CODESYS V3: Installation and Getting Started

User Documentation

Version: 17.0
Template: templ_tecdoc_en_V1.0.docx
File name: CODESYS Installation and Start.docx

CONTENT

1 Welcome

Welcome to the CODESYS V3 Development System!

The CODESYS Development System V3 is part of the CODESYS automation software for the engineering of control systems. Compliant with the IEC 61131-3 standard, CODESYS supports all standard IEC programming languages, and also allows for the inclusion of C routines and supports object-orientated programming. In combination with the CODESYS Control Win V3 runtime, it allows for multi-device and multi-application programming. The component-based structure allows for a customer-specific configuration and extension of the user interface.

2 System Requirements and Installation

Before you install CODESYS, be aware of the system requirements described in the CODESYS Store.

Run the file `Setup_CODESYSV<Version>.exe` to start the installation wizard which will guide you through the installation.

3 Recommendations for Data Protection

In order to minimize the risk of data security breaches, we recommend the following organizational and technical measures for the system that will run your applications:

Whenever possible, avoid exposing PLCs and controller networks to public networks and Internet. Use additional data link layers for protection, such as a VPN for teleaccess, and install firewall mechanisms.

Restrict access to authorized persons only, and change any existing default passwords during the initial commissioning, and change them regularly.

If you still want to publish your web visualization, then we strongly recommend that you assign it at least a simple password protection to prevent access to the functionality of your PLC over the Internet. (As an example, see the project "SimpleWebvisuLogin.project" which is provided with the standard installation setup of the programming system.)

Use the latest versions of Gateway Server and the Web Server. A security hole was detected for the Gateway Server and the Web Server up to V3.4 SP4 Patch 2 (US ICS CERT: ICS-ALERT-12-097-02). This was fixed in V3.5 and later.

4 Start

Start CODESYS from the Start menu (by default, the path is Programs –> CODESYS <version>).

You can also click the CODESYS icon that is located on the desktop after installation.

5 Help

Help is provided in the **Help** menu, as well as the context-sensitive <F1> key. Currently, the web-based online help opens by default. This can be disabled in the CODESYS options so that the offline help (CHM format) is used.

6 Creating and Running a Project

See in the following a description on how to create a simple project containing a PLC program. Moreover, it describes how to download this program via a Gateway Server to the PLC (target system) and run and monitor it. The CODESYS Control Win V3 controller target runtime used for the example is installed with the standard CODESYS setup.

The sample program is created in structured text and consists of a program `PLC_PRG` and a function block `FB1`.`PLC_PRG` contains a counter variable `ivar` and calls function block `FB1` . `FB1` receives a value from `PLC_PRG` at the input in, adds "2" to it, and displays the result in the output `out`. Then `out` is read by `PLC_PRG`.

(Note that the following description refers to the default configuration of the user interface which is available after a default installation of the development system.)

6.1 Start CODESYS and create a project

(1) Start CODESYS

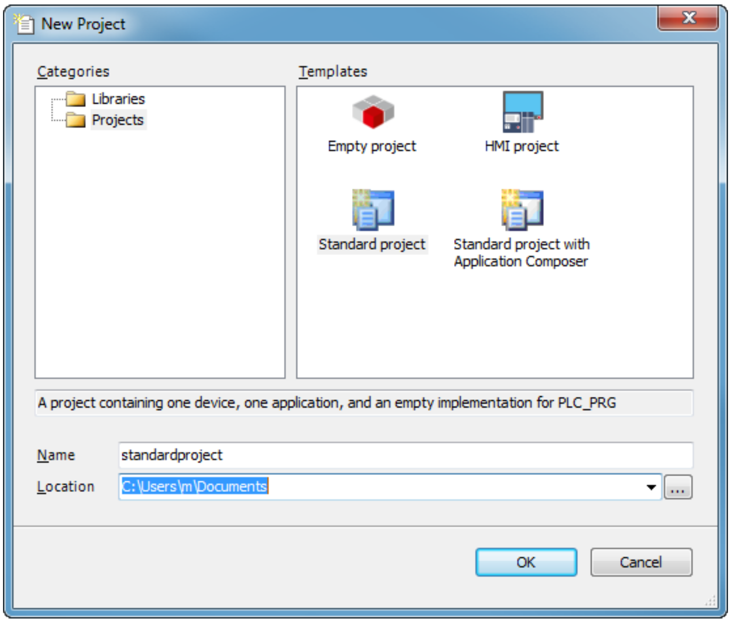
In the Start menu of your computer, click the following path:

Programs > 3S CODESYS > CODESYS > CODESYS <version>

You can also start CODESYS by clicking the CODESYS icon which is available on the desktop after installation.

