

# FactoryTalk® Design Studio™

TECHNICAL PRESENTATION

NOVEMBER 2025



expanding **human possibility**®



PUBLIC

SAAS ENABLED BY THE CLOUD



**The unified experience** (common login, context and data)

DESIGN



Streamlined automation  
system design

OPERATE



Simplified cloud-based  
operations management

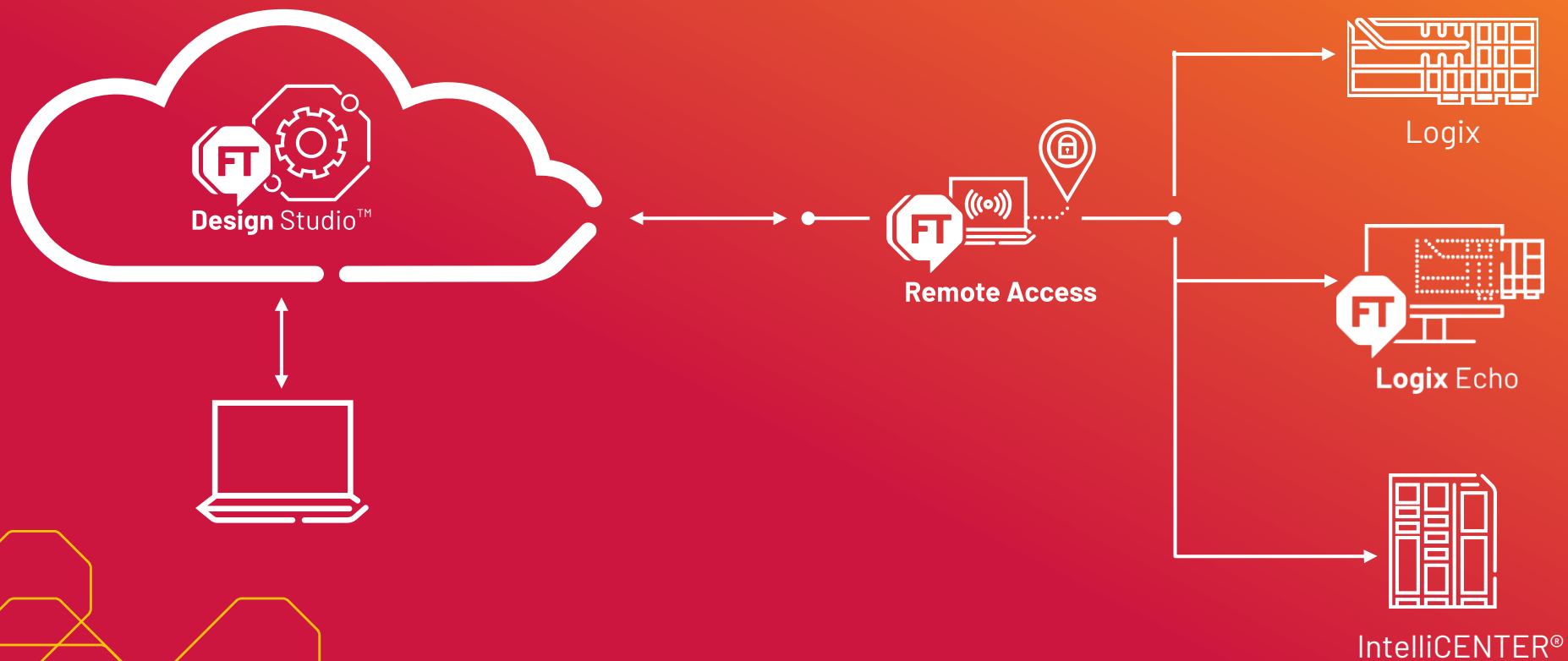
MAINTAIN



Asset management and  
predictive maintenance

**Edge Platform** (data context, applications, analytics)

# Cloud-based design for Logix controllers







# Future-ready industrial automation design

Empowering engineers with cloud-based tools, AI-driven productivity and modern Logix programming.

Built for what's next.



## MODERN

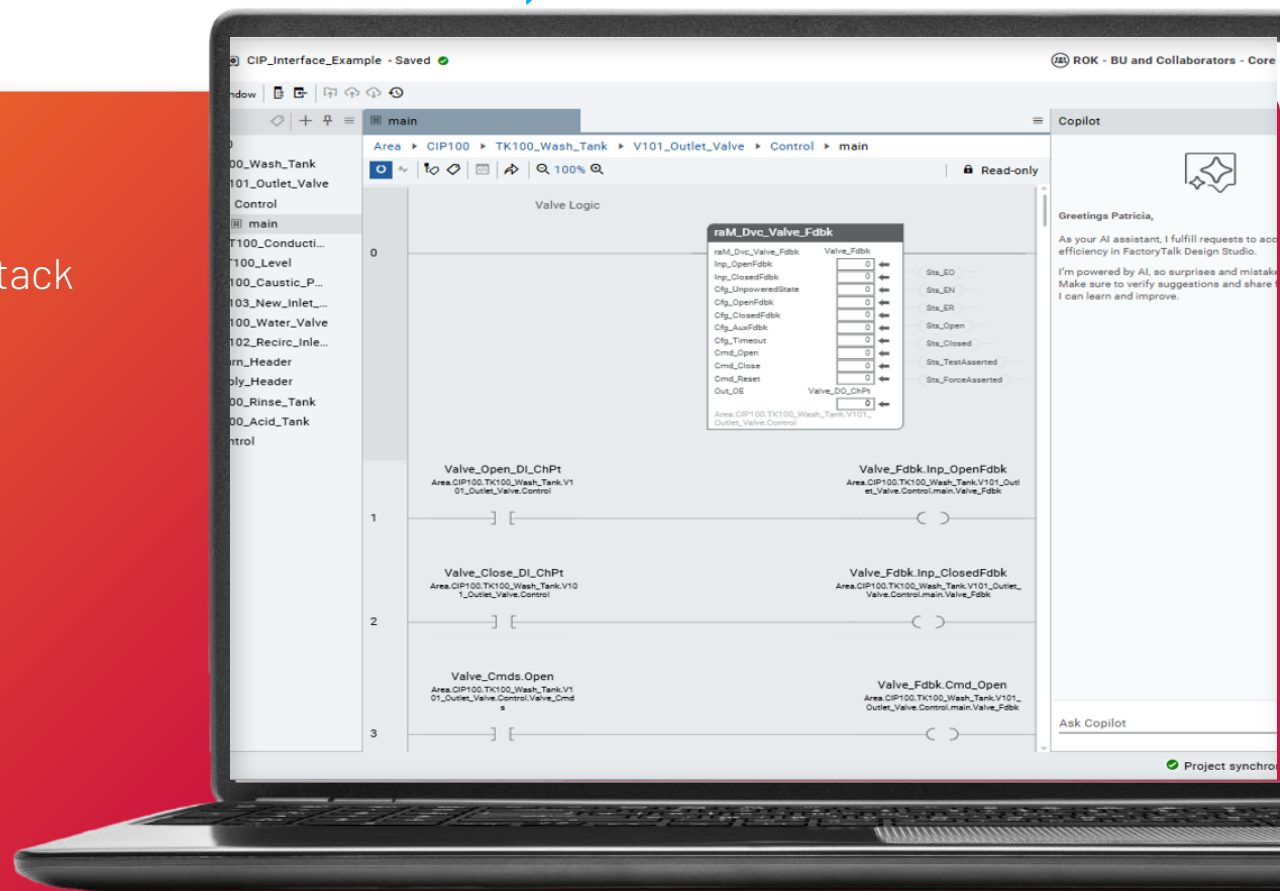
- Driving Logix programming into the future
- Enabling new capabilities with a modern technology stack
- Breakthrough productivity with Generative AI

## MULTI-USER

- Collaboration enabled by integrated version control
- Monitor and approve any project changes
- Simplified steps to build Logix systems

## MULTI-CONTROLLER

- Build a project with multiple Logix controllers
- Connect to multiple firmware revisions
- Deliver a system model abstracted from hardware





# Routine-scoped tags

Fully encapsulate routines for modular project creation and reusability

## BENEFITS

- Fully encapsulate routine code
- Interface with inputs/outputs
- Reuse routines with no program dependency

FT DesignStudio CIP\_Interface\_Example - Saved

File Edit View Window

System

- ▼ CIP100
  - ▼ TK100\_Wash\_Tank
    - ▼ V101\_Outlet\_Valve
    - ▼ CT100\_Conductivity
    - ▼ LT100\_Level
    - ▼ P100\_Caustic\_Pump
      - ▼ **Control**
        - main**
        - ▼ V100\_Water\_Valve
        - ▼ V102\_Recirc\_Inlet\_V...
      - Return\_Header
      - Supply\_Header
      - TK200\_Rinse\_Tank
      - TK300\_Acid\_Tank
      - zControl

Area ▶ CIP100 ▶ TK100\_Wash\_Tank ▶ P100\_Caustic\_Pump ▶ Control ▶ main

100%

Pump\_Cmds.On

\_auto  
Area.CIP100.TK100\_Wash\_Tank.P100\_Caustic\_Pump.Control.main

\_manual  
Area.CIP100.TK100\_Wash\_Tank.P100\_Caustic\_Pump.Control.main

Manual-Off-Auto control

PB\_Manual  
Area.CIP100.TK100\_Wash\_Tank.P100\_Caustic\_Pump.Control.main

PB\_Off  
Area.CIP100.TK100\_Wash\_Tank.P100\_Caustic\_Pump.Control.main

PB\_auto  
Area.CIP100.TK100\_Wash\_Tank.P100\_Caustic\_Pump.Control.main

main

Area ▶ CIP100 ▶ TK100\_Wash\_Tank ▶ P100\_Caustic\_Pump ▶ Control ▶ main

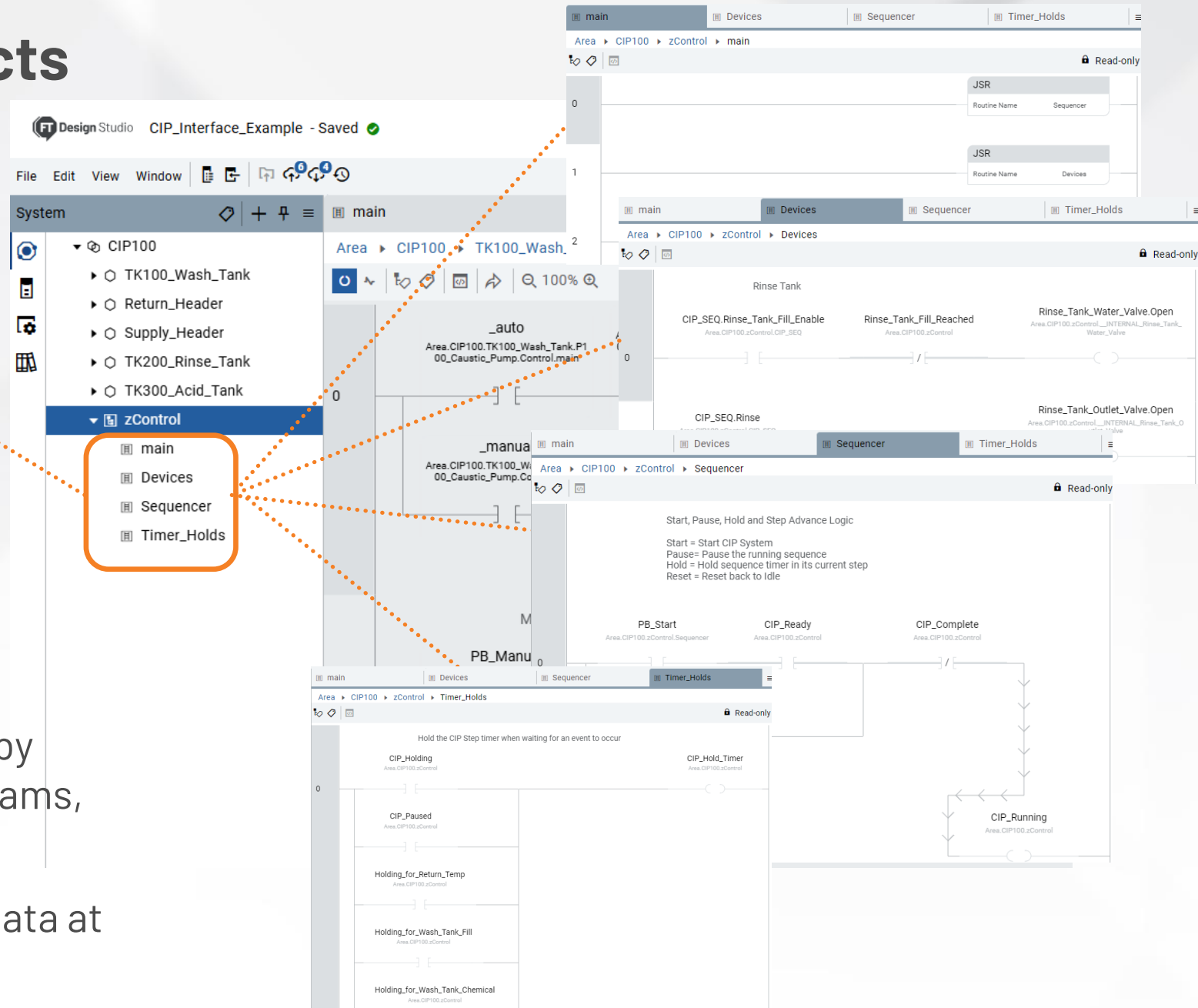
Name	Data Type	Usage	Value	Style	Description
_auto	BOOL	LOCAL	0	DECIMAL	
_manual	BOOL	LOCAL	0	DECIMAL	
_off	BOOL	LOCAL	0	DECIMAL	
▶ myONS	INT	LOCAL	0	DECIMAL	
PB_auto	BOOL	LOCAL	0	DECIMAL	
PB_Manual	BOOL	LOCAL	0	DECIMAL	
PB_Off	BOOL	LOCAL	0	DECIMAL	
Add Tag...					

# FactoryTalk® smart objects

Content reuse and encapsulation with FactoryTalk® Smart Objects

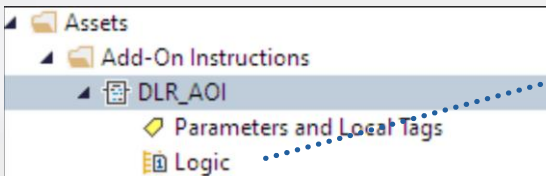
## BENEFITS

- Represent complex functionality by wrapping multiple routines, programs, and other objects together
- Simplify interactions by passing data at the object level

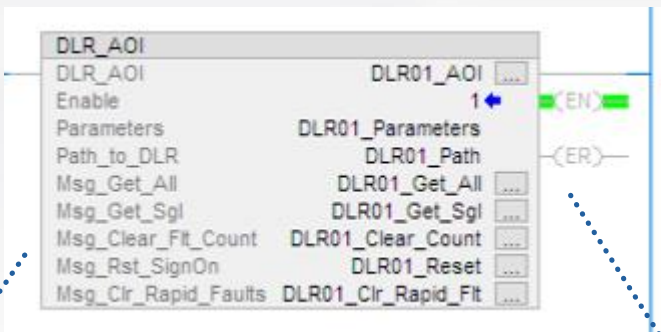


# Smart object comparison to A0Is

## Add-on instructions



1 routine only



Managed as an instruction

Defined once in a project and can be used multiple times

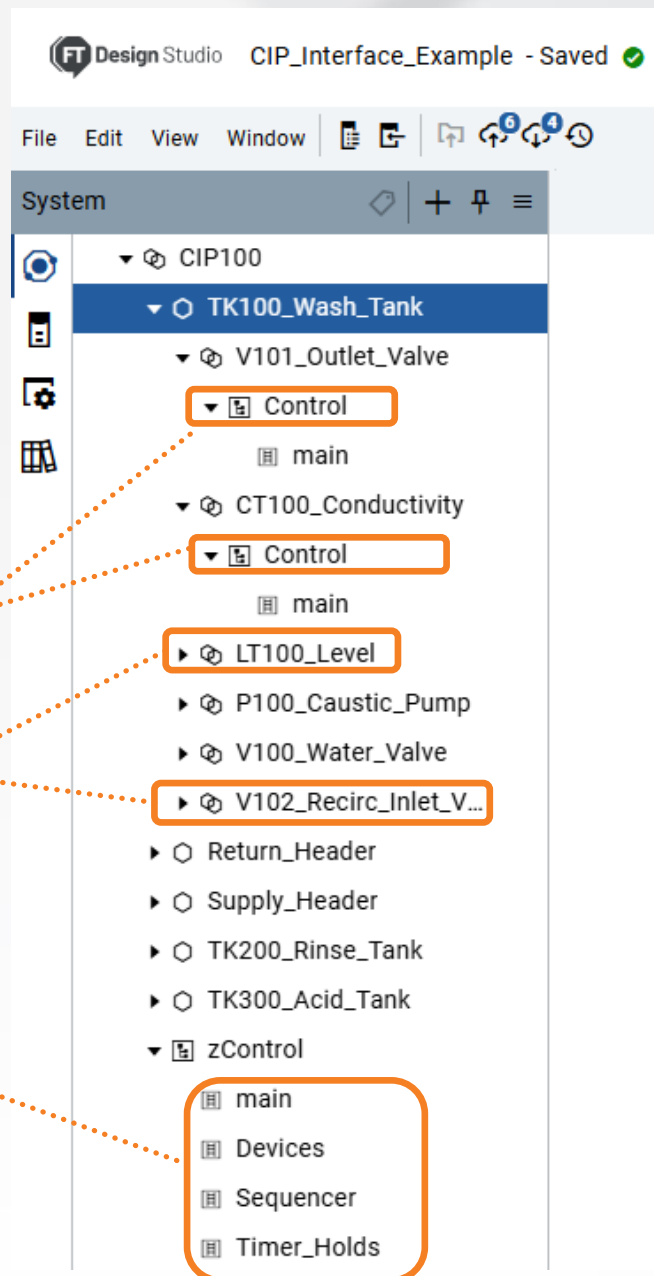
Smart objects allow for logical code segmentation, which eases development and maintenance efforts

