Automation Runtime should be updated along with the application. A version check prevents the target system from being updated unintentionally.

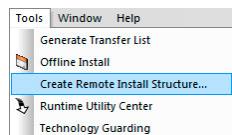


Figure 72: Start Runtime Utility Center

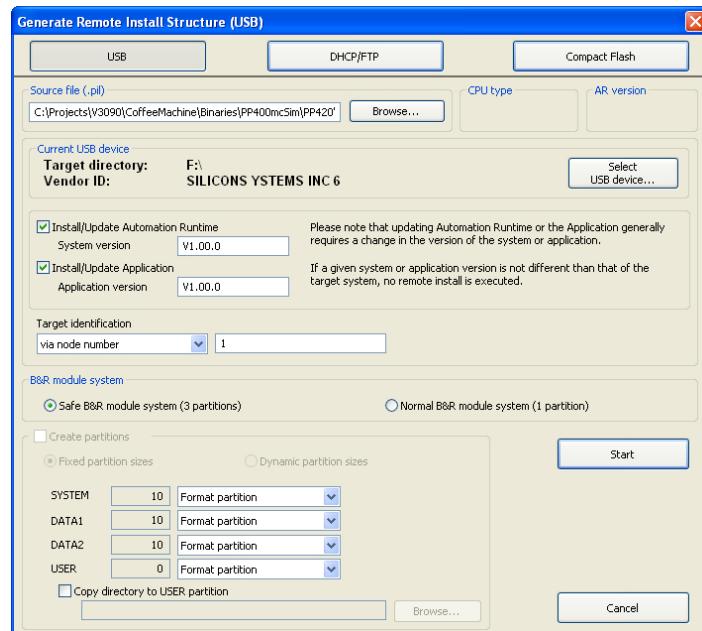


Figure 73: Runtime Utility Center installer dialog box

Pressing the "Start" button copies the remote install structure to the USB flash drive. The system can now be updated using the newly set-up USB flash drive.



Diagnostics and service \ Service tool \ Runtime Utility Center \ Creating a list / data medium \ Remote install structure creation

8 Programming

A program is a POU (program organization unit, as defined in the IEC 61131 standard) that possesses the ability to directly access all global variables, functions and function blocks.

8.1 Programming languages

A variety of different programming languages are available in Automation Studio for creating programs. It is also possible to combine multiple programming languages within a single project.

Programming languages available in Automation Studio

Programming language	IEC61131	Comment
Ladder Diagram (LD)	Yes	Visual
Function Block Diagram (FBD)	Yes	Visual
Continuous Function Chart (CFC)	No	Visual
Sequential Function Chart (SFC)	Yes	Visual and text-based
Instruction List (IL)	Yes	Text-based
Structured Text (ST)	Yes	Text-based
ANSI C and C++	No	Text-based

Table 3: Overview of programming languages

All text-based programming languages in Automation Studio use the same editor. As a result, diagnostic tools always have the same features and are always operated in the same way. This high degree of uniformity simplifies workflows and increases productivity.



Function blocks included in B&R standard libraries can be called and used in all of the programming languages.



Programming \ Programs

8.2 Initialization and cyclic subroutines

When a program is added using the toolbox, the cyclic program section, the initialization subroutine and the exit program are automatically added. Program sections that are not required can be deleted in the Logical View. A program must always have a cyclic program section.

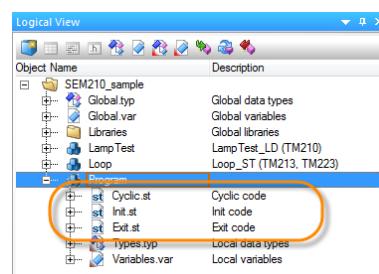


Figure 74: Image of a newly added program in the Logical View - initialization subroutine, cyclic program and exit program.

Task initialization

When the cyclic system is started, each task executes its initialization subroutine. This Init program can contain program code which calculates and describes variable values.

Cyclic subroutine

The program's cyclic subroutine starts once the task's initialization subroutine has completed. Variables which are described there retain their values until they are overwritten or until the system is restarted.

Exit subroutine

A task's exit program is only called when the task is uninstalled (deleted). If resources (e.g. memory, interfaces, etc.) were requested in the initialization or cyclic subroutine, then these resources must be freed up properly.