(2) Create a project

To create a new project, click the *New Project* command in the *File*:



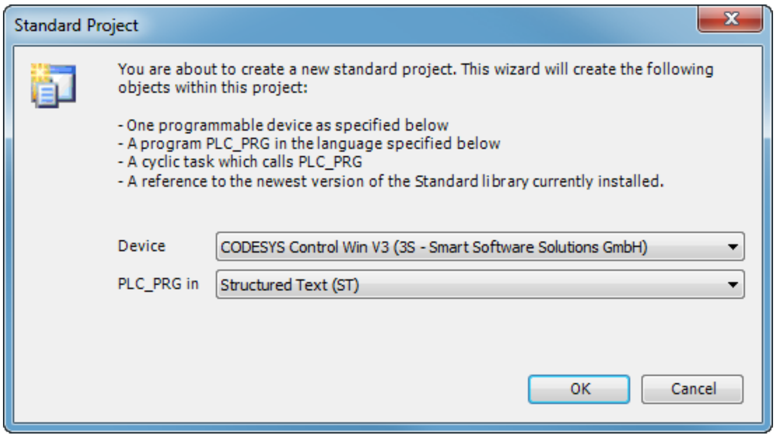
A project containing one device, one application, and an empty implementation for PLC_PRG

Name: standardproject

Location: C:\Users\m\Documents

In the *New Project* dialog, select *Standard project* template and specify a *Name* and *Location* for the project file. Click *OK* to confirm.

The *Standard Project* dialog opens:



You are about to create a new standard project. This wizard will create the following objects within this project:

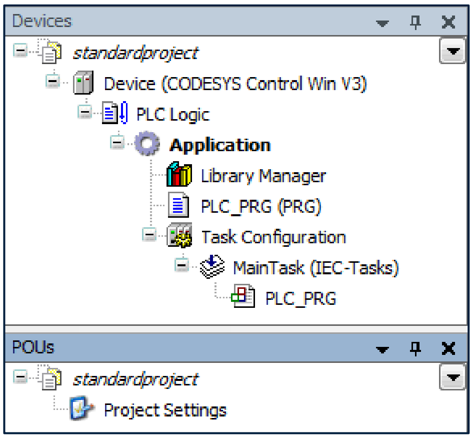
- One programmable device as specified below
- A program PLC_PRG in the language specified below
- A cyclic task which calls PLC_PRG
- A reference to the newest version of the Standard library currently installed.

Device: CODESYS Control Win V3 (3S - Smart Software Solutions GmbH)

PLC_PRG in: Structured Text (ST)

Select the *CODESYS Control Win V3* device and *Structured Text (ST)* implementation language for the automatically created `PLC_PRG`. Click *OK* to confirm.

Now the project name is displayed in the title bar of the CODESYS user interface and as a symbolic root node in the POU view and the device view.



The POU view contains the *Project Settings*.

The device view shows a tree structure with a *Device* of type *CODESYS Control Win V3* with an *Application* inserted below it. This contains the ST program *PLC_PRG* and the required *Task Configuration* with a *MainTask* to control the `PLC_PRG`.

A *Library Manager* with some standard libraries is already inserted. This includes, for example, the `IoStandard.library` which is required for the I/O configuration, as well as the `Standard.library` which prepares all functions and function blocks that are required by the IEC 61131-3 standard as basic POUs for an IEC programming system.

(The additional *PLC Logic* node below the *Device (CODESYS Control Win V3)* node is just a symbolic node which indicates that the device is "programmable".)

You can rename the symbolic device name. Select the item and click the spacebar to open an editing field to change the name. For the example here, change it from *Device* to *PLCWinNT*.Write a PLC program

6.2 Write a control program

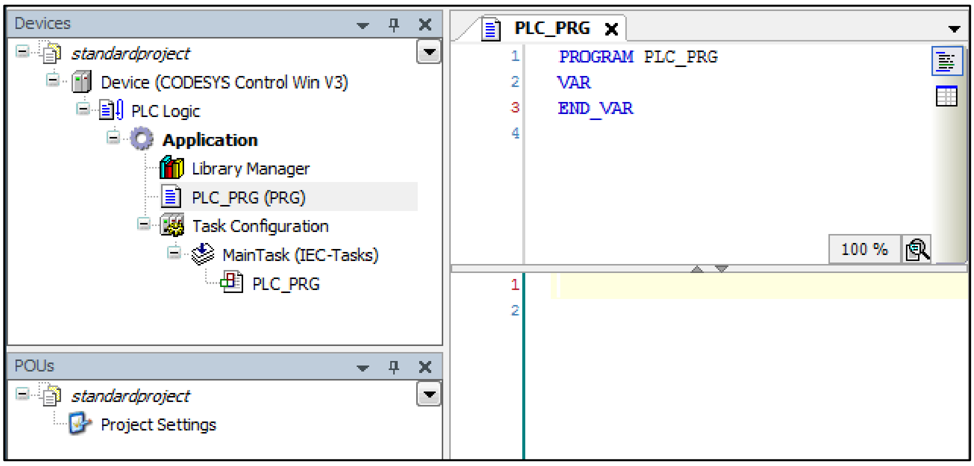
(3) Declare variables in PLC_PRG

In the device tree, double-click the *PLC_PRG* object.

The **ST Editor (Structured Text)** opens a window for `PLC_PRG` in the middle of the user interface. The editor consists of a declaration part (above) and an implementation part (below), separated by an adjustable screen divider.