Multi-program

Multi-object

Multi-routine

## Smart objects





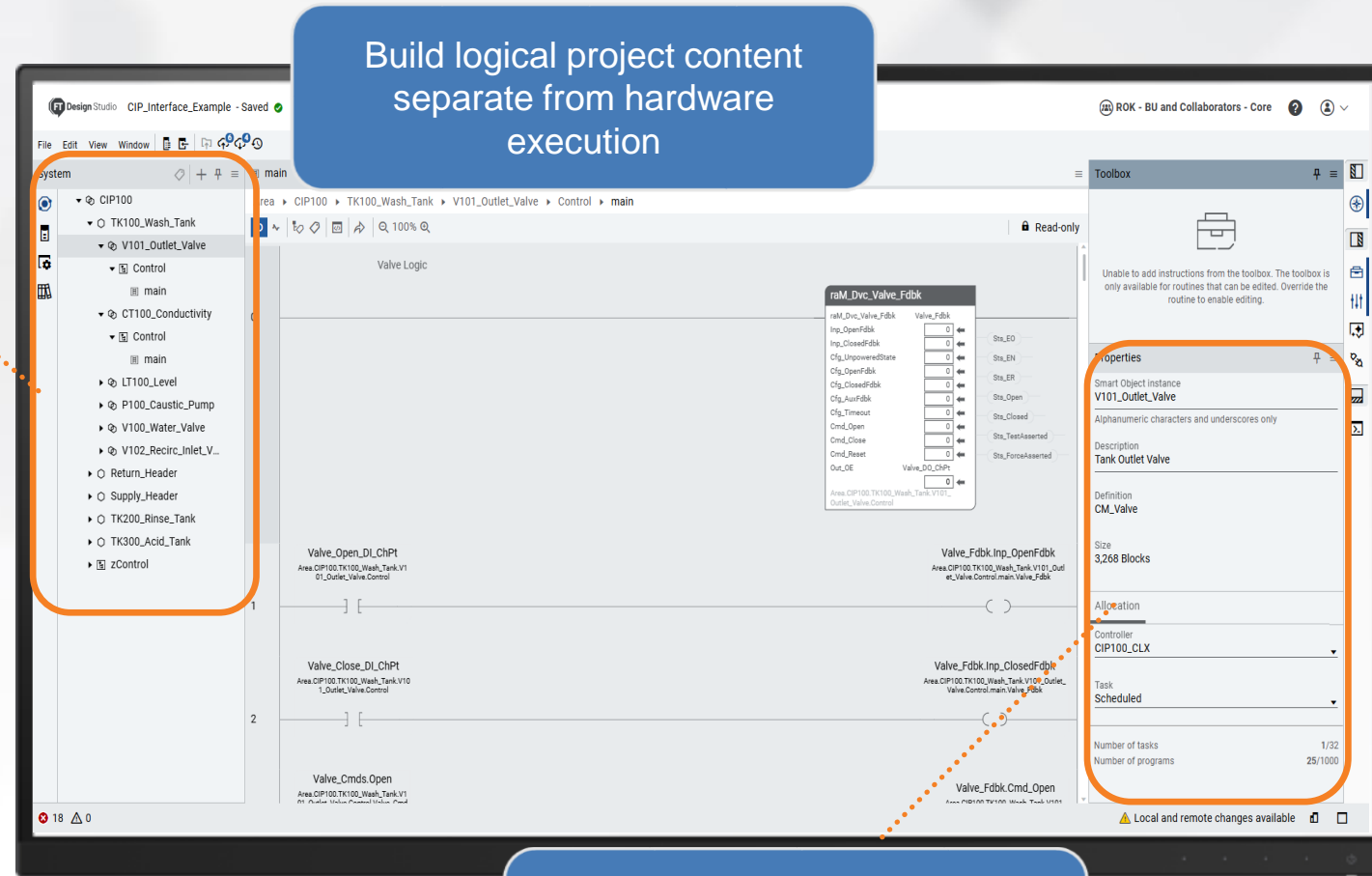
# Logical, hardware-abstracted design

Deliver intuitive projects by building systems based on what they're doing, not where they're executing

## BENEFITS

- Create logical system models that are not dependent on controller hierarchy
- Flexibly allocate logical content to controllers through development lifecycle

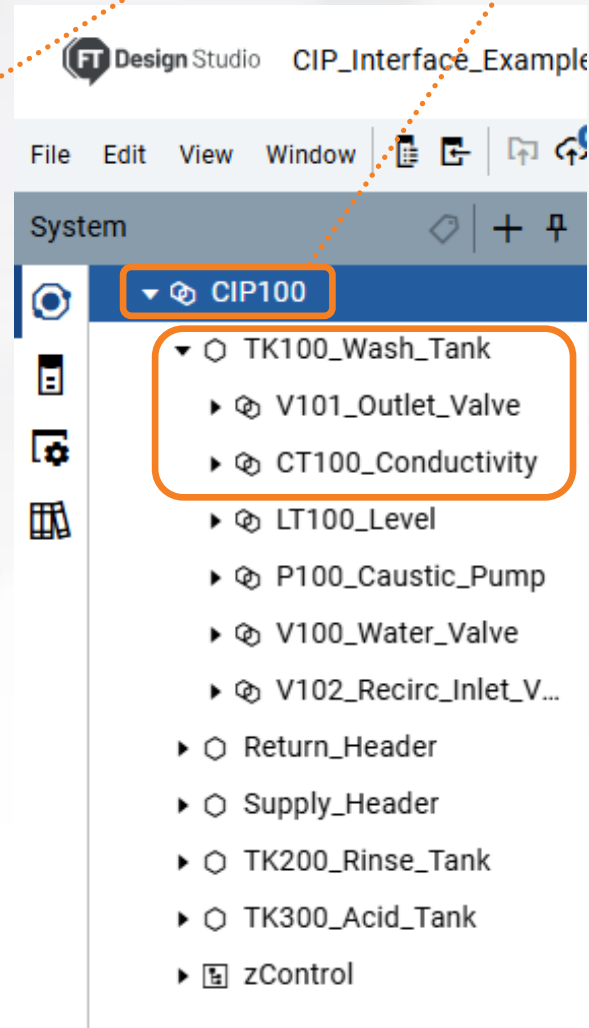
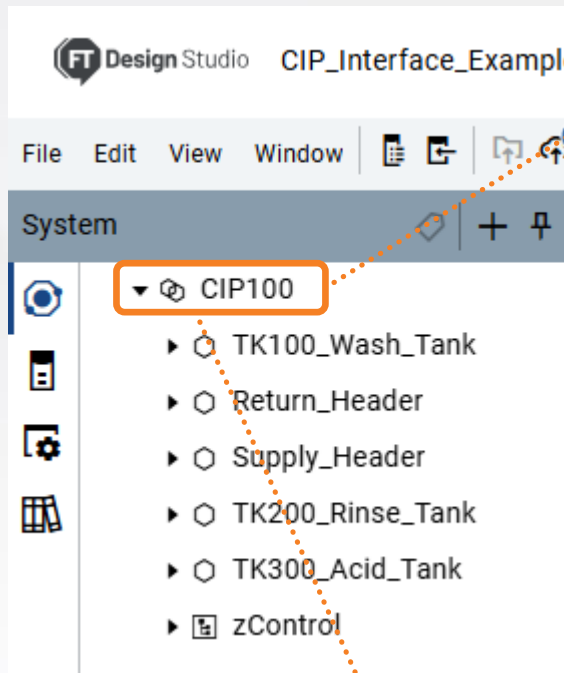
Build logical project content separate from hardware execution



Map code to controllers to create execution

# Creating a logical model

Highest-level code container that houses the application code

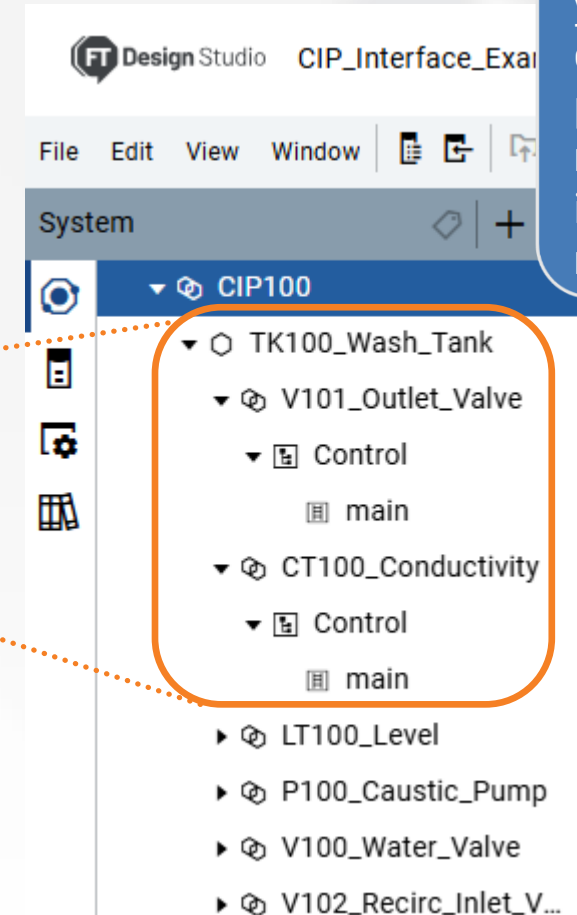
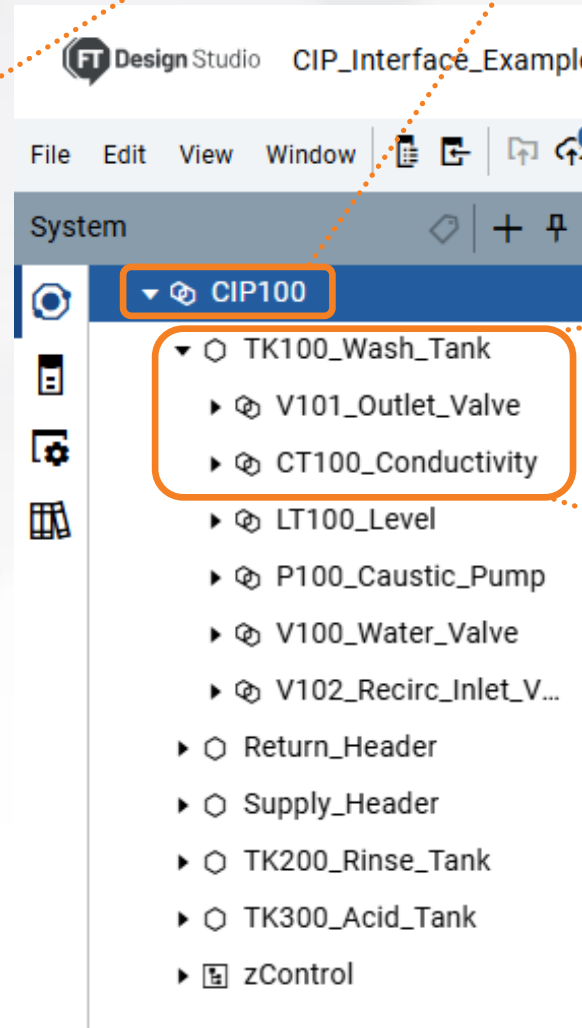
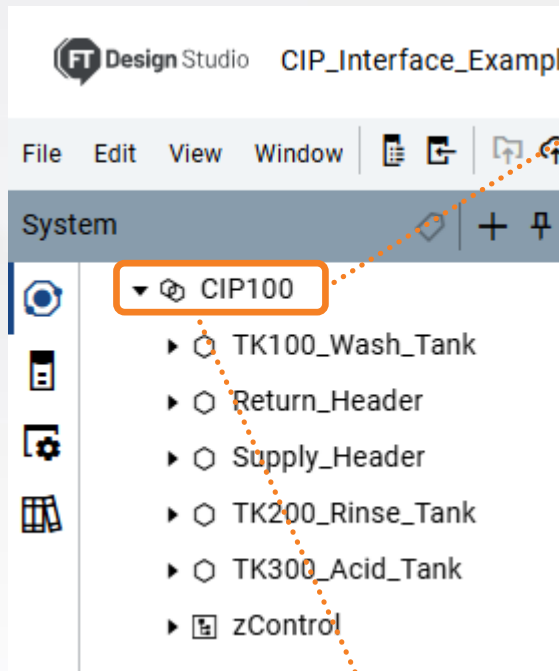


To be part of the logical model in a FactoryTalk® Design Studio™ project, it must reside in a smart object.

# Creating a logical model

Highest-level code container that houses the application code

Smart objects allow you to establish parent-child relationships in the logical model.



To be part of the logical model in a FactoryTalk® Design Studio™ project, it must reside in a smart object.

Code is neatly segmented and encapsulated for easy transportability / reuse



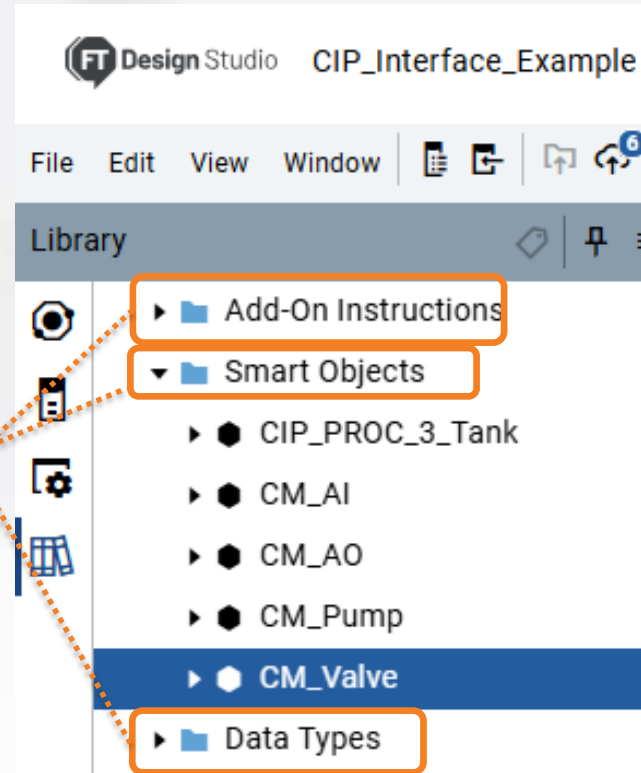
# Integrated reusable content libraries

Reduce repetitive steps and increase standardization with integrated library tools

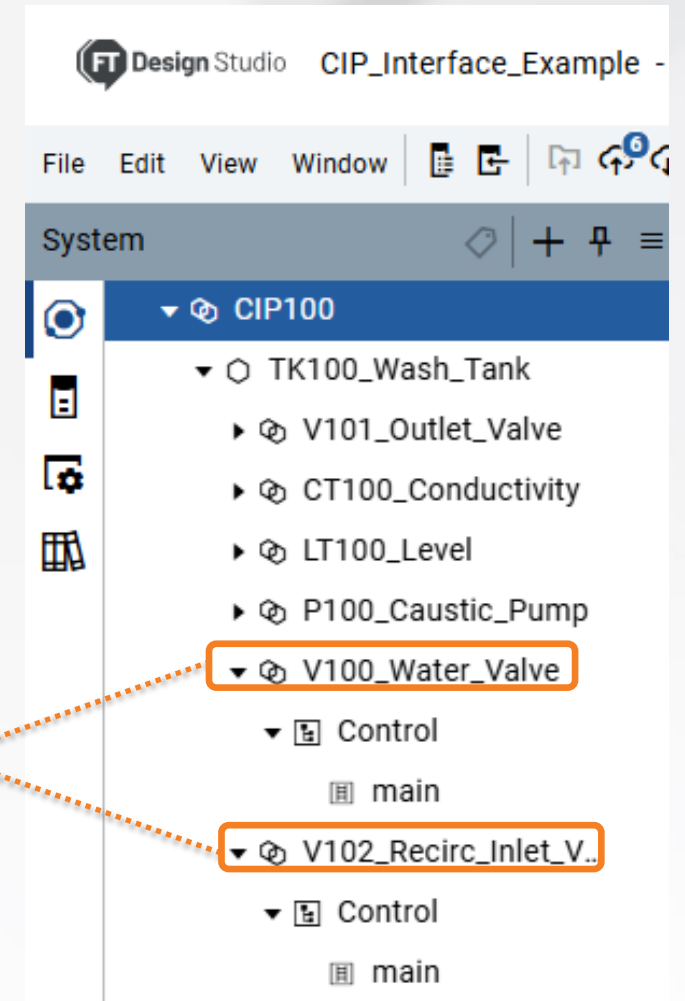
Elements that can reside in the library include AOIs, smart objects, and UDTs.