Additional information is included in the "TM213 – Automation Runtime" training module.



Real-time operating system \ Method of operation \ Runtime performance \ Tasks

8.3 Variables and data types

Variables serve as storage for values. Variables are given a name and are managed by the operating system in the controller's memory. In Automation Studio, variables are declared in files with the extension .var.

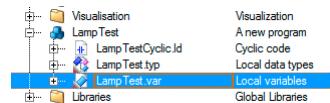


Figure 75: Variable declaration file

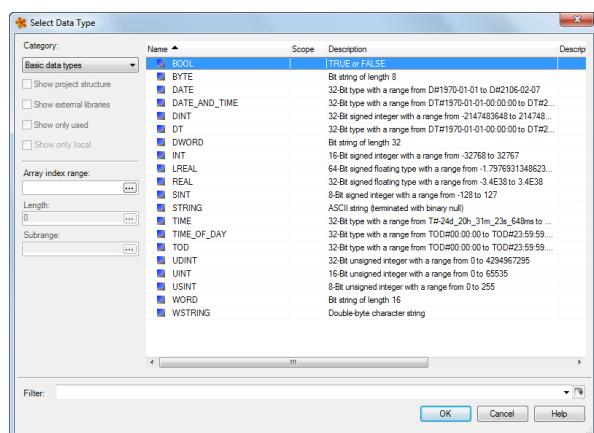


Figure 76: Variable data types

Data types determine the properties of variables. They define things like the range of values, the precision of the number stored in the variable or the possible operations.

It is also possible for the user to create **user-defined data types** that are based on basic data types. In Automation Studio, these user-defined data types (or derived data types) are declared in a file with the extension .typ.

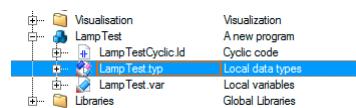


Figure 77: Data type declaration file



Programming \ Variables and data types

- Variables
- Data types \ Basic data types
- Data types \ Derived data types

8.3.1 Variable scope

In the Logical View, you can structure the software as required using packages. This facilitates the simple encapsulation of data and functionality.

This structure determines the scope and visibility of the declared variables and data types. This allows you to define variables the appropriate "logical" place in the project.

Automation Studio controls the scope of variables using the position of the .var file.

(1) Local variables are defined as having a local scope within a program for this reason they are not available to other POUs⁶ in the project.

(2) Package-global variables are declared within a specific package and are only valid in that package as well as in all subordinate packages and programs.

(3) Global variables are located at the highest possible level and are visible throughout the entire project. They can therefore be used in any program, regardless of the hierarchical level of the package containing it.

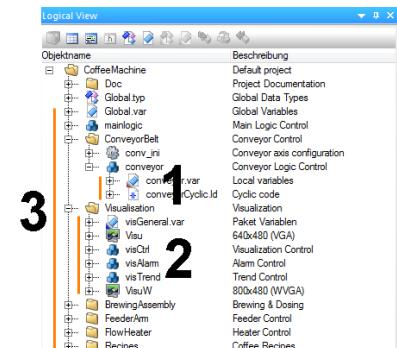


Figure 78: Scopes in the Logical View



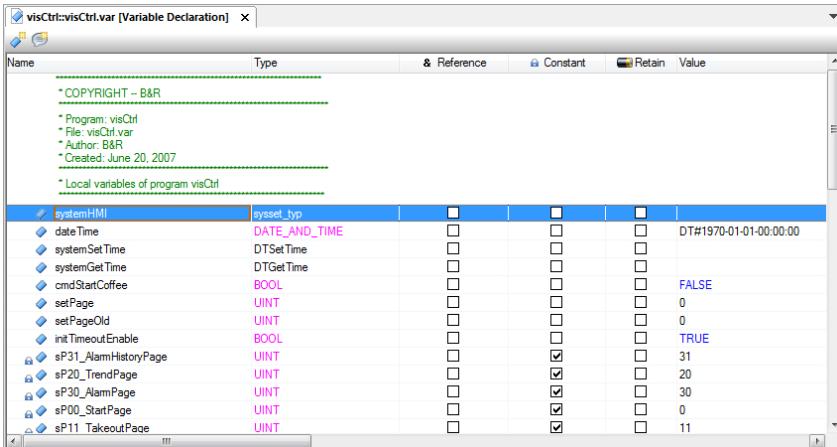
Programming \ Variables and data types \ Scope of declarations

⁶ Program organization units are defined in the IEC 61131-3 standard and referred to as POUs. POUs correspond to programs functions and function blocks into which the control project is divided.