In the textual view shown below (a tabular view is also possible), the first line of the declaration part automatically contains the POU type and name (`PROGRAM PLC_PRG`), which is followed by the keywords `VAR` and `END_VAR` which will encompass the variable declaration.

The implementation part is still blank:



```
PROGRAM PLC_PRG
VAR
END_VAR
```

In the declaration part, place the cursor after `VAR` and press the enter key to insert a blank line. Type in the following declarations here: `ivar` and `erg` of type `INT`, as well as `fbinst` of type `FB1:`.

```
PROGRAM PLC_PRG
VAR
  ivar: INT;
  fbinst: FB1;
  erg: INT;
END_VAR
```

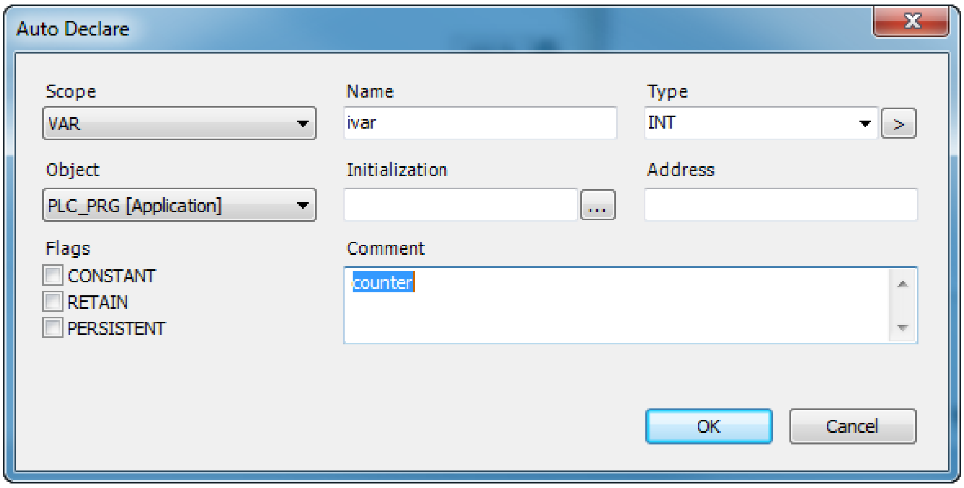
You can also type in lines of code without any previous declarations in the implementation part. Then when you set the focus in a variable name, a light bulb symbol will appear, which you can click to get the declaration possibilities: see (4).

(4) Type in program code into the implementation part of PLC_PRG

In the implementation part of `PLC_PRG`, type in the following lines of code:

```
ivar := ivar+1;            // counter
fbinst(in:=11, out=>erg);  // call function block of type FB1
                           // with input parameter "in"
                           // output is written to "erg"
```

If you have inserted the lines of code without having previously inserted the declarations, then now use the smart tag function. In the implementation part, set the cursor in a variable identifier und click the 💡 symbol below it. Click *Declare variable* to open the following dialog for you to declare a variable:



Scope: VAR

Name: ivar

Type: INT

Object: PLC_PRG [Application]

Initialization:

Address:

Flags

☐ CONSTANT

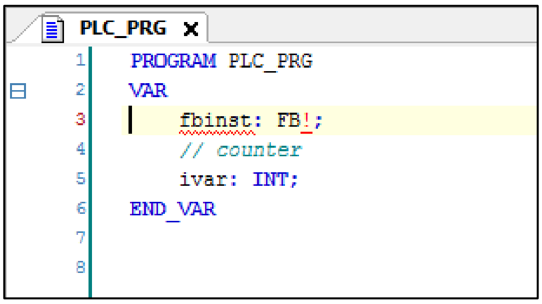
☐ RETAIN

☐ PERSISTENT

Comment: counter

The variables name and its scope, as well as the current POU (*Object*) are provided automatically. Now specify the desired data type according to the declaration described in (3).

Click *OK* to confirm the dialog. Then the declaration of `erg` is inserted in the declaration part. The comment is displayed above the declaration line:



```
PROGRAM PLC_PRG
VAR
    fbinst: FB!;
    // counter
    ivar: INT;
END_VAR
```

(5) Create an additional POU (ST function block FB1)

Add a function block to the application that adds 2 to the value contained in the input variable `in` and sends the result to the output variable `out`:

In the device tree, select the *Application* object. In the context menu or in the *Project* menu, click *Add Object*.

Select *POU*. In the *Add POU* dialog, type in the name `FB1` and for *Type*, select the *Function block* option.

For **Implementation language**, select *Structured Text (ST)*. Click the *Add* button to confirm.

Another editor window opens where the function block `FB1` can be edited. Declare the following variables there:

```
FUNCTION_BLOCK FB1
VAR_INPUT
  in:INT;
END_VAR
VAR_OUTPUT
  out:INT;
END_VAR
VAR
  ivar:INT:=2;
END_VAR
```

In the implementation part of `FB1`, type in the following line of code:

```
out:=in+ivar;
```

6.3 Insert and configure the objects for running and controlling the program on the PLC

(6) Start the Gateway Server and the PLC

Start the Gateway Server:

By default, the *GatewaySysTray* program is installed with the installation of CODESYS. You can use this program to communicate with the Gateway Server. You can call it by means of the ● system tray.