## BENEFITS

- Develop projects and manage reusable content in the same tool
- Inherit library updates with no import/export



Library explorer



# Sharing data between objects

Tag usage is defined at the program level and can be designated as local or shared

main      zControl

CIP\_PROC\_3\_Tank ▶ zControl

	Name	Data Type	Usage	Value	Style
▶	Acid_Tank_Chemical_Pump	IPUMP_CMDS	OUTPUT		
	Acid_Tank_Chemical_Reac...	BOOL	LOCAL	0	DECIMAL
▶	Acid_Tank_Conductivity	IANALOGIN	INPUT		
	Acid_Tank_Fill_Reached	BOOL	LOCAL	0	DECIMAL
	Acid_Tank_Holding_for_Che...	BOOL	LOCAL	0	DECIMAL
	Acid_Tank_Holding_for_Fill	BOOL	LOCAL	0	DECIMAL
▶	Acid_Tank_Level	IANALOGIN	INPUT		

Connections are made to other smart objects from the properties pane

FT Design Studio CIP\_Interface\_Exam

File Edit View Window

Library

- ▼ Add-On Instructions
  - ▼ raM\_Dvc\_AnalogCha...
    - EnableInFalse
    - Logic
  - ▼ raM\_Dvc\_DigitalCha...
    - EnableInFalse
    - Logic
  - ▶ raM\_Dvc\_Valve\_Fdbk
- ▼ Smart Objects
  - ▼ CIP\_PROC\_3\_Tank
    - ▼ zControl
      - main
      - Devices
      - Sequencer
      - Timer\_Holds
  - ▶ CM\_AI
  - ▶ CM\_AO
  - ▶ CM\_Pump
  - ▶ CM\_Valve
- ▼ Data Types
  - ▶ Add-On Defined
  - ▼ User-defined
    - iAnalogIn
    - iPump\_Cmds

# Sharing data between objects

Tag usage is defined at the program level and can be designated as local or shared

Smart objects pass data to each other using connections

FT Design Studio CIP\_Interface\_Example - Saved

File Edit View Window

System main zControl

CIP100

- TK100\_Wash\_Tank
  - V101\_Outlet\_Valve
  - CT100\_Conductivity
    - Control
      - main
  - LT100\_Level
  - P100\_Caustic\_Pump
  - V100\_Water\_Valve
    - Control
      - main
  - V102\_Recirc\_Inlet\_V...
  - Control
    - main
- Return\_Header
- Supply\_Header
- TK200\_Rinse\_Tank
- TK300\_Acid\_Tank
- zControl

CIP\_PROC\_3\_Tank ▶ zControl

Name	Data Type	Usage
▶ Acid_Tank_Chemical_Pump	IPUMP_CMDS	OUTPUT
Acid_Tank_Chemical_Reac...	BOOL	LOCAL
▶ Acid_Tank_Conductivity	IANALOGIN	INPUT
Acid_Tank_Fill_Reached	BOOL	LOCAL
Acid_Tank_Holding_for_Che...	BOOL	LOCAL
Acid_Tank_Holding_for_Fill	BOOL	LOCAL
▶ Acid_Tank_Level	IANALOGIN	INPUT
▶ Acid_Tank_Outlet_Valve	IVALVE_CMDS	OUTPUT
▶ Acid_Tank_Water_Valve	IVALVE_CMDS	OUTPUT
▶ Air_Blow_Valve_Cmds	IVALVE_CMDS	OUTPUT
CIP_Complete	BOOL	LOCAL
CIP_Hold_Timer	BOOL	LOCAL
CIP_Holding	BOOL	LOCAL
CIP_Paused	BOOL	LOCAL
CIP_Ready	BOOL	LOCAL

Connections are made to other smart objects from the properties pane

General Connections

▼ Within Controller

Area.CIP100.TK300\_Acid\_Tank.CT300\_Conductivity.Control.AI



# Smart object instantiation

**Create FactoryTalk Smart Object instance**

Name  
T103\_New\_Inlet\_Valve\_Test

Alphanumeric characters and underscores only

Create an instance from a definition  
☒ Yes

Select definition

Search

Definition	Description
CM_Pump	2 State Discrete Pump Control
CM_AI	Analog Processing
CM_AO	
CM_Valve	2 State Discrete Valve Control

**System**

- ▼ CIP100
  - ▼ TK100\_Wash\_Tank
    - New FactoryTalk Smart Object Instance
    - New program...
    - Cut (Ctrl+X)
    - Copy (Ctrl+C)
    - Paste (Ctrl+V)
    - Delete
    - Allocation
    - Add to library...
    - Code Editor
    - TK200\_Rinse\_Tank
    - TK300\_Acid\_Tank

**Properties**

Smart Object instance  
T103\_New\_Inlet\_Valve\_Test

Alphanumeric characters and underscores only

Description

Definition  
CM\_Valve

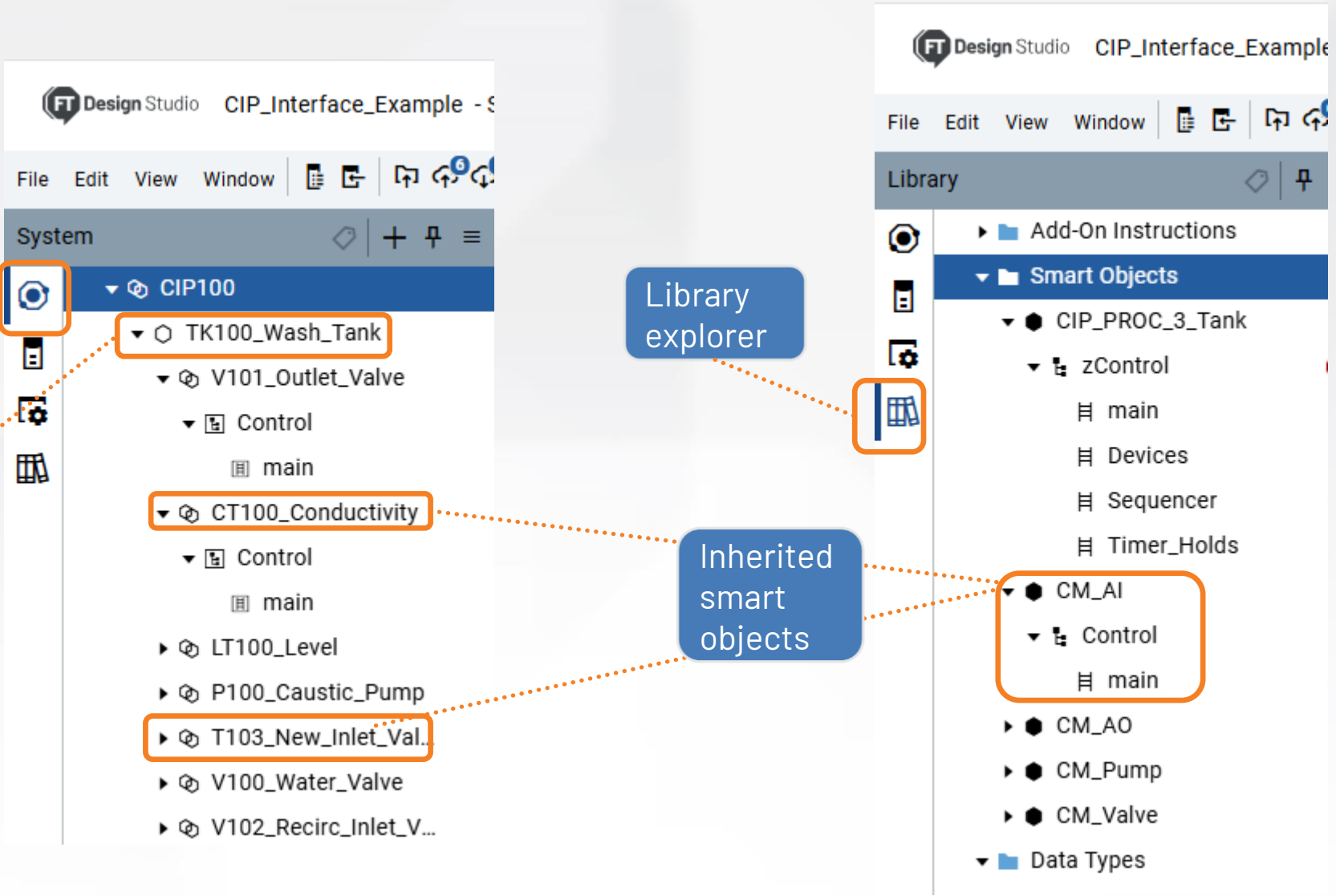
**Instantiate an existing smart object definition**

**The new smart object appears in the system pane with all the inherited content of the existing definition**




**Existing definition**

**Create a new smart object**

# Singleton smart objects vs. smart object definition

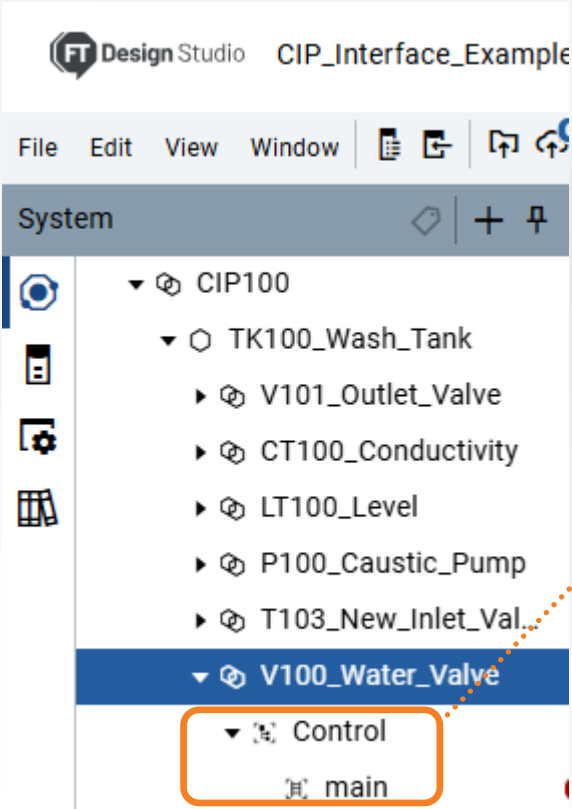
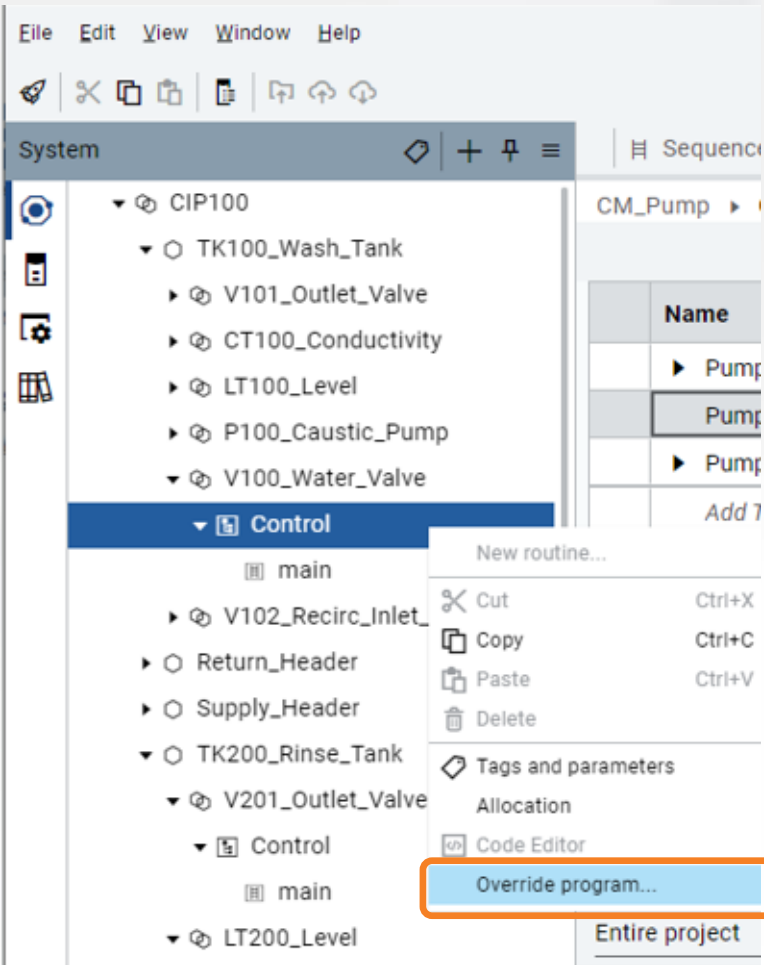


Smart object icon reference

-  Smart object definition
-  Singleton smart object
-  Inherited smart object (i.e. based on a library definition)

# Overriding a smart object definition

Some objects may need to be adjusted from the base definition



Changes to an overridden program do not affect the smart object definition

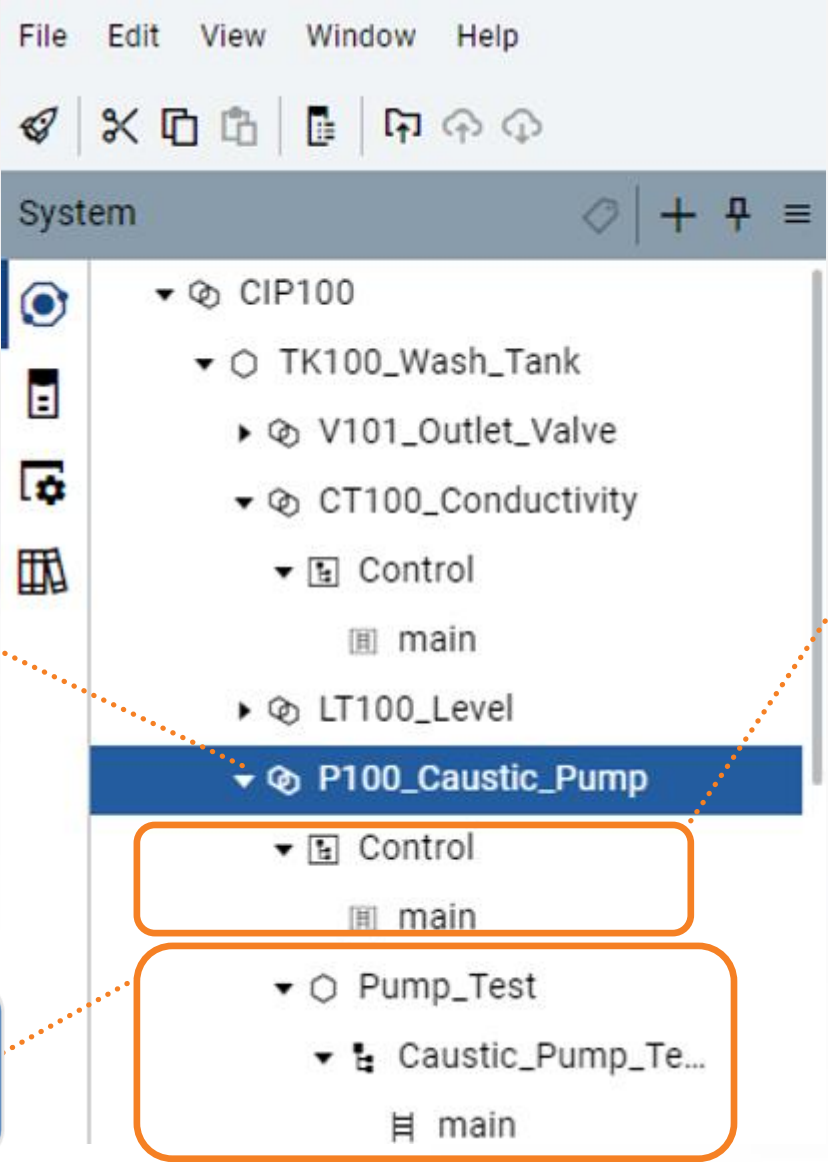


# Extend a library definition

Smart object instances can have additional content added while still inheriting from the base definition