Programming

8.3.2 Initializing and buffering variables

A variable always has a data type as one of its properties. The variable declaration can contain additional properties for the variable.



The screenshot shows a software interface for variable declaration. The title bar is "visCtrl:visCtrl.var [Variable Declaration]". The main area is a table with columns: Name, Type, & Reference, Constant, Retain, and Value. The table contains several rows of variable definitions, including systemHMI, date, systemSetTime, systemGetTime, cmdStartCoffee, setPage, setPageOld, initTimeoutEnable, sP31_AlarmHistoryPage, sP20_TrendPage, sP30_AlarmPage, sP00_StartPage, and sP11_TakeoutPage. The "Value" column for the first row contains "DT#1970-01-01-00:00:00".

Name	Type	& Reference	Constant	Retain	Value
*COPYRIGHT - B&R					
* Program: visCtrl					
* File: visCtrl.var					
* Author: B&R					
* Created: June 20, 2007					
* Local variables of program visCtrl					
systemHMI	sysset_typ				DT#1970-01-01-00:00:00
date	DATE_AND_TIME				
systemSetTime	DTSetTime				
systemGetTime	DTGetTime				
cmdStartCoffee	BOOL				
setPage	UINT				
setPageOld	UINT				
initTimeoutEnable	BOOL				
sP31_AlarmHistoryPage	UINT		<input checked="" type="checkbox"/>		31
sP20_TrendPage	UINT		<input checked="" type="checkbox"/>		20
sP30_AlarmPage	UINT		<input checked="" type="checkbox"/>		30
sP00_StartPage	UINT		<input checked="" type="checkbox"/>		0
sP11_TakeoutPage	UINT		<input checked="" type="checkbox"/>		11

Figure 79: Example of a variable declaration

Constants are variables whose values cannot be changed while a program is being executed. A constant is assigned its initial value when the software is created (Value column).

RETAIN variables are protected in buffered memory so that they can be reloaded after a system restart (warm restart). Unlike RETAIN variables, permanent variables are also protected against a cold restart. In order for variables to be stored in the permanent memory area, they have to be defined as RETAIN and Global in the variable declaration window.

Depending on the target system used, a buffer battery is used in the CPU to retain the data. More detailed information is listed in the data sheet of the respective device.



Programming \ Variables and data types

Real-time operating system \ Method of operation \ Module / data security \

- Power-off handling
- Power-on handling

8.4 Creating an application

Automation Studio includes, in addition to a wide selection of programming languages, a number of components for efficient and platform-independent project configuration.

8.4.1 Libraries, samples and solutions

The standard libraries delivered with Automation Studio make it possible for users to access many different system functions. They allow, for example, physical interfaces to be freely programmed, reports to be used in a targeted manner and hardware to be configured with a high level of flexibility.

It is also possible to import numerous sample programs, which demonstrate how the libraries can be used. These encompass executable program code for the controller or the simulation.

With solutions that are installed via the Automation Studio upgrade dialog box, you receive complete project templates with process control, simulation and visualization. These are imported into the project by the user and adjusted as required.



Programming

- Libraries
- Examples

Solutions \ Technology Solutions

8.4.2 mapp Technology

With mapp Technology⁷, we offer users an easy-to-use interface for implementing comprehensive functionality. Many complex operations, such as loading and saving recipe data, controlling a drive axis and recording process values, can be implemented quickly and easily using mapp Technology components.



Figure 80: mapp Technology logo

mapp Technology unites configuration and programming. Functionality is implemented in the application program using standard libraries. In addition, mapp provides configuration interfaces that allow the functionality of mapp components to be influenced independently of where they are used in the application software.



Application layer - mapp Technology

- Concept
- Getting started
- Components

8.5 Import, export and team functions

Working in a team means that the various responsibilities have to be divided up between team members. Automation Studio includes functions designed to help teams work more efficiently:

- Passing on project data via export with small file sizes
- Applying project data via import
- Working with source control systems



Project management

- Automation Studio project \ Project import/export
- Distributed development
- Using source control systems

⁷ mapp Technology stands for "Modular APPlication technology".