The Gateway Server is started automatically as a service when the system starts. Check whether or not the ● symbol is displayed in the system tray at the bottom edge of the screen. When the ● symbol is displayed, the Gateway is stopped.

Start the PLC:

By default, the *ControlSysTray* program is installed with the installation of CODESYS. You can use this program to show the CODESYS Control service. The controller (CODESYS Control Win V3) is available as a service when the system starts. In V3.5 SP2 and later, the control service is no longer automatically started when the system starts. To start the PLC, do the following:

In the system tray, right-click the ▥ icon. A context menu opens.

In the context menu, click *Start PLC*. The icon changes to ▥.

Note: Check the options for accessing the PLC. PLCs must never be accessible from the Internet or untrustworthy networks under any circumstances. In particular, the programming ports of the PLC must never be accessible from the Internet without security protection (mostly UDP ports 1740..1743 and TCP ports 1217 + 11740 and the PLC-specific ports). If access from the Internet must nevertheless be granted, then it is imperative that a secure method be selected to connect to the PLC (e.g. VPN).

(7) Set the “Active Application”

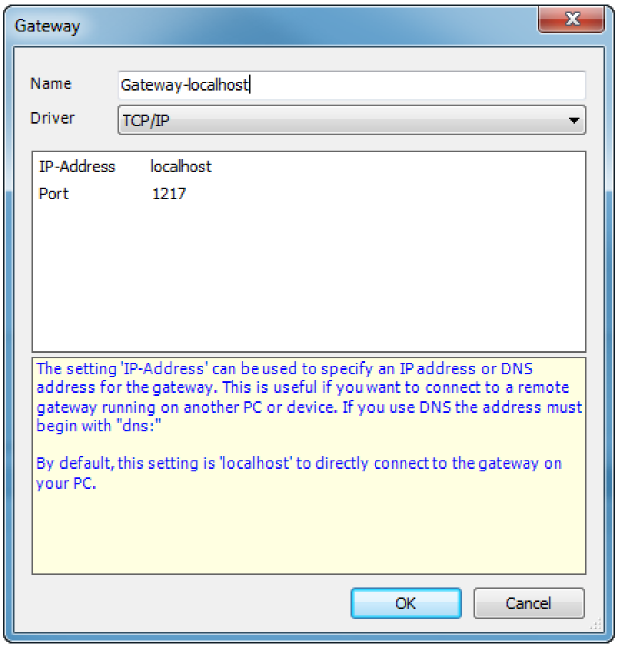
In the Devices window the name ‘Application’ is displayed in bold letters. This means that this application is set as “active application”. Thus all commands and actions concerning the communication with the PLC will refer to this application. In order to set an application as “active one” select the application entry in the Devices window and choose command **Set Active Application** from the context menu.

(8) Configure the communication channel to the PLC

In the device view, double-click *PLCWinNT (CODESYS Control Win V3)*. The *Device* dialog opens with the *Communication Settings* tab. Here you have to configure the connection between the PLC (device) and the development system. To do this, follow the steps below:

If this is your first communication configuration with CODESYS V3, then you have to define the local Gateway Server first. If you have already done this, then it will be displayed in the dialog as seen in the image below. In this case, you can jump to the definition of the *communication channel to the device*.

The Gateway Server is provided with the CODESYS setup. In the *Gateway* menu, click *Add New Gateway* to open the following Gateway dialog:



Name: Gateway-localhost

Driver: TCP/IP

IP-Address: localhost

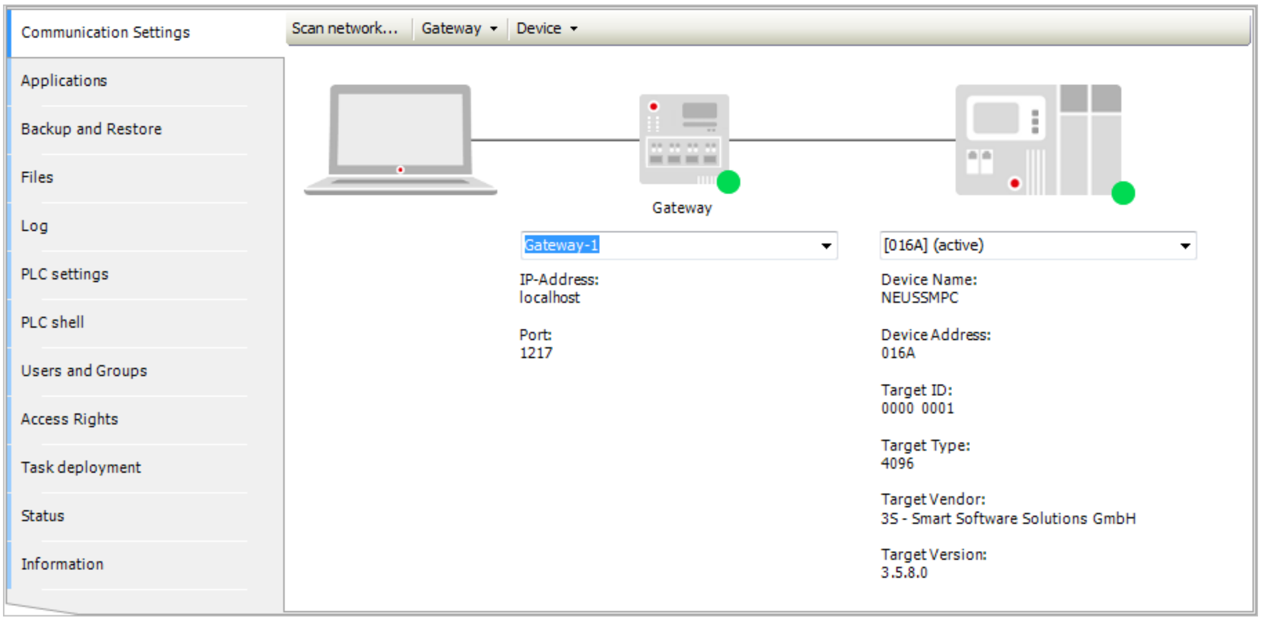
Port: 1217

The setting 'IP-Address' can be used to specify an IP address or DNS address for the gateway. This is useful if you want to connect to a remote gateway running on another PC or device. If you use DNS the address must begin with "dns:"

By default, this setting is 'localhost' to directly connect to the gateway on your PC.

Specify a symbolic *Name* for the gateway. Set the driver type to *TCP/IP* and specify the IP address as *localhost*. (In the *Value* column, select the field. Press the space bar to open an input field.) Leave the *Port* unchanged and click *OK*.

The Gateway is entered in the field on the left part of the *Communication Settings* dialog and added to the list box for *Select the Network Path to the Controller*. When the gateway is running correctly, a solid green circle is displayed at the symbol of the entry. Otherwise, a red circle is displayed.



Gateway-1

IP-Address:
localhost

Port:
1217

[016A] (active)

Device Name:
NEUSSMPC

Device Address:
016A

Target ID:
0000 0001

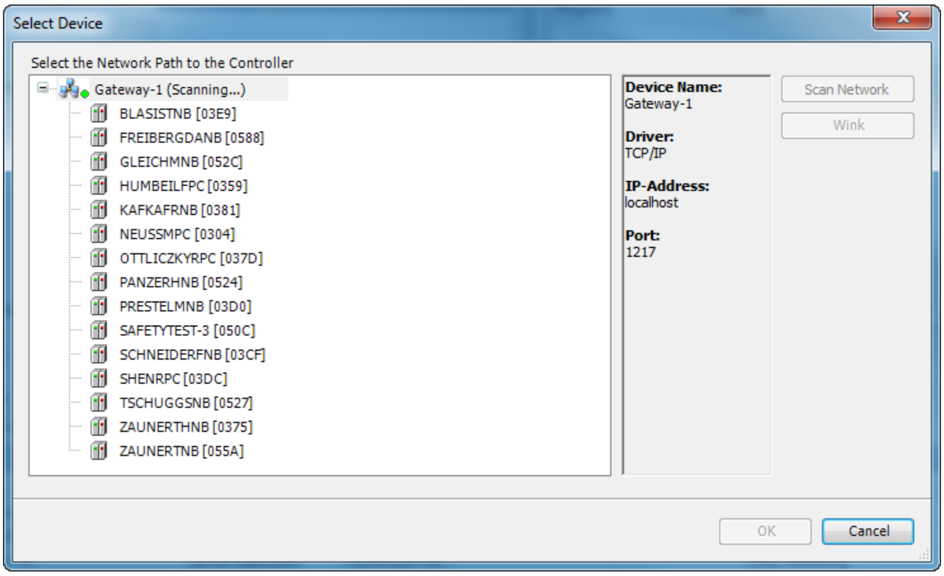
Target Type:
4096

Target Vendor:
3S - Smart Software Solutions GmbH

Target Version:
3.5.8.0

Now define the communication channel to the device, which is then used via the set Gateway.

Click the *Scan Network* button to search for all devices in the local network that you can communicate with.



Select Device

Select the Network Path to the Controller

- Gateway-1 (Scanning...)
  - BLASISTNB [03E9]
  - FREIBERGDANB [0588]
  - GLEICHMNB [052C]
  - HUMBEILFPC [0359]
  - KAFKAFRNB [0381]
  - NEUSSMPC [0304]
  - OTTLICZKYRPC [037D]
  - PANZERHNB [0524]
  - PRESTELMNB [03D0]
  - SAFETYTEST-3 [050C]
  - SCHNEIDERFNB [03CF]
  - SHENRPC [03DC]
  - TSCHUGGSNB [0527]
  - ZAUNERTHNB [0375]
  - ZAUNERTNB [055A]

Device Name: Gateway-1

Driver: TCP/IP

IP-Address: localhost

Port: 1217

Select the device with the name of your computer and click *OK*.