Smart object inherited from library

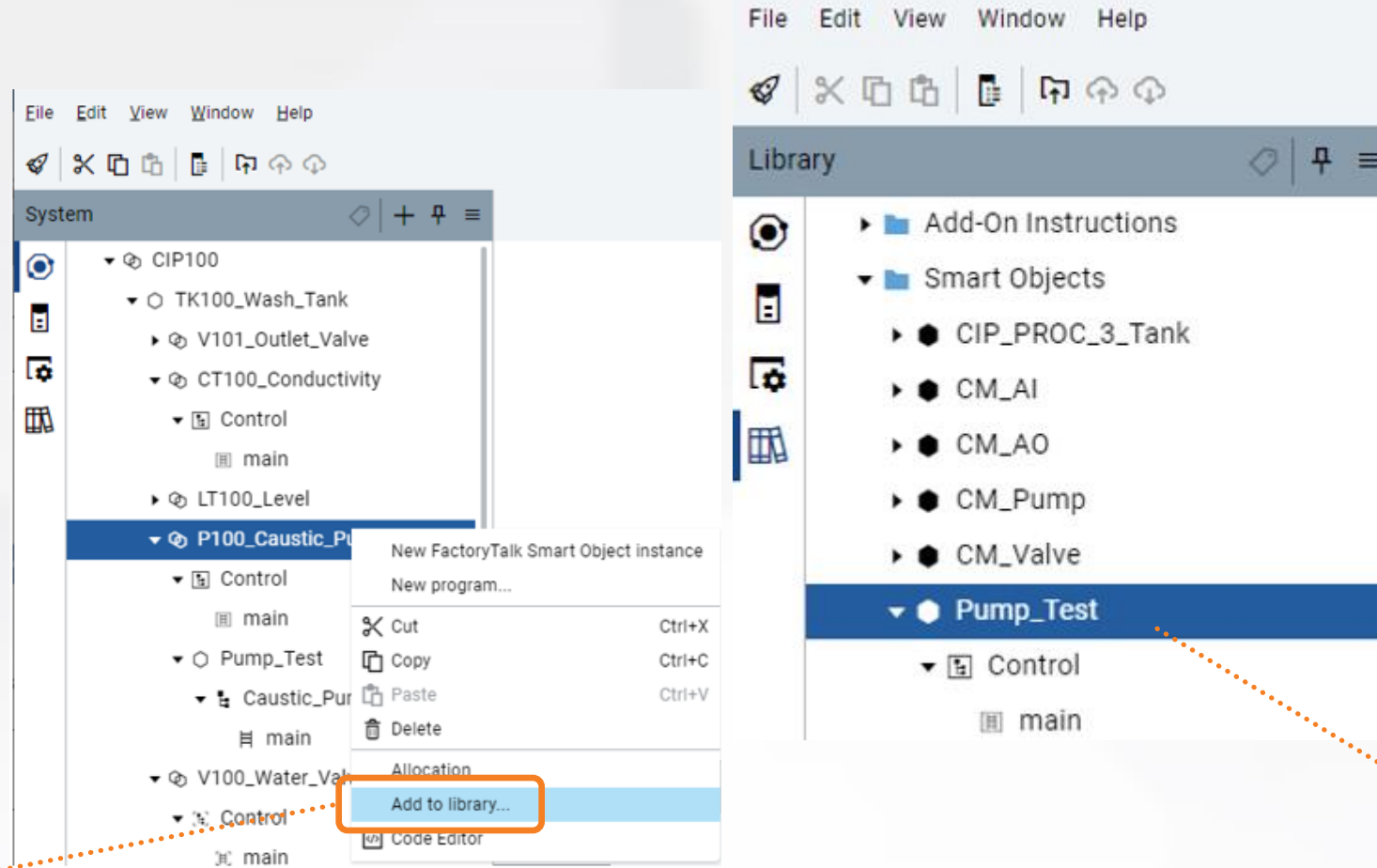
Extension of the P100\_Caustic\_Pump smart object



This portion of the smart object will maintain its inheritance ability from the library object

Content extending an object does not alter the base definition

# Add a new object back to library



Make changes,  
then add back  
to library

New smart  
object  
definition  
created and  
saved back  
to library

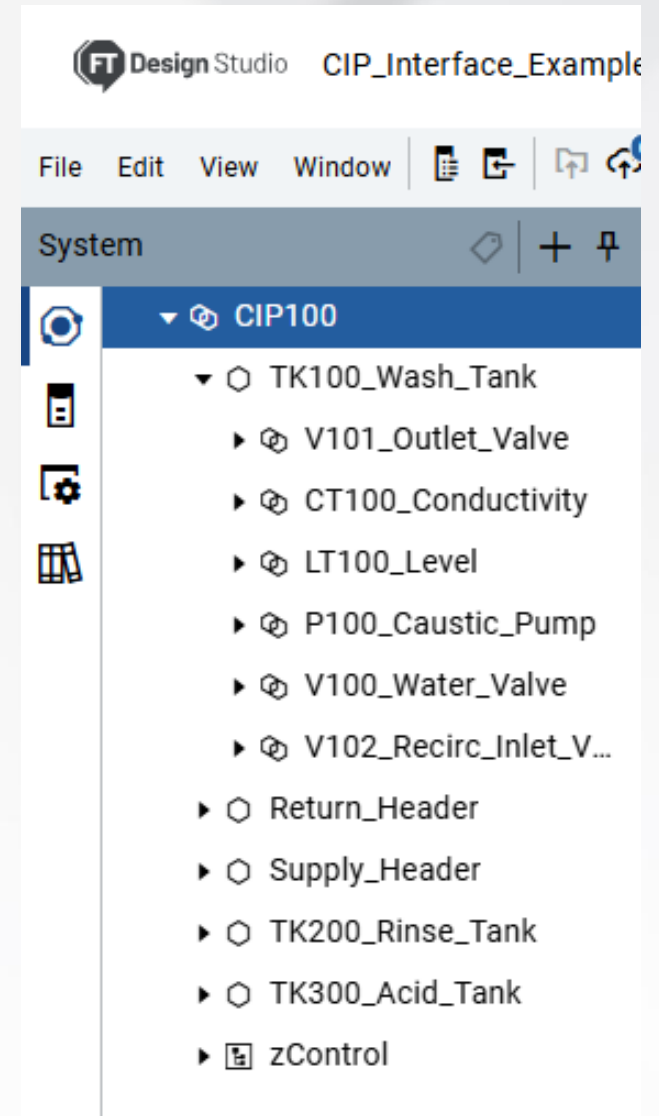
Integrated library management allows  
definition edits in addition to inheritance

# Smart object recap

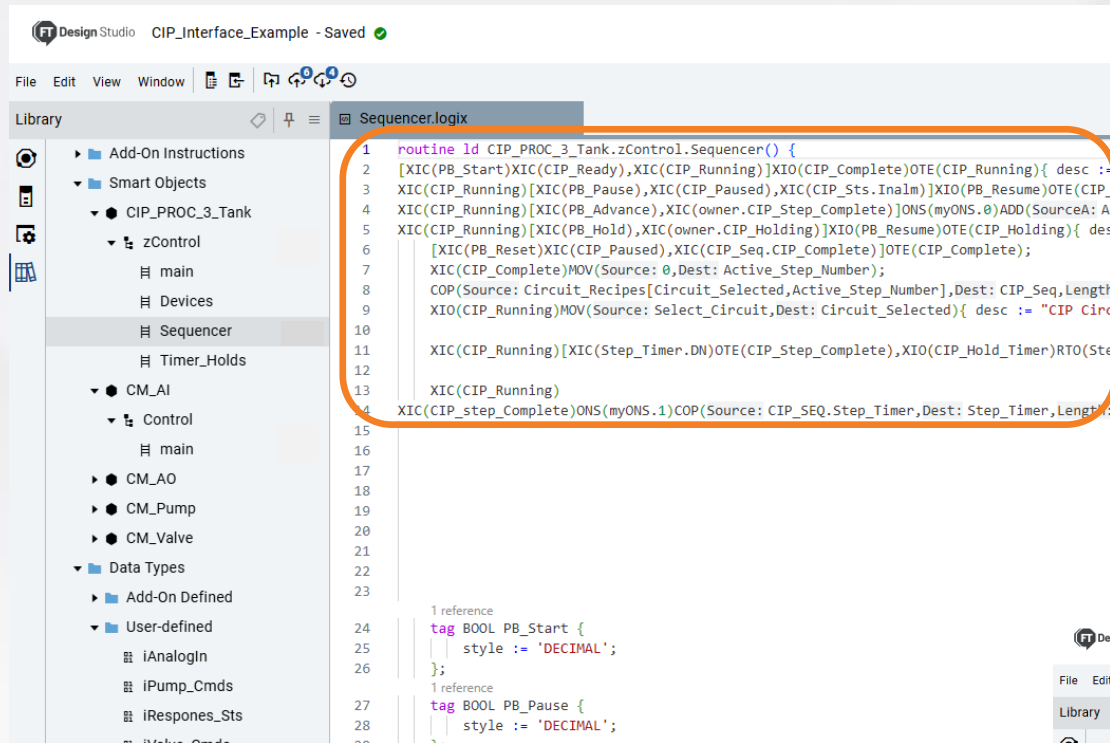
Smart objects are setting the foundation for the modern software experience in FactoryTalk® Design Studio™. They ...

- provide the basis for a modular, scalable design environment.
- expand on the idea of the AOI but can encapsulate other smart objects as well as multiple programs and routines.
- allow you to create a logical hierarchy that's easily understandable.
- can be standalone or inherited from the library.
- can be created in the integrated library so they are easily transported and instantiated.
- can be extended to account for differences in the project design.
- can interact with other smart objects by passing data in and out through a program parameter connection.
- will continue to be expanded on through future iterations of the product.

The integrated library currently supports one project but will be expanded in future updates to be shared among many FactoryTalk® Design Studio™ projects.



# Power-user development



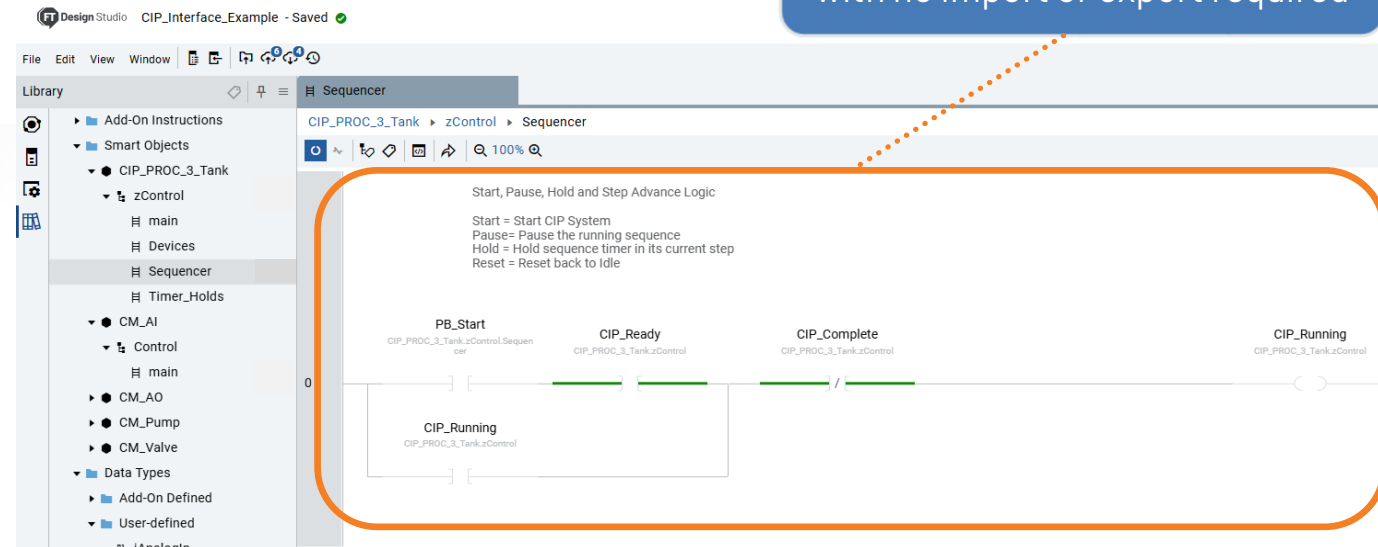
Program in the FactoryTalk®  
Design Studio™ domain-  
specific language (DSL)

Automatically renders graphically  
with no import or export required

Develop faster with text-based  
programming that displays  
graphically

## BENEFITS

- Support for graphical and text-based skill sets in the same environment
- Streamlined mass creation and mass edit workflows



# Integrated version control

Reduce project change management complexity with integrated git technology

Track complete project change history including user identification, time/date, and comments

## BENEFITS

- Log all project updates with change tracking and on-demand detailed comparison
- Easy testing and risk-free updates with checkpoints that can be rolled back

Compare different points in project history

The screenshot displays a software development environment with a file explorer on the left, a 'Change history' table in the center, and a 'Compare commits' window on the right.

**Change history table:**

Description	Update date	Email	ID
added rungs to light seq	08/31/22 - 21:12	jarobinson@ra.rockwell.com	5e53cee
Merge branch master of http...	08/31/22 - 18:45	ftdsdemo01@gmail.com	642f4cc
Added Orlando Controller	08/31/22 - 18:44	ftdsdemo01@gmail.com	221535c
Added NOP to Light Control	08/31/22 - 18:41	jarobinson@ra.rockwell.com	4dd46df
Added tags to Light control	08/31/22 - 18:41	jarobinson@ra.rockwell.com	4dd46df
FTDS: Project upgraded to ver...	08/29/22 - 17:...		
First commit	08/29/22 - 17:...		

**Compare commits window:**

**Commit A**  
Description: Added Orlando Controller  
Date: 08/31/22 - 18:44  
Author: ftdsdemo01@gmail.com

**Commit B**  
Description: added rungs to light seq  
Date: 08/31/22 - 21:12

**Status and Path:**

Status	Path
Updated	model/Milwaukee/Ward_3rd/StPaul_Jefferson/Control/LightControl_Control_DO_Lights.logix
Updated	model/Milwaukee/Ward_3rd/StPaul_Jefferson/Control/LightControl_Control_SEQ_Lights.logix

**Code Comparison:**

```
Commit A: model/Milwaukee/Ward_3rd/StPaul_Jefferson/Control/LightControl_Control_DO_Lights.logix
1 routine Id LightControl.Lights_Control.DO_Lights()
2   Lights(D1,SEQ.D1_Active,SEQ.Normal){ desc := "D
3   Lights(D2,SEQ.D2_Active,SEQ.Normal){ desc := "D
4 }
```

```
Commit B: model/Milwaukee/Ward_3rd/StPaul_Jefferson/Control/LightControl_Control_DO_Lights.logix
1 routine Id LightControl.Lights_Control.DO_Lights()
2   Lights(D1,SEQ.D1_Active,SEQ.Normal){ desc := "D
3   Lights(D2,SEQ.D2_Active,SEQ.Normal){ desc := "D
4+   OTE();
5+   OTE();
6 }
```



# Version control now easier using copilot commands

Use the copilot help you manage changes in your project

- Ask for a list and uncommitted changes
- Ask for a description of the uncommitted changes
- Ask for descriptions to be added
- Ask to commit, push, and pull changes

Explain the uncommitted changes

Here are the current uncommitted changes in your FactoryTalk Design Studio (FTDS) project:

- Files marked with **A** are newly added and not yet committed.
- Files marked with **M** have been modified but not yet committed.

File Path	Status	Description
model/CIP100/SO_new/Control/Control.logix	A	Newly added file
model/CIP100/SO_new/Control/fault.logix	A	Newly added file
model/CIP100/SO_new/Control/main.logix	A	Newly added file
applications/CLX_CIP100/CLX_CIP100.logix	M	Modified file
applications/CM_b/CM_b.logix	M	Modified file
model/Area_CIP100_TK100_Wash_Tank_P100_Caustic_Pump_01_Control.logix	M	Modified file
model/Area_CIP100_TK100_Wash_Tank_xP100_Caustic_Pump_01_Control.logix	M	Modified file
model/area.model	M	Modified file

**What this means:**  
These files have changes that are not yet committed to your version control system. You can review, commit, or discard these changes as needed.

Would you like to see the details of these changes, commit them, or perform another version control operation?

Ask Copilot

**Copilot**

In FactoryTalk Design Studio (FTDS) version control, the status codes are:

- M** = Modified (the file has changes that are not yet committed)
- A** = Added (the file is new and has not been committed)
- D** = Deleted (the file has been removed but the change is not yet committed)
- R** = Renamed (the file has been renamed)
- U** = Untracked (the file is new and not yet added to version control)

So, in your case, model/Area\_CIP100\_TK100\_Wash\_Tank\_P100\_Caustic\_Pump\_01\_Control.logix has been changed but those changes have not been committed yet.

can you please describe the change in the uncommitted changes?

There are currently no merge conflicts or detailed diff information available for the uncommitted changes. The version control system only indicates that the file model/Area\_CIP100\_TK100\_Wash\_Tank\_P100\_Caustic\_Pump\_01\_Control.logix has been modified, but it does not provide a line-by-line summary of what was changed.

If you need to see the specific changes, you may need to review the file directly in FactoryTalk Design Studio or use the built-in comparison tools to view the differences before committing. Let me know if you want to proceed with staging, committing, or discarding these changes!

yes, I'd like to proceed with staging, committing these changes.