Programming

8.5.1 Exporting projects

An export function allows Automation Studio projects to be shared with other programmers.

Projects can be exported by selecting **<File> / <Save project as Zip>** or **<File> / <Save project as Zip without upgrades>** from the main menu.

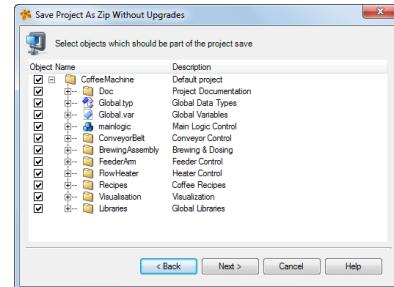


Figure 81: Exporting software components from the Logical View

Project Management \ Automation Studio project \ Project import/export

8.5.2 Exporting and importing software components

Completed software components can be exported and imported so that they can be shared or reused in other projects. Exports are performed individually for each package in the Logical View.

If a package includes dependencies to libraries, they can be entered in the properties of the package. These properties are checked when importing a package so that any required libraries are added automatically.

Preprepared samples are available in Automation Studio as packages for B&R standard libraries that can be imported into an existing project whenever needed.

Importing to the Logical View is carried out using the toolbox.

Programming \ Examples

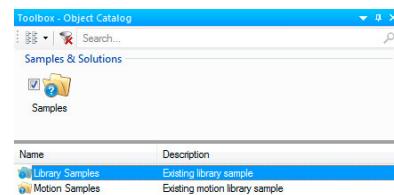


Figure 82: Adding samples from the toolbox

9 Updates and licenses

In addition to the offline installation function and opening System Diagnostics Manager, Automation Studio's "Tools" menu offers other functions as well. This section briefly explains the function of the Upgrades dialog box as well as Technology Guarding.

9.1 Hardware and software upgrade

The ability to upgrade components makes it possible to update hardware and motion libraries as well as upgrade to newer versions of Visual Components or Automation Runtime.

Upgrading components online is handled directly in Automation Studio.

The upgrade process is started by selecting <Tools> / <Upgrades> from the menu. A dialog box opens, showing available upgrades.