This sets the channel as active, which means that all communication actions apply specifically to it. Remember this later when you have multiple communication channels in the project.

Click *OK* to close the *Communication Settings* dialog and apply the settings.

6.4 Run and watch the application on the PLC

(9) Compile and download the application to the PLC

If you only want to perform a syntactic validation of your "active" application, then use the *Build* command (in the context menu when the application object is selected, or in the *Build* menu). Caution: No code will be generated in this case. The results of the validation are displayed in the message view, which is located in the bottom middle area of the user interface by default.

Even without the preceding validation, you can log in to the PLC. (First make sure that the PLC is running: the symbol in the system bar is in color).

Click *Login* (in the context menu when the application object is selected, or in the *Online* menu). A dialog opens:

"*Application ‘Application‘ does not exist on device ‘PLCWinNT‘. Do you want to create it and continue with download?*"

Click *Yes* to start the build and download of the application.

The build messages are displayed in the message view. In the case that the example has been configured correctly, no compile errors are expected and the application can now be started on the PLC (see (11)).

(10) Run and watch an application

When the "standardproject" has been created and downloaded as described in the previous steps, the error-free *Application* can be used on the CODESYS Control Win V3 ("PLCWinNT") control device.

Start the application on the PLC:

Select the application. In the context menu, click *Start*. The program is running. In the device view, the entries for the PLC and the application are highlighted in green and *[run]* is displayed after *Application*.

Watch the application:

There are several ways to monitor the variables of the application program and to influence them in the watch view:

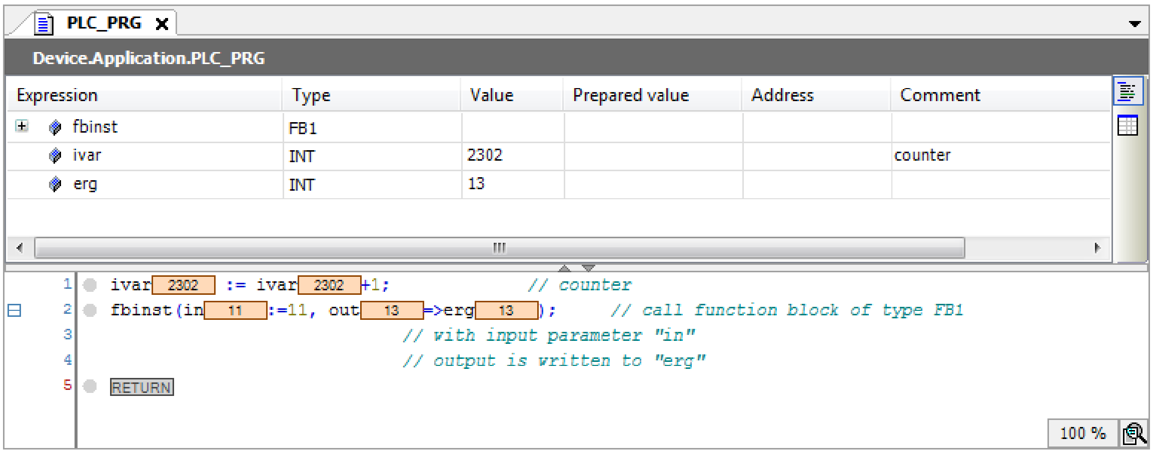
1. Online views of individual POUs
2. Write and force variable values
3. Defined variable lists in separate watch lists

1. Online view of a POU

This view of a POU shows the current values of the watchable expressions contained in it in a table in the declaration part, and – if enabled – also in the implementation part in the form of "inline monitoring".

To open the online view, in the device tree double-click *PLC_PRG*, or select it and in the context menu click *Edit Object*.

In the bottom part of the view, you see the lines of code as specified in offline mode. They are supplemented by the little inline monitoring views after each variable which show the current value. In the top part, a table shows the watchable expressions of the POU. This means the corresponding variables with *Type* and current *Value*.



PLC_PRG

Device.Application.PLC_PRG

Expression	Type	Value	Prepared value	Address	Comment
⊞ fbinst	FB1				
ivar	INT	2302			counter
erg	INT	13			

```
ivar 2302 := ivar 2302 +1;            // counter
fbinst(in 11 :=11, out 13 =>erg 13 );      // call function block of type FB1
                         // with input parameter "in"
                         // output is written to "erg"
RETURN
```

2. Writing and forcing of variables:

You can write or force a *Prepared Value* on the PLC for example for the variable `ivar`, which means that `ivar` gets this value on the PLC at the beginning of the next processing cycle (and for *Forces*). To do this, in the *Prepared Value* column select the field, press the spacebar to open an input field, specify an integer value, and press the enter key or click outside of the field to close the field. Then, in the *Debug* menu, click *Write Values* or *Force Values*. You see the corresponding result in the *Value* column.