Please provide a commit message to describe the changes you are committing. This message will help you and others understand the purpose of the changes in the future.

Once you provide the commit message, I will proceed with staging and committing the changes.

**DesignStudio** myCIP\_Start\_RSS - Saved

File Edit View Window

System CIP100 SO\_ne... TK100... zCont...

Change history

Description	Update date	Email	ID
Added new control logic files ...	08/11/25 - 09:48	ewelina.adamczak@rockwellaut...	d770db8
FTDS: Added new control logic files for SO_new (Control, fault, main); modified controller applications	08/05/25 - 11:23	enader@rockwellautomation.com	b04a85f
deplc (CLX_CIP100_CM_b); updated wash tank caustic pump control logic and area model.	04/25/25 - 09:35	enader@rockwellautomation.com	20fd029
deplc (CLX_CIP100_CM_b); updated wash tank caustic pump control logic and area model.	04/25/25 - 08:30	enader@rockwellautomation.com	3896b38
deplc (CLX_CIP100_CM_b); updated wash tank caustic pump control logic and area model.	04/25/25 - 08:18	enader@rockwellautomation.com	e1fa3c5
deplc (CLX_CIP100_CM_b); updated wash tank caustic pump control logic and area model.	04/25/25 - 08:16	enader@rockwellautomation.com	c8d4003
deplc (CLX_CIP100_CM_b); updated wash tank caustic pump control logic and area model.	04/25/25 - 08:12	enader@rockwellautomation.com	9b3a047
deploy - 2025-04-25T12:11:20.91...	04/25/25 - 08:11	enader@rockwellautomation.com	3e475f6
deploy - 2025-04-21T18:36:35.74...	04/21/25 - 14:36	enader@rockwellautomation.com	8e3ffda
deploy - 2025-04-17T17:05:57.10...	04/17/25 - 13:05	enader@rockwellautomation.com	dea26e2
deploy - 2025-04-17T15:48:31.97...	04/17/25 - 11:48	enader@rockwellautomation.com	4d49ed7
deploy - 2025-04-17T15:46:04.98...	04/17/25 - 11:46	enader@rockwellautomation.com	54349bc
deploy - 2025-04-17T14:56:01.46...	04/17/25 - 10:56	enader@rockwellautomation.com	32750e4
deploy - 2025-04-16T15:21:55.09...	04/16/25 - 11:21	enader@rockwellautomation.com	1cd80e9
deploy - 2025-04-16T14:39:26.62...	04/16/25 - 10:39	enader@rockwellautomation.com	704cf3e
Project copied from myCIP_Start	04/16/25 - 09:33	enader@rockwellautomation.com	124b53f
FTDS: The project has been upgr...	04/16/25 - 09:32	enader@rockwellautomation.com	654177b
FTDS: The project has been upgr...	10/11/24 - 10:12	ajgregory@ra.rockwell.com	ab3cf9a
Fixed strings and NULLs	09/20/24 - 12:29	ajgregory@ra.rockwell.com	fd7bdsb
Added project to the default Vaul...	09/20/24 - 11:56	alemkelde@ra.rockwell.com	6a2b612
FTDS: The project has been upgr...	09/20/24 - 11:56	alemkelde@ra.rockwell.com	3f4da76
deployment of sample projects	09/20/24 - 11:52	cdsteinf@ra.rockwell.com	5032b55

**Copilot**

model/CIP100/SO\_new/Control/main.logix  
applications/CLX\_CIP100/CLX\_CIP100.logix  
applications/CM\_b/CM\_b.logix  
model/Area\_CIP100\_TK100\_Wash\_Tank\_P100\_Caustic\_Pump\_01\_Control.logix  
model/Area\_CIP100\_TK100\_Wash\_Tank\_xP100\_Caustic\_Pump\_01\_Control.logix  
model/area.model

**What this means:**  
These files have changes that are not yet committed to your version control system. You can review, commit, or discard these changes as needed.

Would you like to see the details of these changes, commit them, or perform another version control operation?

commit changes and create a message that summarizes made

All uncommitted changes have been committed with the following summary message:

- Added new control logic files for SO\_new (Control, fault, main)
- Modified controller applications (CLX\_CIP100\_CM\_b)
- Updated wash tank caustic pump control logic and area model

Your project is now up to date in version control. If you need to push these changes to a remote repository or perform another operation, let me know!

push these changes

All committed changes have been successfully pushed to the remote repository.

Your project is now synchronized with the remote version control system. If you need to pull updates, check status, or perform any other version control operation, just let me know!

Ask Copilot



# Project history

FT Design Studio CIP\_Interface\_Example - Saved

File Edit View Window

System

CIP100

TK100\_Wash\_Tank

V101\_Outlet\_Valve

CT100\_Conductivity

LT100\_Level

P100\_Caustic\_Pump

Control

main

main

main

Change history

Description	Update date	Email
Uncommitted changes		
FTDS: The project has been upgraded.	01/28/25 - 13:44	pyneimes
FTDS: The project has been upgraded.	01/22/25	
Added project to the default Vault soluti...	07/16/24	
FTDS: The project has been upgraded.	07/16/24	

View information on where changes have been made, when it was made, who it was made by, and what was changed

View uncommitted changes and discard if you don't want them

When remote changes are available, a cloud icon will appear next to the description in the change history table

Change history	
Description	Update date
Uncommitted changes	
program created at the Ch...	05/15/23 - 20:30

Uncommitted changes

Status	Path
Added	libraries/project/smart objects/Pump_Test/Pump_Test/Caustic_Pump_Test/main.logix
Added	model/CIP100/TK100_Wash_Tank/V100_Water_Valve/Control/Control.logix
Added	model/CIP100/TK100_Wash_Tank/V100_Water_Valve/Control/main.logix
Updated	applications/CIP100_CLX/CIP100_CLX.logix
Updated	model/area.model

Last committed version  
applications/CIP100\_CLX/CIP100\_CLX.logix

Uncommitted changes  
applications/CIP100\_CLX/CIP100\_CLX.logix

35 }  
36  
37 controller LogixApp\_CIP100\_CLX\_Definition;  
38 LogixApp\_CIP100\_CLX\_Definition LogixApp\_CIP100\_CLX\_  
39  
40 deployment LogixApp\_CIP100\_CLX\_Deployment {  
41 LogixApp\_CIP100\_CLX -> LogixApp\_CIP100\_CLX\_I  
42 topology -> production;  
43 }  
44

24 }  
25  
26 controller LogixApp\_CIP100\_CLX\_Definition;  
27 LogixApp\_CIP100\_CLX\_Definition LogixApp\_CIP100\_CLX\_  
28  
29 deployment LogixApp\_CIP100\_CLX\_Deployment {  
30 LogixApp\_CIP100\_CLX -> LogixApp\_CIP100\_CLX\_Inst  
31 topology -> production;  
32 }  
33

Help Close Discard uncommitted changes

# Detailed change information

Open any point in the project history for information on each edit or compare two points in history

Select a file in the table to compare changes from the previous version to the current

See details of what was modified

The screenshot shows the Design Studio interface with a 'Compare commits' dialog open. The dialog has a header section with 'Description', 'Date', and 'Author' information. Below this is a table of changes. A blue box highlights the first row of the table, which shows an 'Updated' status for the file 'applications/BottlingLine\_Ctrl/BottlingLine\_Ctrl.logix'. Below the table is a side-by-side comparison of the code for the selected file, showing the 'Previous version' and the 'Current version'. The code is highlighted with a blue box, showing changes in the task definition and watchdog settings.

**Compare commits**

Description: Updated description of FillStation program  
Date: 01/20/23 - 15:56  
Author: teat@ra.rockwell.com

Status	Path
Updated	applications/BottlingLine_Ctrl/BottlingLine_Ctrl.logix
Added	model/BottlingLine/CapStation/CapStation.logix
Added	model/BottlingLine/CapStation/main.logix
Updated	model/area.model

**Previous version**  
applications/BottlingLine\_Ctrl/BottlingLine\_Ctrl.logix

```
1 controllerApp LogixApp_BottlingLine_Ctrl() {  
2- task CONTINUOUS DefaultTask {  
3 desc := "";  
4- watchdog := 500.0;  
5  
6 inhibited := false;  
7 disableAutomaticOutput := false;  
Area.BottlingLine.FillStation;  
8  
9  
10  
11
```

**Current version**  
applications/BottlingLine\_Ctrl/BottlingLine\_Ctrl.logix

```
1 controllerApp LogixApp_BottlingLine_Ctrl() {  
2+ task EVENT DefaultTask {  
3 desc := "";  
4+ period := 250.0;  
5+ priority := 10;  
6+ watchdog := 300.0;  
7  
8 inhibited := false;  
9 disableAutomaticOutput := false;  
10 Area.BottlingLine.FillStation;  
11 executeTaskNoEvent := false;  
12 trigger := EVENT_INSTRUCTION_ONLY;  
13
```

Local changes available

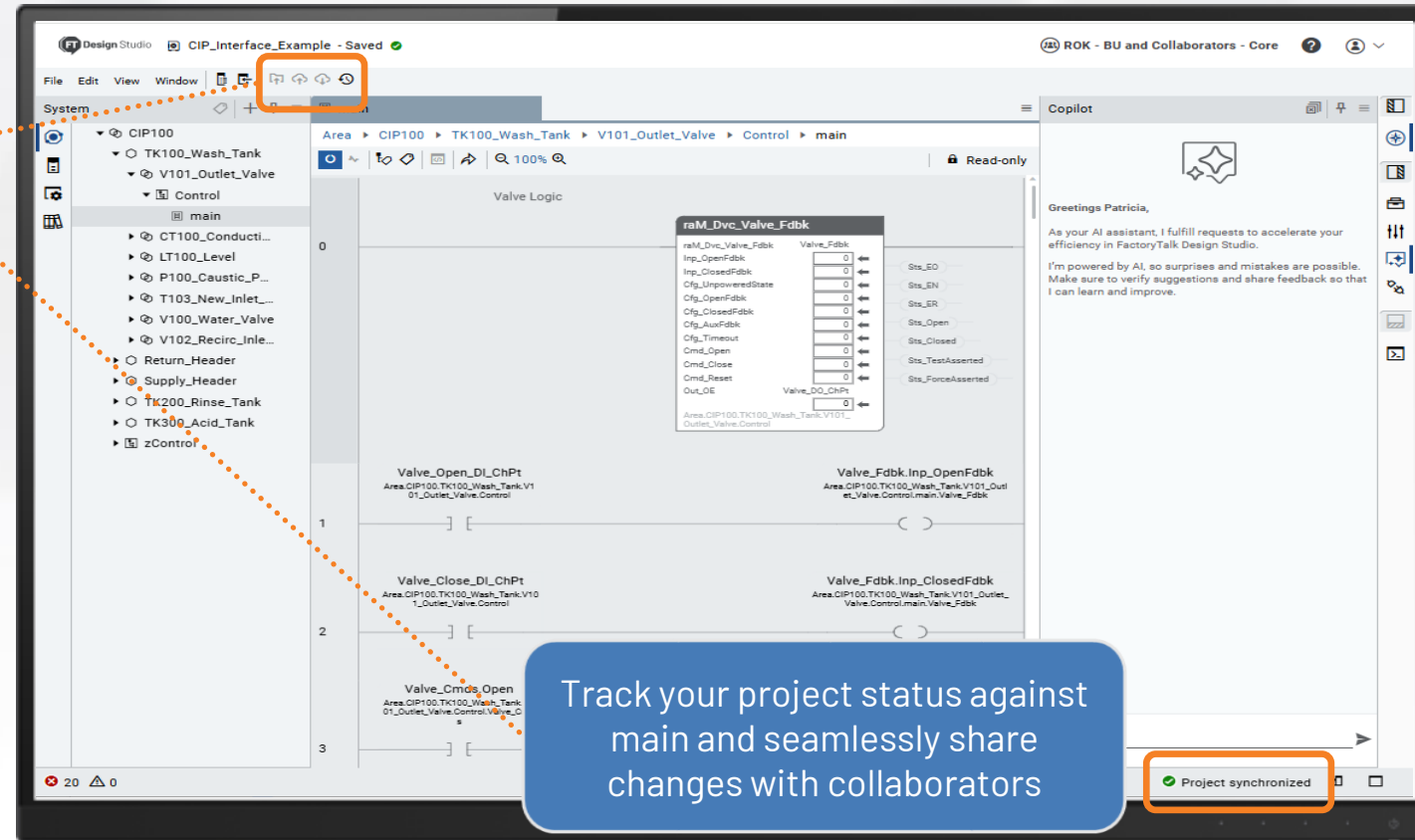
# Multi-user collaboration

Automatically merge changes across multiple users working independently

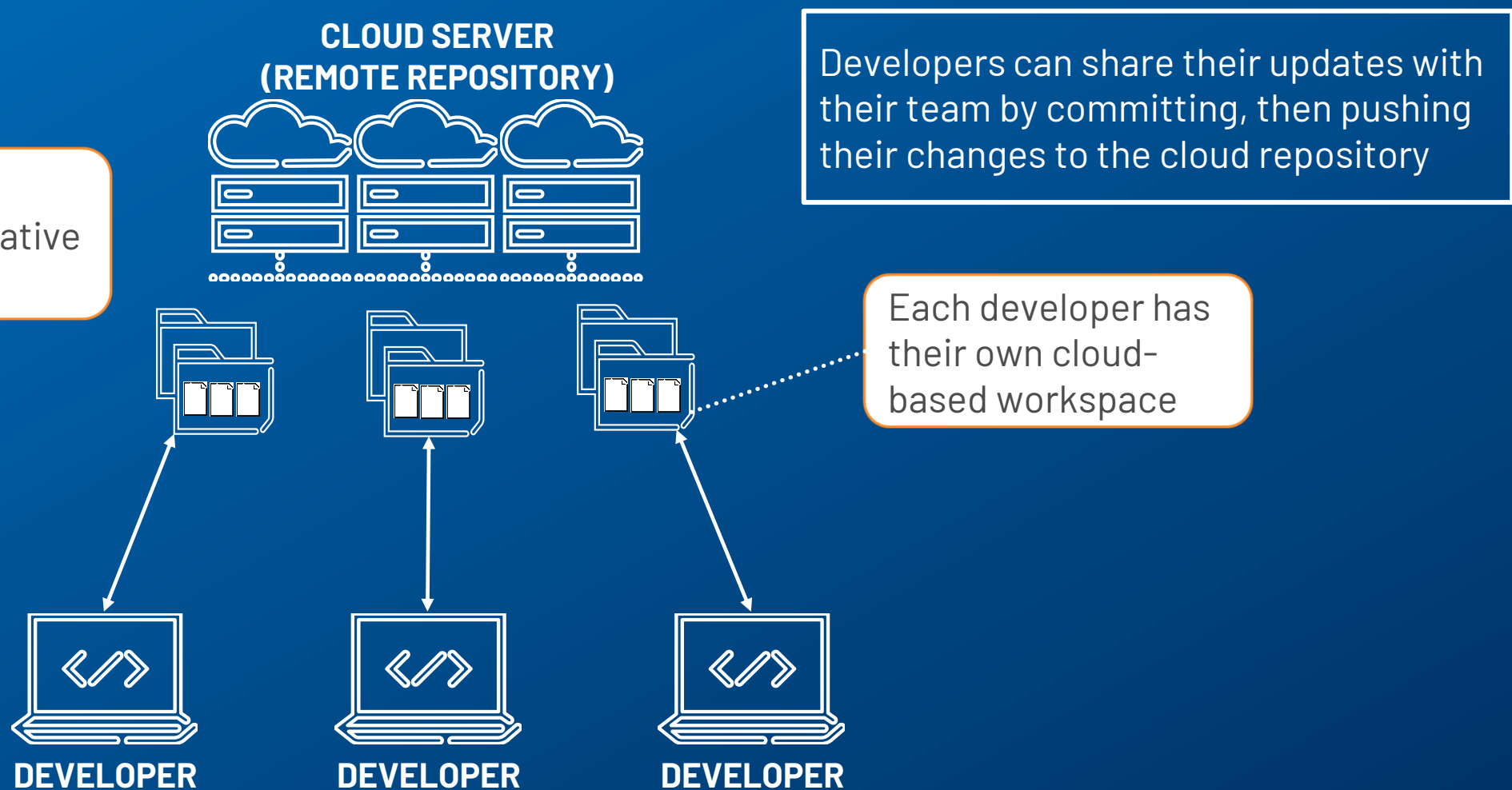
Share project updates with collaborators using built-in tools

## BENEFITS

- Maximize collaboration efficiency with simultaneous project access
- Streamline parallel development by detecting and merging changes from collaborators