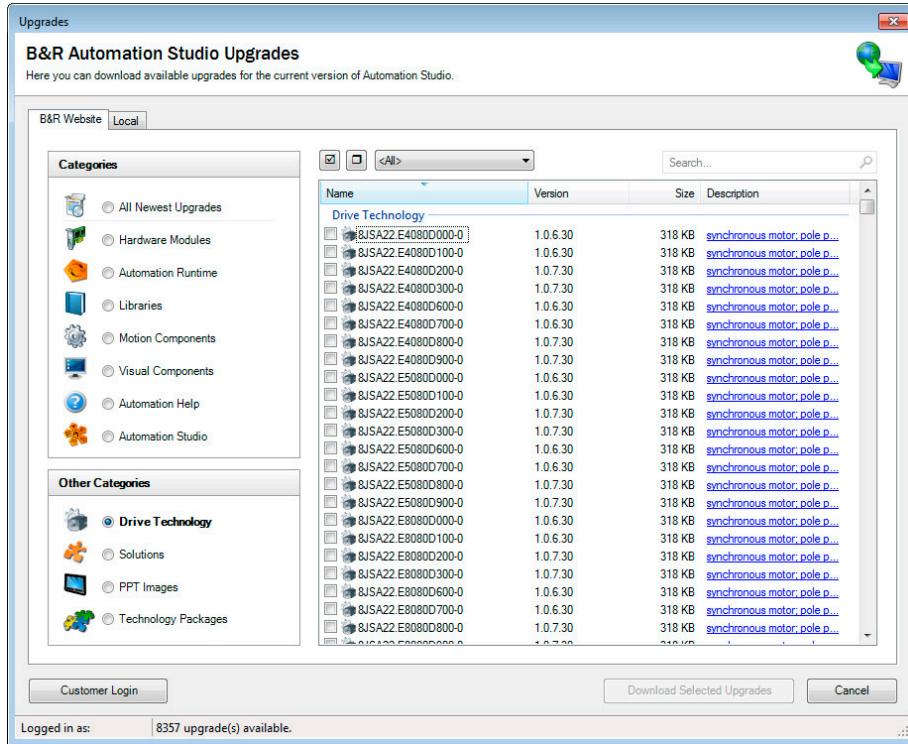


Figure 83: Selecting subsequently loadable components in Automation Studio



Updates and licenses

9.2 Technology Guarding

Technology Guarding is used to license protected libraries. License protection can also be implemented for machine software. This technology protects against unauthorized duplication of machine software and facilitates implementation of machine options. Licenses are stored in encrypted form to prevent tampering. Licenses can come preinstalled on a Technology Guard dongle from B&R or downloaded in the field using the Technology Guarding function in Automation Studio. The Technology Guarding portal on the B&R website provides users full transparency of available and already activated licenses.

The Technology Guard provides two manipulation-proof operating time counters and permanent data storage. These functions can be used via the AsGuard library in the application software.

Licenses on the inserted Technology Guard are verified automatically at runtime by Automation Runtime.

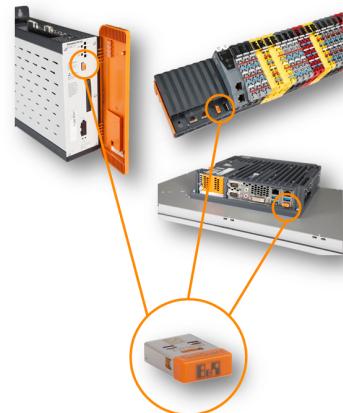


Figure 84: Technology Guard (0TG1000.01) for tamper-proofing and for license protection on the machine



Automation software \ Technology Guarding
Programming \ Libraries \ Configuration, system \ AsGuard

10 Summary

Automation Studio is more than just a programming tool. It provides support for the user throughout the entire lifecycle of a machine – from the initial concept all the way to the finished project.

Whether it's being able to clearly structure software to match the physical parts of the machine, working with different configurations to design multiple variants of the same machine or working together concurrently in a team on the same project – Automation Studio always has the right tools for the job at hand.

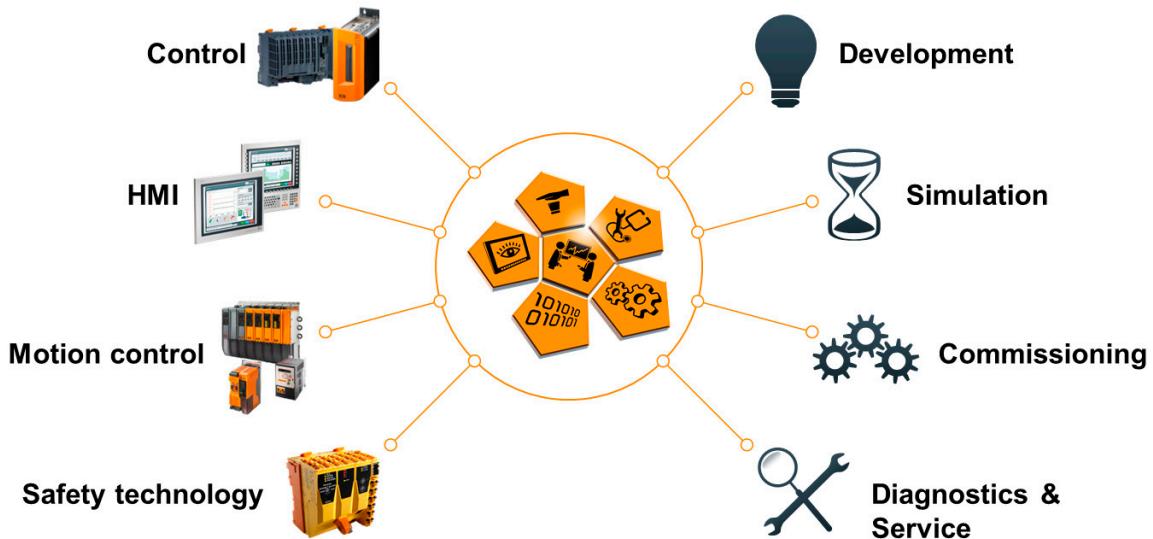


Figure 85: Automation Studio: one engineering tool for the machine's entire lifecycle

Automation Studio is an invaluable resource for programmers, service technicians and maintenance engineers for every stage of a machine's lifecycle.