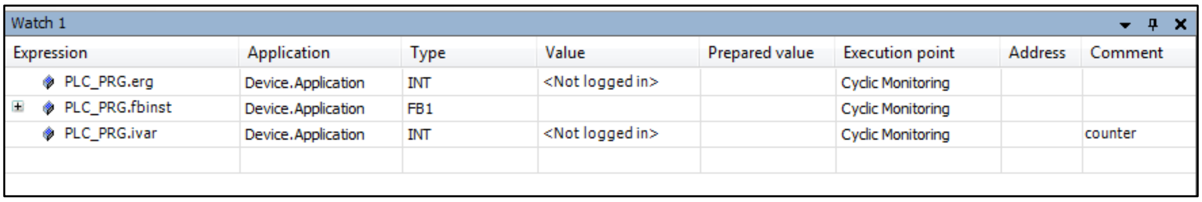
3. Using watch lists:

Watch lists can be used to compile in different lists the expressions from the application to be watched. This can be useful for debugging purposes when specific variables need to be watched at a glance.

In the *View* menu, click "*Watch* - *Watch 1*". The view opens.

In the *Expression* column, click in the first row to open an input field. Specify the path for the variable to be watched. We recommend that you use the Input Assistant for this (... button): "`PLCWinNT.PLC_PRG.ivar`". Press the enter key to close the input field. The data type of the variable is configured automatically in the row.

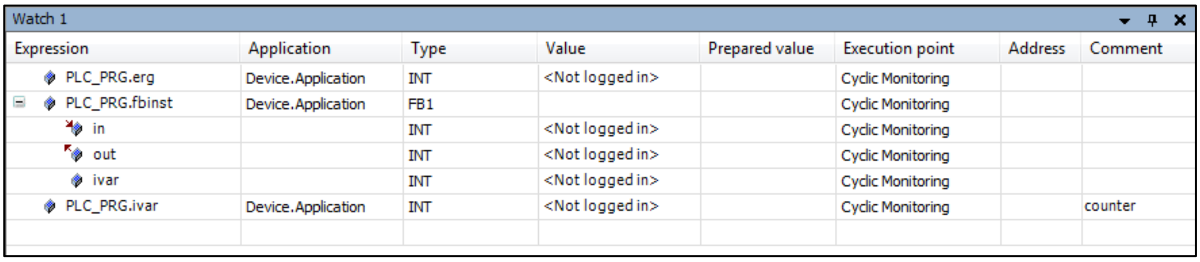
Insert more rows for other variables. The watch list displayed below contains only expressions from PLC_PRG, but of course any set of variables from all POUs of the project can be combined. Note in the case of instance variables, such as for those of `FB1` instances, that it is enough to specify the expression "PLC_PRG.fbinst". The individual parameters are inserted automatically and the corresponding rows can be opened by means of the plus sign at the front of each row. The current value of each expression is displayed in the value column:



Watch 1

Expression	Application	Type	Value	Prepared value	Execution point	Address	Comment
PLC_PRG.erg	Device.Application	INT	<Not logged in>		Cyclic Monitoring		
⊞ PLC_PRG.fbinst	Device.Application	FB1			Cyclic Monitoring		
PLC_PRG.ivar	Device.Application	INT	<Not logged in>		Cyclic Monitoring		counter

If you have not already done it, in the online menu, click *Start Application*. Now the program runs on the PLC and the current values are displayed in the *Value* column:



Watch 1

Expression	Application	Type	Value	Prepared value	Execution point	Address	Comment
PLC_PRG.erg	Device.Application	INT	<Not logged in>		Cyclic Monitoring		
PLC_PRG.fbinst	Device.Application	FB1			Cyclic Monitoring		
in		INT	<Not logged in>		Cyclic Monitoring		
out		INT	<Not logged in>		Cyclic Monitoring		
ivar		INT	<Not logged in>		Cyclic Monitoring		
PLC_PRG.ivar	Device.Application	INT	<Not logged in>		Cyclic Monitoring		counter

Writing and forcing of values is possible in the same way as described above in 2.

To disconnect from the PLC, select the application object and in the context menu click *Logout*.

6.5 Debug an application

(11) Set breakpoints and step into a program

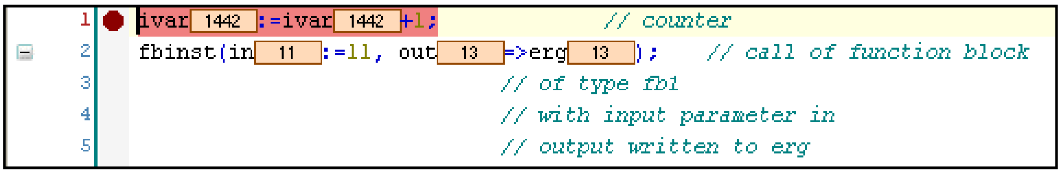
In online mode, you can set breakpoints where any program execution should be halted.

When a breakpoint is reached, the program can be processed in steps. At each breakpoint and in each step, you can check the current value of the variables in the watch views.