# High-level diagram of Version Control Systems (VCS)

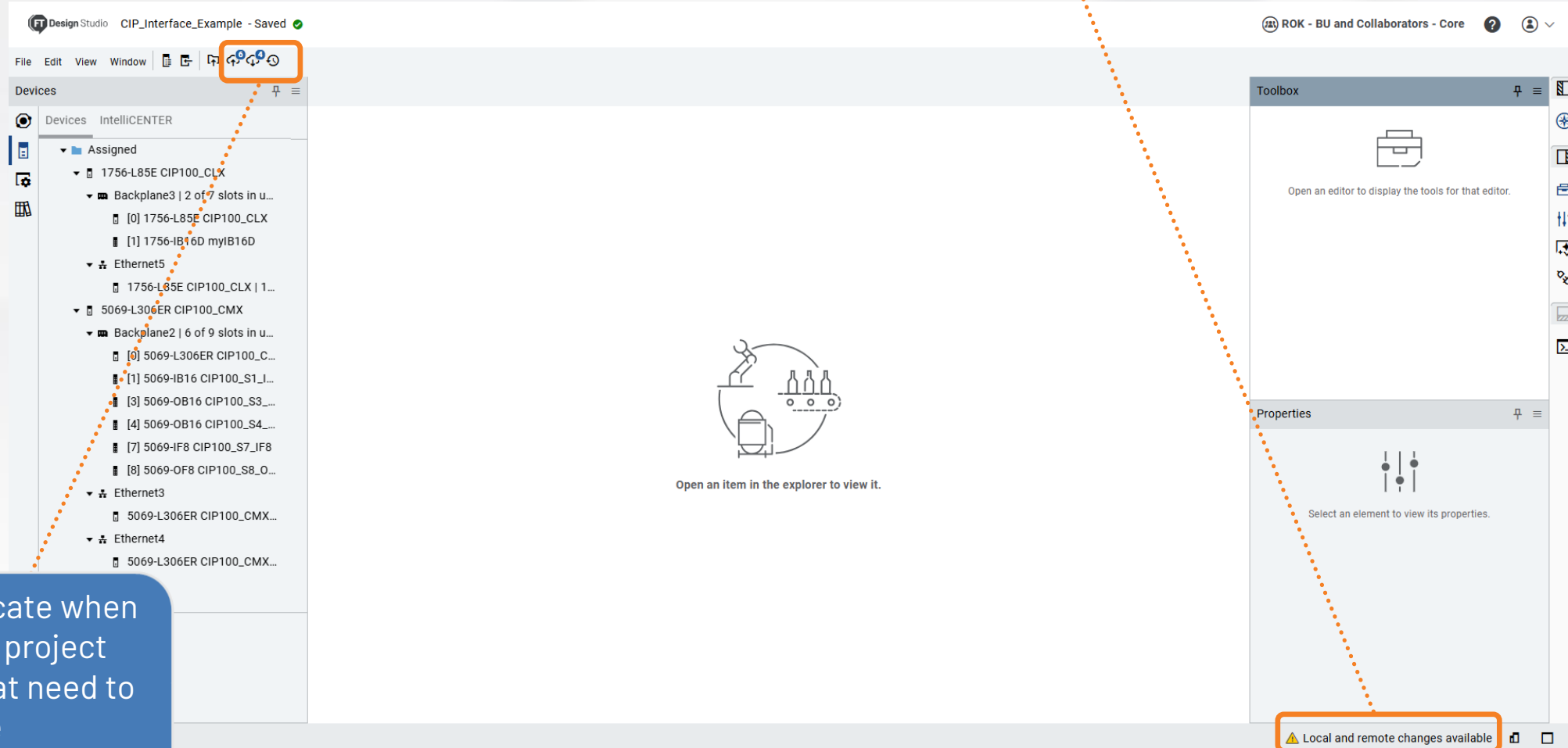


# Project status

Not only do we provide cues to what's going on in your project, but we also prompt you to take action to synchronize your environments

VCS badge updates indicate when you have changes to the project history, and commits that need to be pushed to the remote repository

Awareness of the latest changes to your system



## Pulling/resolving merge conflicts

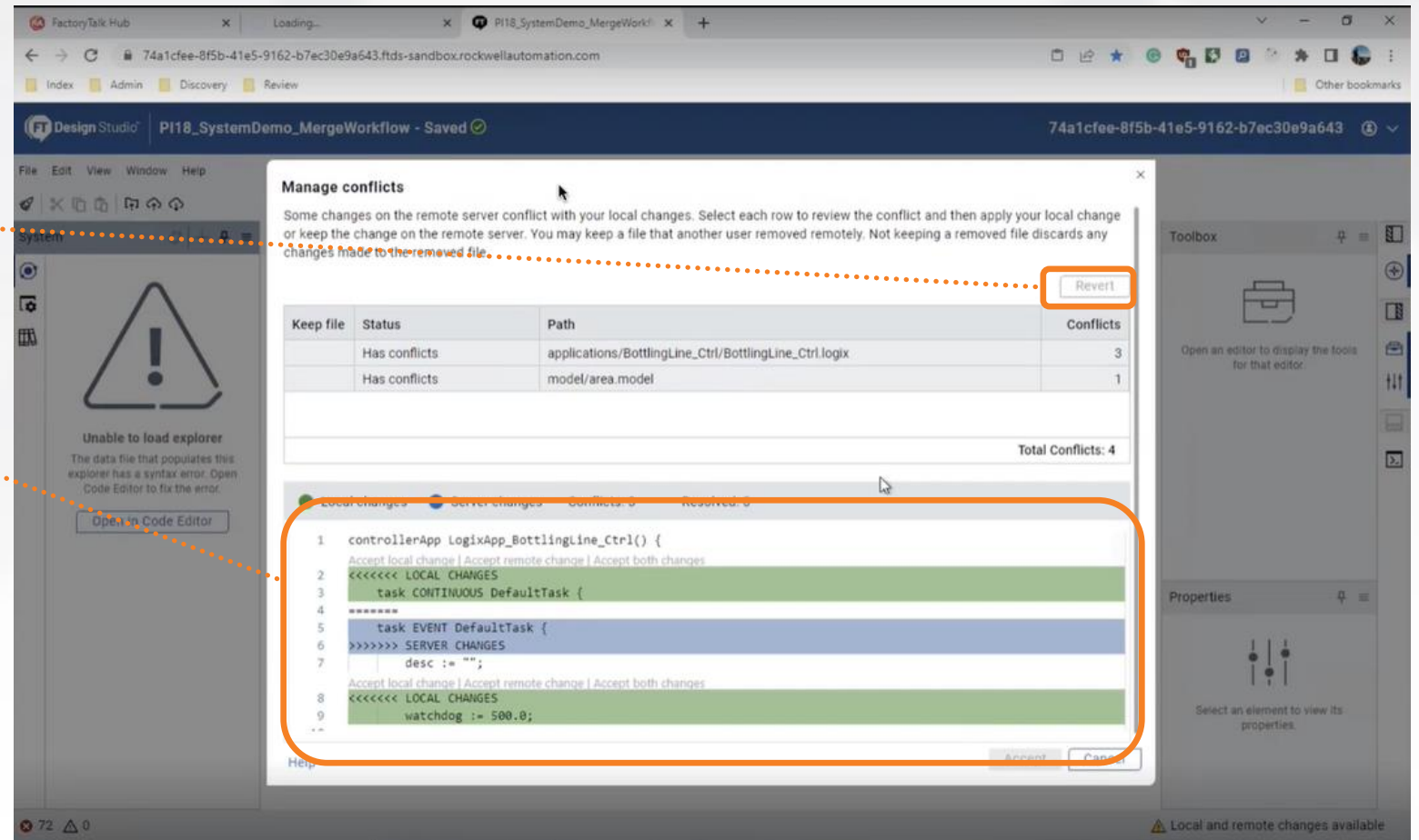
Revert previous file to its conflict state so you can make a different resolution choice

Local changes shown in green;  
server changes in blue

FOR MORE INFO

Visit these KnowledgeBase articles for more details on merging conflicts:

1. How to manage merges with devices?  
[https://rockwellautomation.custhelp.com/app/answers/answer\\_view/a\\_id/1139599](https://rockwellautomation.custhelp.com/app/answers/answer_view/a_id/1139599)
2. How to resolve error you get when merging both changes?  
[https://rockwellautomation.custhelp.com/app/answers/answer\\_view/a\\_id/1139656](https://rockwellautomation.custhelp.com/app/answers/answer_view/a_id/1139656)



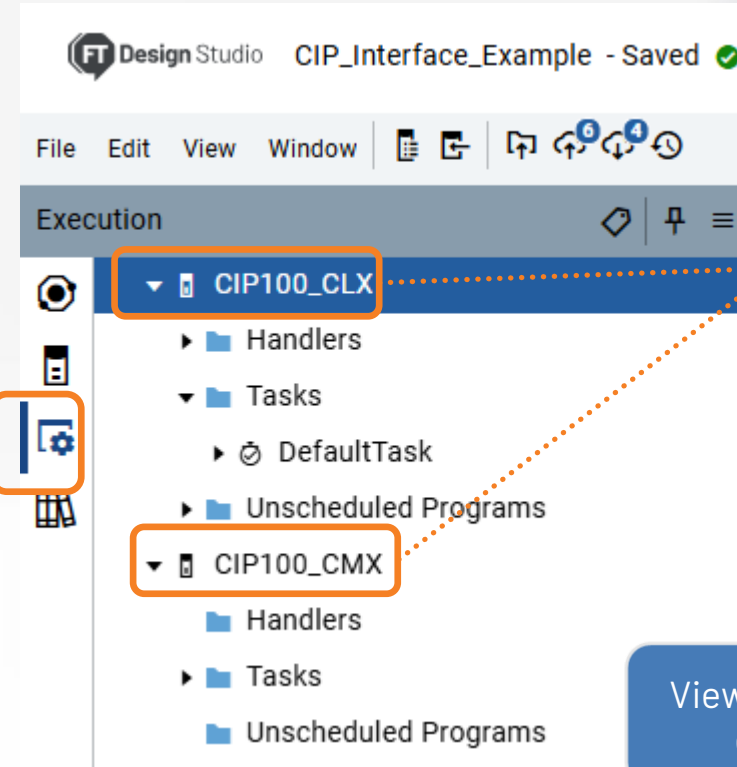


# Multiple controllers, single project

Execution Explorer: Streamline development by managing all controllers in one development space

## BENEFITS

- Scale projects easier than ever with the ability to add and manage multiple controllers in a single project
- Configure and troubleshoot content based on a familiar execution view



# Devices explorer

View the system topology through the Devices Explorer

See all controllers and their related devices in one place

The screenshot displays the Rockwell Automation Design Studio interface. The top bar shows the project name "CIP\_Interface\_Example - Saved" and the user "ROK - BU and Collaborators - Core". The main window is divided into several panes:

- Devices Explorer (Left):** A tree view showing the system topology. It includes a "Devices" icon (highlighted with an orange box) and a list of devices under "Assigned". The list includes:
  - 1756-L85E CIP100\_CLX
    - Backplane3 | 2 of 7 slots in u...
      - [0] 1756-L85E CIP100\_CLX
      - [1] 1756-IB16D myIB16D
    - Ethernet5
      - 1756-L85E CIP100\_CLX | 1...
  - 5069-L306ER CIP100\_CMX
    - Backplane2 | 6 of 9 slots in u...
      - [0] 5069-L306ER CIP100\_C...
      - [1] 5069-IB16 CIP100\_S1\_...
      - [3] 5069-OB16 CIP100\_S3\_...
      - [4] 5069-OB16 CIP100\_S4\_...
      - [7] 5069-IF8 CIP100\_S7\_IF8
      - [8] 5069-OF8 CIP100\_S8\_O...
    - Ethernet3
      - 5069-L306ER CIP100\_CMX...
    - Ethernet4
      - 5069-L306ER CIP100\_CMX...
  - Unassigned

- Device list / Deploy center (Bottom Left):** A small pane for managing device lists.
- Configuration Pane (Center):** Displays the configuration for the selected device, "1756-L85E: CIP100\_CLX". It shows the parent as "<none>", slot as "0", and IP address as "1.1.1.1". The controller connection is "Offline" and "Not Connected". The pane includes tabs for "INFORMATION", "Overview", "CONFIGURATION", "DIAGNOSTICS", and "Faults". The "Overview" tab is active, showing the "1756-L85E ControlLogix® 5580 Controller" with a 3D model. It also displays the "Definition" section with revision "34.001", electronic keying "Disabled Keying", chassis size "7", and slot "0". A "Device definition..." button is present.
- Toolbox (Right):** A pane for selecting components to add to the system.
- Properties (Bottom Right):** A pane for viewing the properties of the selected element.

The bottom status bar shows "20" errors and "0" warnings, and a message "Local and remote changes available".

# Catalog dialogue

Choose from a core set of devices to add to your system

Choose from controllers, analog I/O, discrete I/O, switches, and Ethernet adaptors, and more

**Add devices to project**

Show devices from: ☒ Catalog ☐ Project

Category filter:

- ☐ Analog
- ☒ Communication
- ☐ Controller
- ☐ Digital
- ☐ Input
- ☐ Miscellaneous
- ☐ Output
- ☐ Specialty

Catalog Number	Category	Description
<input type="checkbox"/> 1756-EN2F	Communication	1756 10/100 Mbps Ethernet Bridge, Fiber Media
<input type="checkbox"/> 1756-EN2T	Communication	1756 10/100 Mbps Ethernet Bridge, Twisted-Pair M...
<input type="checkbox"/> 1756-EN2TP	Communication	1756 10/100 Mbps Ethernet Bridge, 2-Port, Twisted...
<input type="checkbox"/> 1756-EN2TR	Communication	1756 10/100 Mbps Ethernet Bridge, 2-Port, Twisted...
<input type="checkbox"/> 1756-EN3TR	Communication	1756 10/100 Mbps Ethernet Bridge, 2-Port, Twisted...
<input type="checkbox"/> 1756-EN4TR	Communication	1756 Ethernet Bridge, 2-Port, Twisted-pair Media
<input type="checkbox"/> 1756-IA16	Digital, Input	16 Point 79V-132V AC Input
<input type="checkbox"/> 1756-IA16I	Digital, Input	16 Point 79V-132V AC Isolated Input
<input type="checkbox"/> 1756-IA32	Digital, Input	32 Point 74V-132V AC Input
<input type="checkbox"/> 1756-IA8D	Digital, Input	8 Point 79V-132V AC Diagnostic Input
<input type="checkbox"/> 1756-IB16	Digital, Input	16 Point 10V-31.2V DC Input
<input type="checkbox"/> 1756-IB16D	Digital, Input	16 Point 10V-30V DC Diagnostic Input

Select a device to configure

Continue Cancel

**Add devices to project**

Show devices from: ☒ Catalog ☐ Project

Category filter:

- ☐ Analog
- ☐ Communication
- ☐ Controller
- ☐ Digital
- ☒ Encoder
- ☐ Input
- ☐ Miscellaneous

Catalog Number	Category	Description
<input type="checkbox"/> 843E-MIPxxBAX	Encoder	Multi-turn Absolute Encoder
<input type="checkbox"/> 843E-SIPxxBAX	Encoder	Single-turn Absolute Encoder

**Add devices to project**

Show devices from: ☒ Catalog ☐ Project

Category filter:

- ☐ Analog
- ☒ Communication
- ☐ Controller
- ☐ Digital
- ☐ Encoder
- ☐ Input
- ☐ Miscellaneous

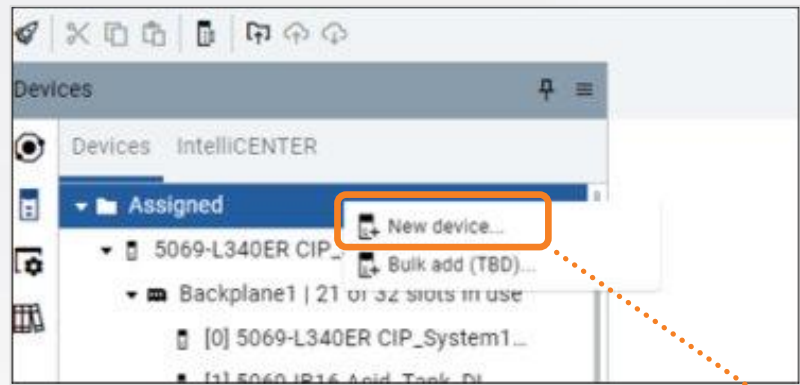
Search: 5800

Catalog Number	Category	Description
<input type="checkbox"/> 1783-MMS10	Communication	Stratix 5800 10 Port Expandable Base
<input type="checkbox"/> 1783-MMS10B	Communication	Stratix 5800 10 Port Non-Expandable Base
<input type="checkbox"/> 1783-MMS10BE	Communication	Stratix 5800 10 Port Non-Expandable Base PoE
<input type="checkbox"/> 1783-MMS10E	Communication	Stratix 5800 10 Port Expandable Base PoE
<input type="checkbox"/> 1783-MMS10EA	Communication	Stratix 5800 10 Port Expandable Base PoE Advanced
<input type="checkbox"/> 1783-MMS10EAR	Communication	Stratix 5800 10 Port Expandable Base PoE Advance...
<input type="checkbox"/> 1783-MMS10ER	Communication	Stratix 5800 10 Port Expandable Base PoE Layer 3
<input type="checkbox"/> 1783-MMS10R	Communication	Stratix 5800 10 Port Expandable Base Layer 3

New in August 2025 (FTDS 2.02 update): 843E encoders and Stratix® 5800 switches

Available devices are a subset of the device catalog, but will continue to be built out with future iterations of FactoryTalk® Design Studio™

# Mass creation workflow for devices



Choose from a core set of available devices

Add quantities according to your system design

Save time adding devices to your projects by adding multiple devices in multiple quantities at the same time

**Add devices to project**

How devices from

☒ Catalog

☐ Project

**Category filter**

- ☐ Analog
- ☐ Communication
- ☐ Controller
- ☐ Digital
- ☐ Input
- ☐ Miscellaneous
- ☐ Output
- ☐ Specialty

Search

	Catalog Numb...	Category	Description
<input type="checkbox"/>	1756-IF16	Analog, Input	16 Channel Non-Isolated ...
<input checked="" type="checkbox"/>	1756-IF16H	Analog, Input	16 Channel HART Analog ...
<input type="checkbox"/>	1756-IF16IH	Analog, Input	16 Channel Isolated HAR...
<input checked="" type="checkbox"/>	1756-IF4FXOF2F	Analog, Input, Out...	4 Current/Volt Inputs/2 C...
<input type="checkbox"/>	1756-IF8	Analog, Input	8 Channel Non-Isolated V...
<input type="checkbox"/>	1756-IF8H	Analog, Input	8 Channel HART Analog I...
<input type="checkbox"/>	1756-IF8I	Analog, Input	8 Channel Voltage/Curren...
<input type="checkbox"/>	1756-IF8IH	Analog, Input	8 Channel Isolated HART ...
<input type="checkbox"/>	1756-IG16	Digital, Input	16 Point 0V-5.5VDC TTL I...
<input type="checkbox"/>	1756-IH16I	Digital, Input	16 Point 90V-146V DC Iso...
<input checked="" type="checkbox"/>	1756-IH16ISOE	Digital, Input	16 Channel Isolated 125V...