Seminars and training modules

Seminars and training modules

The Automation Academy provides targeted training courses for our customers as well as our own employees.

At the Automation Academy, you'll develop the skills you need in no time!

Our seminars make it possible for you to improve your knowledge in the field of automation engineering.

Once completed, you will be in a position to implement efficient automation solutions using B&R technology. This will make it possible for you to secure a decisive competitive edge by allowing you and your company to react faster to constantly changing market demands.



Automation Studio seminars and training modules

Programming and configuration	Diagnostics and service
<p>SEM210 – Basics SEM246 – IEC 61131-3 programming language ST* SEM250 – Memory management and data storage</p> <p>SEM410 – Integrated motion control* SEM441 – Motion control: Electronic gears and cams** SEM480 – Hydraulics** SEM1110 – Axis groups and path-controlled movements**</p> <p>SEM510 – Integrated safety technology* SEM540 – Safe motion control***</p> <p>SEM610 – Integrated visualization*</p>	<p>SEM920 – Diagnostics and service for end users SEM920 – Diagnostics and service with Automation Studio SEM950 – POWERLINK configuration and diagnostics*</p> <p>If you don't happen to find a seminar on our website that suits your needs, keep in mind that we also offer customized seminars that we can set up in coordination with your sales representatives: SEM099 – Individual training day</p>

Please visit our website for more information****: www.br-automation.com/academy

Overview of training modules

TM210 – Working with Automation Studio TM213 – Automation Runtime TM223 – Automation Studio Diagnostics TM230 – Structured Software Development TM240 – Ladder Diagram (LD) TM241 – Function Block Diagram (FBD) TM242 – Sequential Function Chart (SFC) TM246 – Structured Text (ST) TM250 – Memory Management and Data Storage	TM600 – Introduction to Visualization TM610 – Working with Integrated Visualization TM630 – Visualization Programming Guide TM640 – Alarm System, Trends and Diagnostics TM670 – Advanced Visual Components
TM400 – Introduction to Motion Control TM410 – Working with Integrated Motion Control TM440 – Motion Control: Basic Functions TM441 – Motion control: Electronic gears and cams TM1110 – Integrated Motion Control (Axis Groups) TM1111 – Integrated Motion Control (Path Controlled Movements) TM450 – Motion Control Concept and Configuration TM460 – Initial Commissioning of Motors	TM920 – Diagnostics and service TM923 – Diagnostics and Service with Automation Studio TM950 – POWERLINK Configuration and Diagnostics
TM500 – Introduction to Integrated Safety TM510 – Working with SafeDESIGNER TM540 – Integrated Safe Motion Control	TM280 – Condition Monitoring for Vibration Measurement TM480 – The Basics of Hydraulics TM481 – Valve-based Hydraulic Drives TM482 – Hydraulic Servo Pump Drives TM490 – Printing Machine Technology

In addition to the printed version, our training modules are also available on our website for download as electronic documents (login required):

Please visit our website for more information:
www.br-automation.com/academy

Process control seminars and training modules

Process control standard seminars	Process control training modules
SEM841 – Process Control Training: Basic 1 SEM842 – Process Control Training: Basic 2 SEM890 – Advanced Process Control Solutions	TM800 – APROL System Concept TM810 – APROL Setup, Configuration and Recovery TM811 – APROL Runtime System TM812 – APROL Operator Management TM813 – APROL Web Portal TM820 – APROL Solutions TM830 – APROL Project Engineering TM835 – APROL ST-SFC Configuration TM840 – APROL Parameter Management and Recipes TM850 – APROL Controller Configuration and INA TM860 – APROL Library Engineering TM865 – APROL Library Guide Book TM870 – APROL Python Programming TM880 – APROL Reporting TM890 – The Basics of LINUX

* SEM210 - Basics is a prerequisite for this seminar.

** SEM410 - Integrated motion control is a prerequisite for this seminar.

*** SEM410 - Integrated motion control and SEM510 - Integrated safety technology are prerequisites for this seminar.

****Our seminars are listed in the Academy/Seminars area of the website.

*****Seminar titles may vary by country. Not all seminars are available in every country.

Seminars and training modules

V2.2.0.2 ©2016/07/07 by B&R. All rights reserved.
All registered trademarks are the property of their respective owners.
We reserve the right to make technical changes.



TM210TRE.425-ENG