Try out the following:

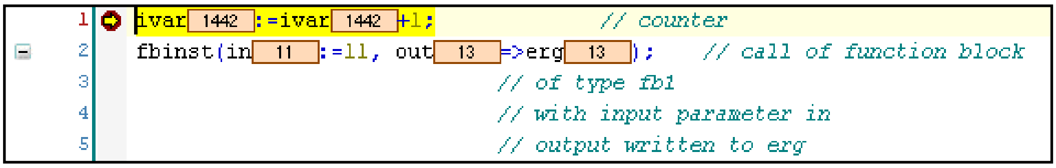
In `PLC_PRG`, set the cursor in line 1. In the *Debug* menu, press <F9> to execute the *Switch Breakpoint* command.

The breakpoint is displayed in the program. When the application is currently in the stop status, it will look like this:



```
1  ivar 1442 :=ivar 1442 +1;                       // counter
2  fbinst(in 11 :=11, out 13 =>erg 13 );     // call of function block
3                                               // of type fb1
4                                               // with input parameter in
5                                               // output written to erg
```

When the application is running, it halts at the breakpoint:



```
1  ivar 1442 :=ivar 1442 +1;                       // counter
2  fbinst(in 11 :=11, out 13 =>erg 13 );     // call of function block
3                                               // of type fb1
4                                               // with input parameter in
5                                               // output written to erg
```

Now you can press <F8> repeatedly to execute the *Step Into* command in the *Debug* menu and run the program in steps. This also steps into the function block instance. To skip this function block processing, instead of <F8> you can press <F10> to execute the *Step Over* command. The current variable values are displayed at the processing position just reached.

See also the *Breakpoints* dialog (*View* -> *Breakpoints*). The currently defined breakpoints are listed here and can be edited or new ones can be added.

Note that the breakpoint positions are saved even when you log out of the PLC. The next time you log in, they will be displayed as light red markers and can be reactivated.

Now you have built your first project with CODESYS V3 and run it in the PLC. For more information about using the development system, see the current help.

We wish you success in your programming, and we appreciate any feedback.

7 Remove, Modify, or Repair the Installation

Removing the installation

Normally, you can install a new version of CODESYS without having to uninstall the current version. If you want to uninstall specifically, then run the file `Setup_CODESYSV<version>.exe` as well. In the Installation Wizard, in the *Program Maintenance* dialog, select the *Remove* option and click *Next*. You can also use the general uninstall process in the System Control dialogs of your system.

Modifying the installation

If you have to change the installation, for example to change the installed components, run the file `Setup_CODESYSV<version>.exe`. In the Installation Wizard, in the *Program Maintenance* dialog, select the *Modify* option and click *Next* to run through the installation dialogs and modify the settings of the previous installation.

Repairing the installation

If there were any problems with the installation, then you can repair it. To do this, run the file `Setup_CODESYSV<version>.exe`. In the Installation Wizard, in the *Program Maintenance* dialog, select the *Repair* option and click *Next*. The installation is performed again with the same settings as configured in the previous installation.

Change History

Version	Description	Author	Date
0.1	Issued	MN	16.05.2006
0.2	Description of sample project	MN	26.01.2007
0.3	Update of sample project, Review	MN/MW	08.02.2007
0.4	Corrections acc. To Review	MN	09.02.2007
1.0	Release	MN	09.02.2007
2.0	Release after corrections in chap.6 concerning device name at several positions	MN	21.02.2007
3.0	Release after adaptations in chap.6.3 (default device) and 6.4 (monitoring possibilities)	MN	26.03.2007
4.0	Release after adaptations to CODESYS V3.1.3.0 according to the German document version 4.0	MN	14.11.2007
4.1	Adaption to CODESYS V3.2 acc. To help	SE	20.02.2008
5.0	Formal Review, Release	MN	28.02.2008
6.0	Adaption to CODESYS V3.2.0.4 acc. To help, Release	MN	04.07.2008
7.0	Adaption to CODESYS V3.4.0.0 acc. To help, Release	MN	19.03.2010
8.0	CDS-17521	MN	05.11.2010
9.0	Adaption to CODESYS V3.5.1.0 help, chap.4 (security recommendations) added, Release	MN	29.05.2012
9.1	CDS-29303	MN	19.09.2012
9.2	CDS-31487 (no automatic start of CODESYS Control Win)	MN	09.11.2012
10.0	Release for CODESYS V3.5.2.0	MN	28.11.2012
11.0	New template applied (templ_tecdoc_en_V1.0.docx); Formal Review, Release	MN	03.03.2014
11.1	Update for CODESYS V3.5.8.0	MN	27.11.2015
12.0	Release for CODESYS V3.5.8.0	MN	30.11.2015
12.1	Geographic Restrictions added	MN	12.09.2016
13.0	Release	MN	12.09.2016
14.1	Geographic Restrictions removed	MN	17.10.2016
15.0	Release	MN	26.06.2018
15.1	CDS-60855 Adjust Terminology for “online help”	MaH	29.01.2019
16.0	Release after formal review	MaH	08.07.2019
16.1	Update: CDS-67427, CDS-67422, CDS-67019	MN	14.10.2019

Version	Description	Author	Date
17.0	Release after formal review	MN	20.04.2020