**Devices to add**

1756-IF4FXOF2F	3	
1756-IF16H	4	
1756-IH16ISOE	2	

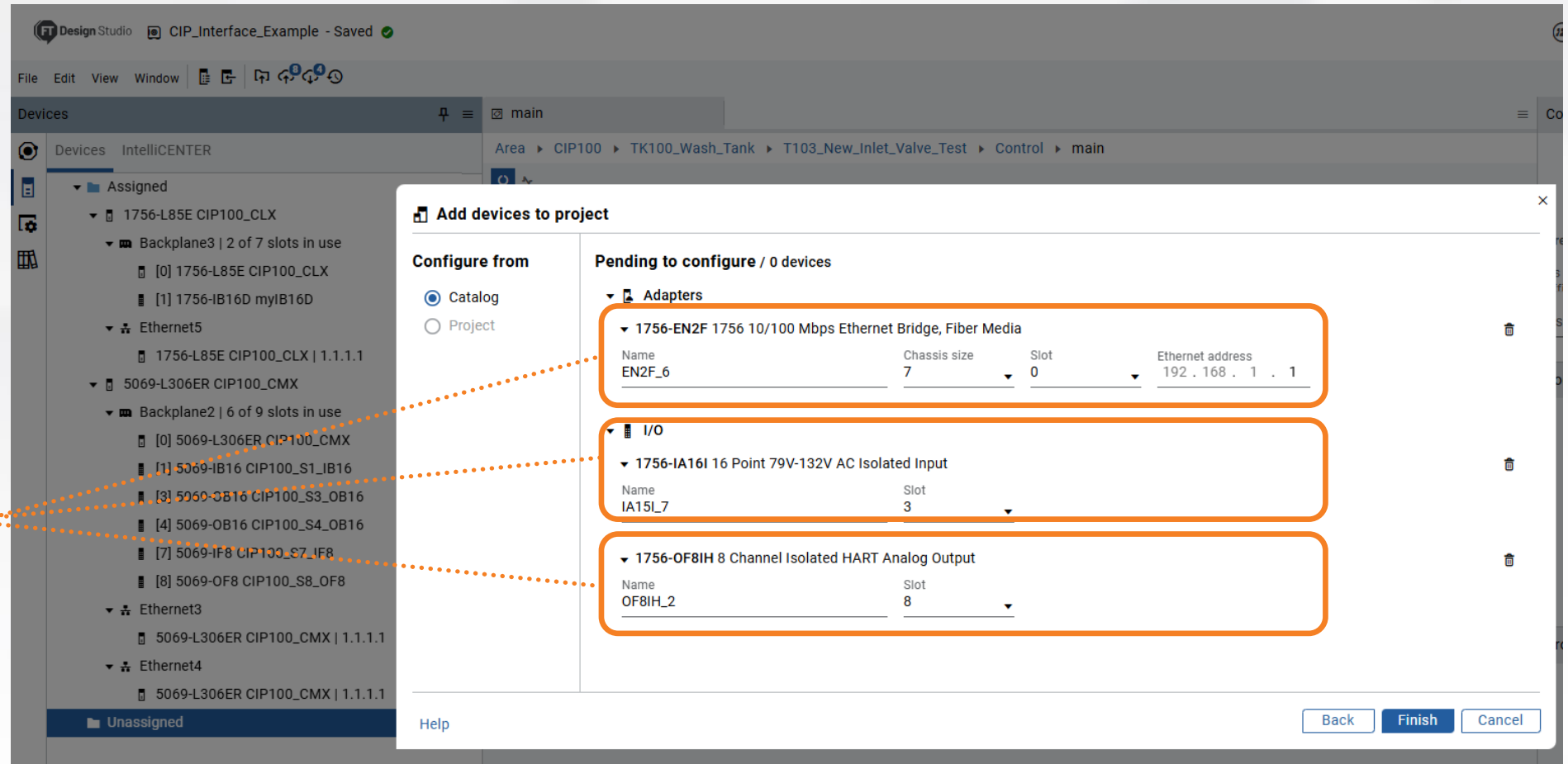
Help

Continue Cancel

# Quick configure workflow for devices

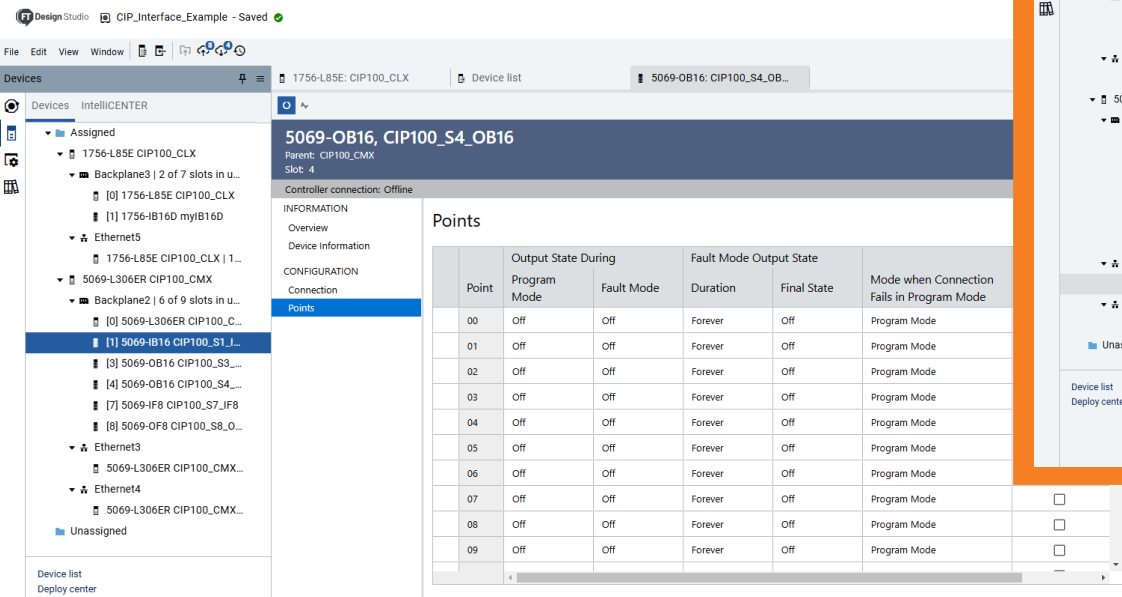
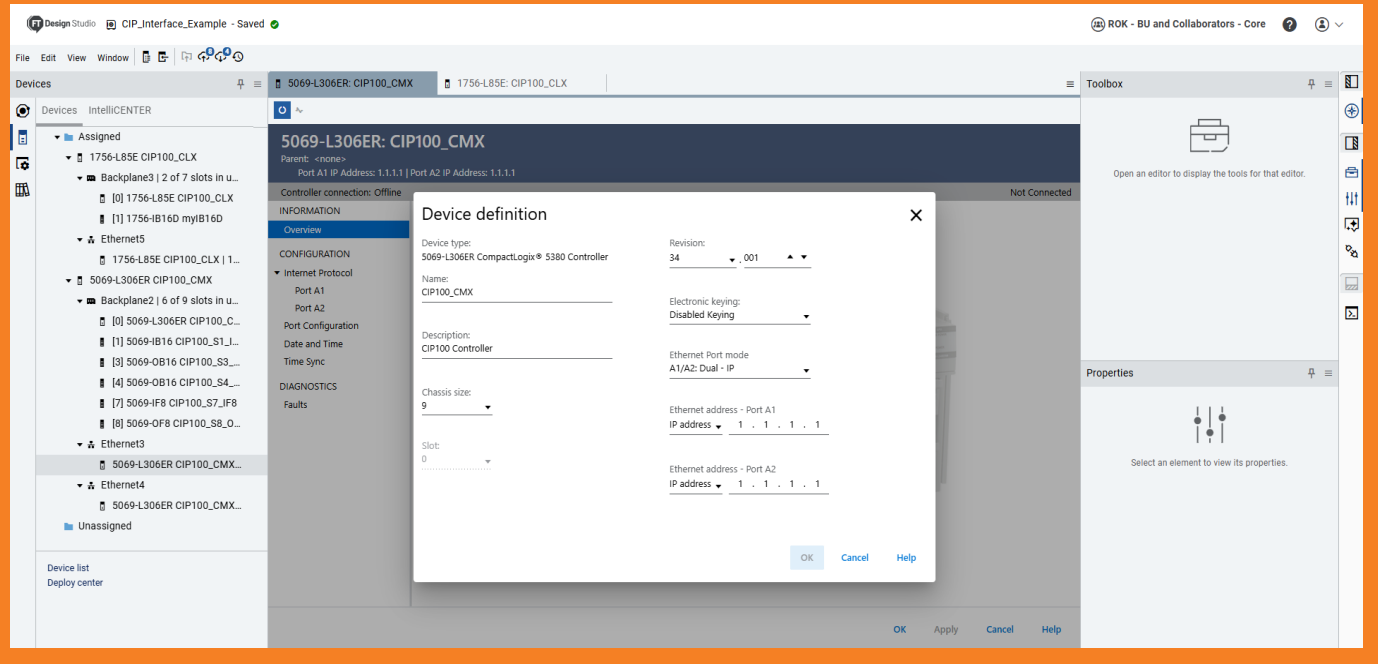
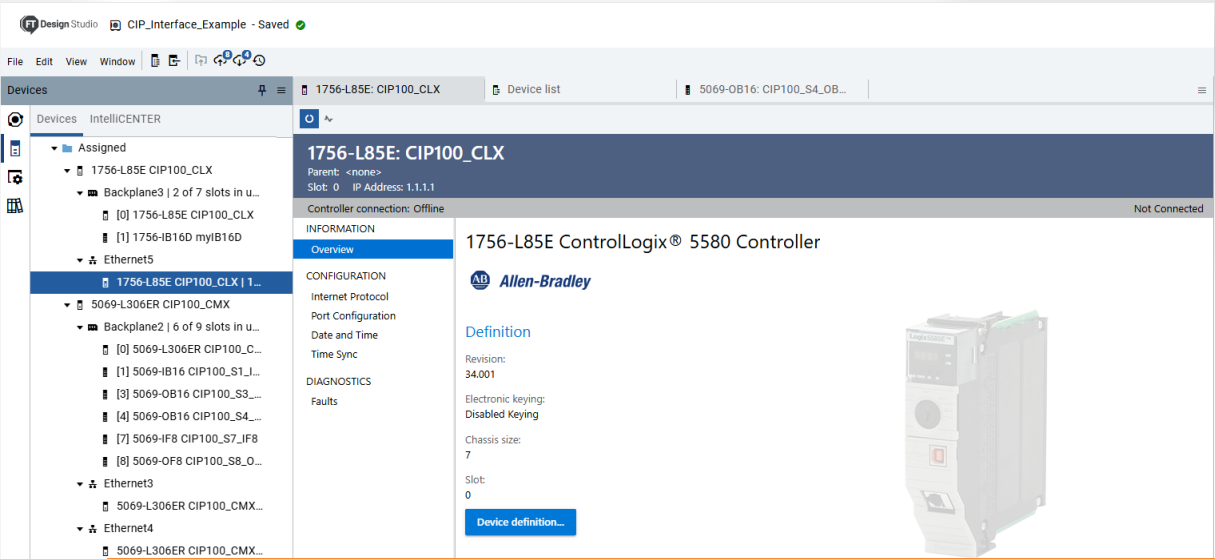
Improve design efficiency by adding and configuring devices in your project in multiple quantities at the same time

Choose from a core set of available devices in any quantity, then quickly define initial configuration of each in one workflow



# Device profile updates

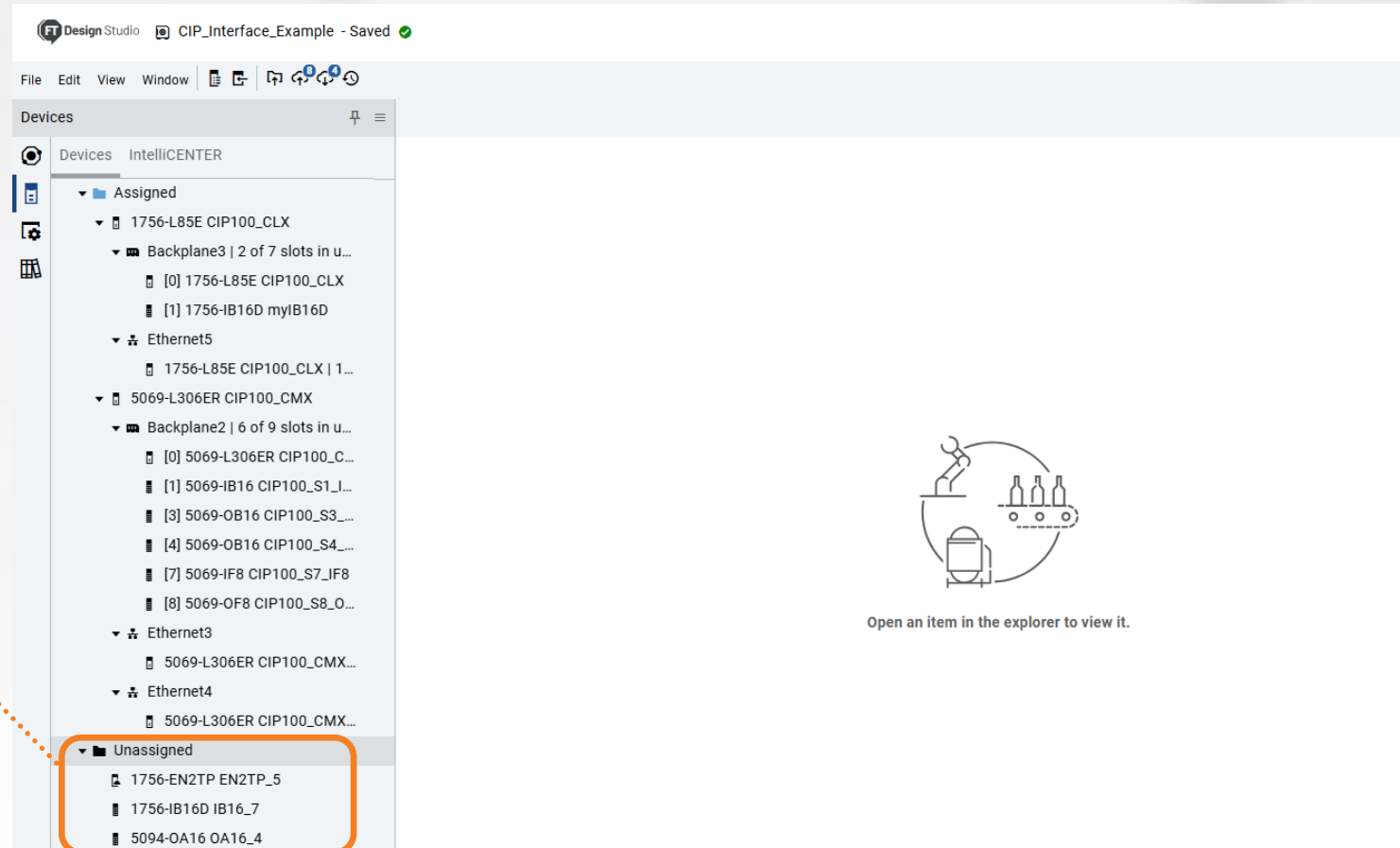
Manage controller and device configuration from built-in device profiles





# Unassigned devices

Add devices to your project then assign to a controller later



Change system topology at any time without losing your I/O or configuration (will automatically be reallocated as an unassigned device)

# Device list

See device firmware version, catalog number, catalog description, vendor and more

Design Studio CIP\_Interface\_Example - Saved

ROK - BU and Collaborators - Core

File Edit View Window

1756-L85E: CIP100\_CLX

Device list

Devices IntelliCENTER

Assigned

1756-L85E CIP100\_CLX

Backplane3 | 2 of 7 slots in u...

[0] 1756-L85E CIP100\_CLX

[1] 1756-IB16D myIB16D

Ethernet5

1756-L85E CIP100\_CLX | 1...

5069-L306ER CIP100\_CMX

Backplane2 | 0 of 9 slots in u...

[0] 5069-L306ER CIP100\_C...

[1] 5069-IB16 CIP100\_S1\_I...

[3] 5069-OB16 CIP100\_S3\_...

[4] 5069-OB16 CIP100\_S4\_...

[7] 5069-IF8 CIP100\_S7\_IF8

[8] 5069-OF8 CIP100\_S8\_O...

Ethernet3

5069-L306ER CIP100\_CMX...

Ethernet4

5069-L306ER CIP100\_CMX...

Unassigned

Search

Name	Description	Revision	Catalog Num...	Cate...	Vendor	Electronic Ke...
CIP100_CLX		34.001	1756-L85E	Controller	Rockwell Automation/Allen...	Disable Keying
CIP100_CMX	CIP100 Controller	34.001	5069-L306ER	Controller	Rockwell Automation/Allen...	Disable Keying
CIP100_S1_IB16		2.001	5069-IB16	Digital, Input	Rockwell Automation/Allen...	Compatible Mod...
CIP100_S3_OB16		3.001	5069-OB16	Digital, Out...	Rockwell Automation/Allen...	Compatible Mod...
CIP100_S4_OB16		3.001	5069-OB16	Digital, Out...	Rockwell Automation/Allen...	Compatible Mod...
CIP100_S7_IF8		2.001	5069-IF8	Analog, Inp...	Rockwell Automation/Allen...	Compatible Mod...
CIP100_S8_OF8		2.001	5069-OF8	Analog, Ou...	Rockwell Automation/Allen...	Compatible Mod...
myIB16D		3.001	1756-IB16D	Digital, Input	Rockwell Automation/Allen...	Compatible Mod...

Toolbox

Open an editor to display the tools for that editor.

Properties

Select an element to view its properties.

Easily view all devices in the system in one place

Device list can now be invoked from the Devices Explorer

# Device I/O tags – direct reference

Design Studio | ScreenCaptures - Saved

Execution | Device list | 5069-IF8: Te... | TempCheck1 | main | CLX\_Controller3 | Getting starte... | CMX\_Controller...

CLX\_Controller3 | P1 | main

Home | Execution | CLX\_Controller3 | Controller scoped tags and parameters

Name	Data Type	Value	Descripti...
ConveyorTempCheck2.I.Ch7Fault	BOOL	0	
ConveyorTempCheck2.I.ModuleFaults	INT	0	
ConveyorTempCheck2.I.AnalogGroupF...	BOOL	0	
ConveyorTempCheck2.I.Calibrating	BOOL	0	
ConveyorTempCheck2.I.CalFault	BOOL	0	
ConveyorTempCheck2.I.ChkStatus	INT	0	

Browse directly to the physical device tag

Design Studio | ScreenCaptures - Saved

Execution | Device list | 5069-IF8: Te... | TempCheck1 | main | CLX\_Controller3 | Getting starte... | CMX\_Controller...

CLX\_Controller3 | P1 | main

ConveyorTempCheck2.I.CalFault

This instruction is directly referencing device tags in the code

Output

100%

# Device I/O tags – binding physical to logical

FT Design Studio CIP\_Interface\_Example - Saved

File Edit View Window

Devices

Devices IntelliCENTER

Assigned

- 1756-L85E CIP100\_CLX
  - Backplane3 | 2 of 7 slots in use
    - [0] 1756-L85E CIP100\_CLX
    - [1] 1756-IB16D mylB16D
  - Ethernet5
    - 1756-L85E CIP100\_CLX | 1.1.1.1
- 5069-L306ER CIP100\_CMX
  - Backplane2 | 6 of 9 slots in use
    - [0] 5069-L306ER CIP100\_CMX
    - [1] 5069-IB16 CIP100\_S1\_IB16
    - [3] 5069-OB16 CIP100\_S3\_OB16
    - [4] 5069-OB16 CIP100\_S4\_OB16
    - [7] 5069-IF8 CIP100\_S7\_IF8
    - [8] 5069-OF8 CIP100\_S8\_OF8
  - Ethernet3
    - 5069-L306ER CIP100\_CMX | 1.1.1.1
  - Ethernet4
    - 5069-L306ER CIP100\_CMX | 1.1.1.1

FT Design Studio CIP\_Interface\_Example - Saved

File Edit View Window

System

- CIP100
  - TK100\_Wash\_Tank
    - V101\_Outlet\_Valve
    - CT100\_Conductivity
    - LT100\_Level
    - P100\_Caustic\_Pump
      - Control
        - main
    - T103\_New\_Inlet\_Valve\_Test
      - V100\_Water\_Valve
      - V102\_Recirc\_Inlet\_Valve
    - Return\_Header
    - Supply\_Header
    - TK200 Rinse Tank

Name	Data Type	Usage	Value	Style
Valve_Close_DI_ChPt	BOOL	INPUT	0	DECIMAL
▶ Valve_Cmds	IVALVE_CMDS	INPUT		
Valve_DO_ChPt	BOOL	OUTPUT	0	DECIMAL
Valve_Open_DI_ChPt	BOOL	INPUT	0	DECIMAL
▶ Valve_Sts	IRESPONES_STS	OUTPUT		
Add Tag...				

General Connections

Within Controller

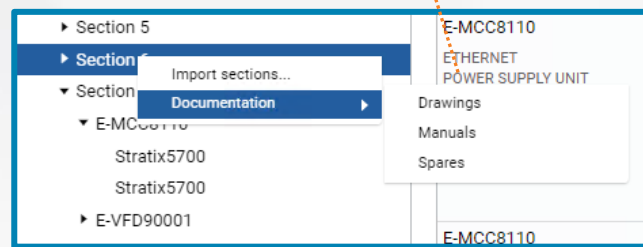
CIP100\_S1\_IB16.I.Pt00.Data

Smart object connections to physical devices preserve code modularity

Smart objects pass data between physical devices using connections

# IntelliCENTER® integration

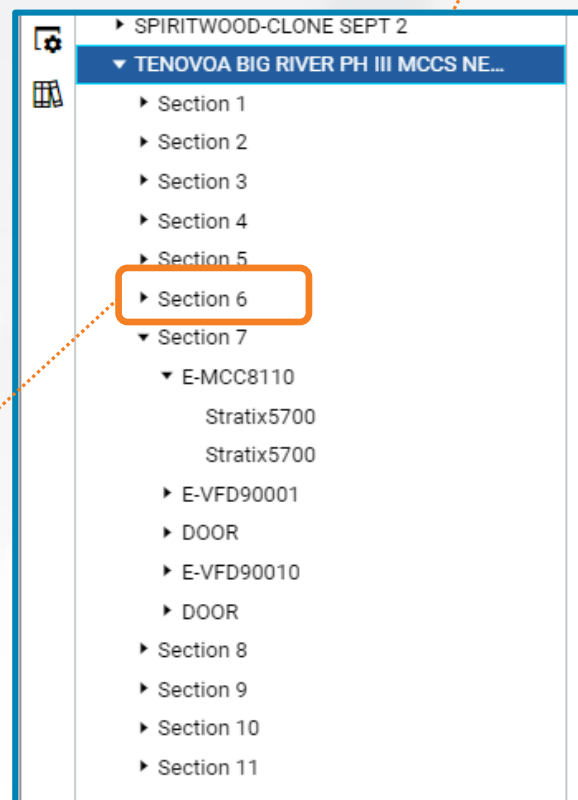
Double-click on an MCC section to access documentation such as electrical drawings, user manuals, and spare parts list



## Import an MCC lineup from the IntelliCENTER® tab in the Device Explorer

Easy access to lineup,  
section, and unit details

See a virtual representation of your MCC and online monitor the devices in your lineup to see current status.



→ ↩ ↺

ftdsdc012adhoc19.ftds-dev.rockwellautomation.com

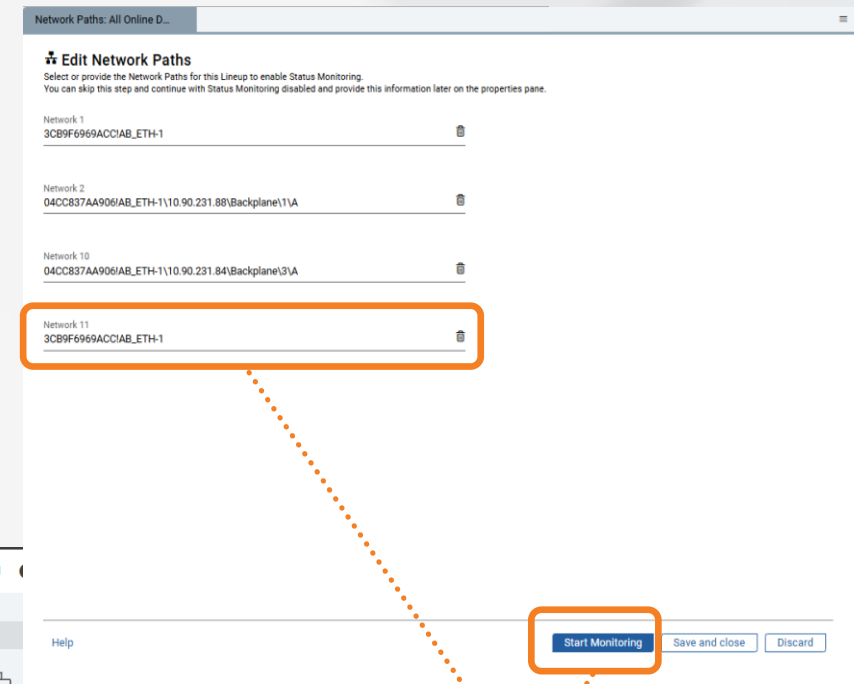
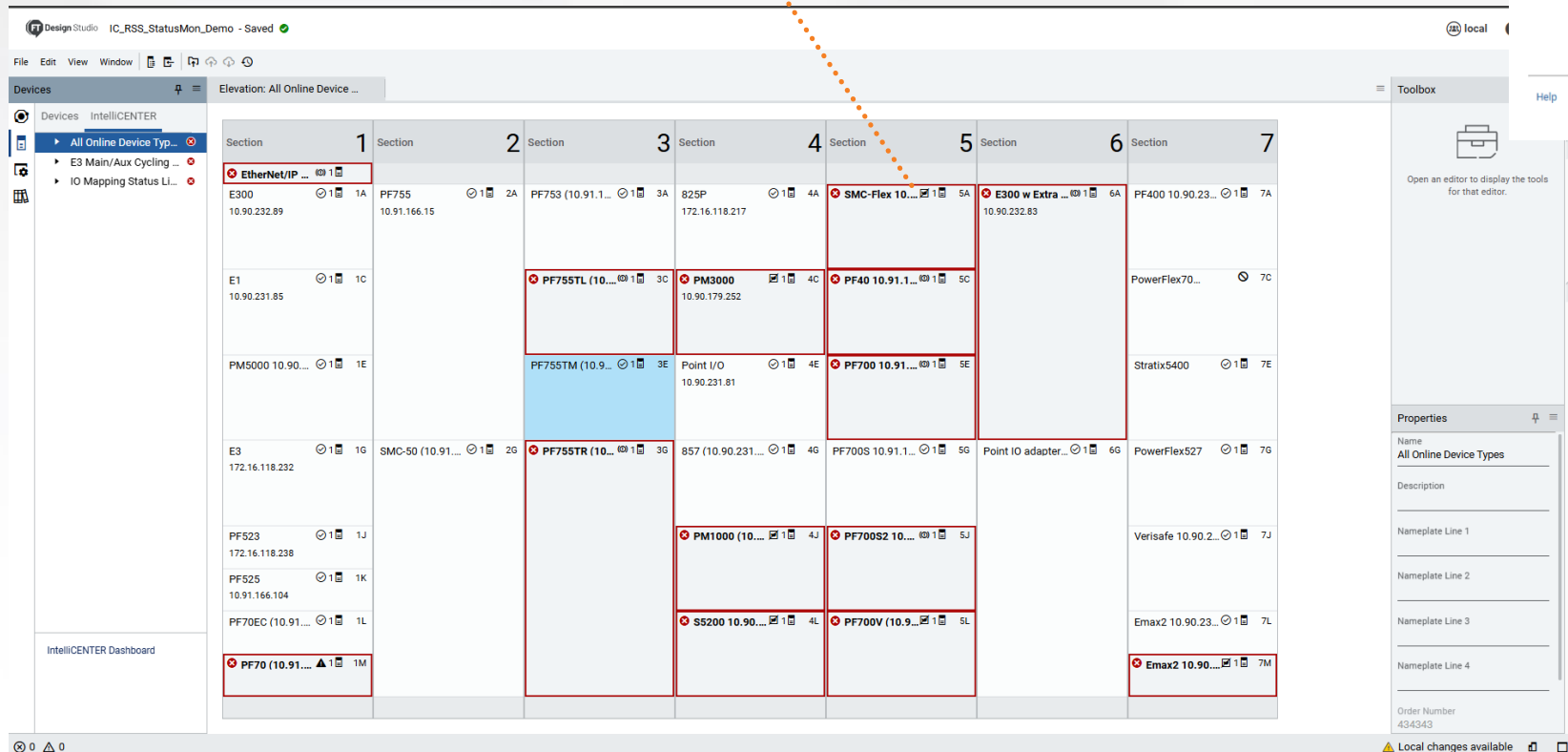
Design Studio

Large\_MCC\_3 - Saved

File Edit View Window Help

# Status monitoring of an MCC lineup

Status indicators for devices in an MCC lineup will show ready, faulted, or warning states. Devices with faulted or warning status will be outlined in red. Note the absence of green power bars such as you would see when online monitoring a controller. Status monitoring is different than online monitoring of a controller.



Set the network path to start monitoring status of devices in the lineup.



# IntelliCENTER® view updates

As of May 2025 (FTDS 2.01 update), we are enhancing some of the IntelliCENTER® views to make more detailed information available.

### I/O Mapping: E300 10.90.119.107

Status Monitoring Disabled

E300

**E300**

E300

E300

Unit Level view that allows easy viewing of all devices in a unit

E300\_10\_90\_232\_92\_Network\_2

The input and output mappings for this device are listed below.

Input

Output

OUT0

RVAT Speed 2 Coil

OUT1

RVAT Speed 1 Coil

OUT2

RVAT Run Coil

Just main warning | Location: 1A

Status Monitoring Enabled

Warning Status ▲

Unit Type

my unit

Documentation

Drawings

Manuals

Spares

2

DEVICES

Devices

Main E3

Auxiliary E3

Status ▲ Warning

Status ○ Ready

Fault/Warning Codes

Code 128

Clickable Explorer

Device Name	Device Type	IP Address	MCC Network Number
E3_172.16.118.223_Network_2	E3	172.16.118.223	2
E3_172.16.118.228_Network_2	E3	172.16.118.228	2

Unit level I/O mapping view now allows viewing of input and output separately for each device in the unit and also indicates which devices don't have I/O mappings

If a device returns a fault or warning code, it can be interrogated to learn more. This can be done by either hovering over the device or visiting the unit profile. Not only is the warning code exposed, but a link to helpful documentation can provide direction in mitigation efforts.

# IntelliCENTER® view updates in popular E300 parameters

**I/O Mapping: PF525 (10.91.166.104)**

Status Monitoring Enabled Running Status

PowerFlex525

E3

IO State

IO Mapping Definition

E3\_172\_16\_118\_232\_Network\_2

The input and output mappings for this device are listed below.

Input	Output	Device Status	Device Status when Off	Device Status when On
IN1	Fault Speed 1	None	None	Faulted
IN2	Select a definition	None	None	None
IN3	OverLoad	None	None	Faulted
IN4	AlwaysRunningInput	Running	Running	Running

Status

PF755T (10.91.166.220) | 1J

Status Monitoring Enabled

Nameplates

Only Inputs and Only Outputs

Documentation

Drawings

Manuals

Spares

2 DEVICES

Devices

Main PowerFlex755T Status Faulted Code 35

Auxiliary E300 Status Ready

8 - E300 Parameters

View Bit enumerated parameter details

Parameter Details

Parameter Name

TripEnable

This dialog is not currently monitoring the value

<input checked="" type="checkbox"/> Bit 0 - OverloadTrip	<input type="checkbox"/> Bit 8 - L2UnderCurrTrip
<input checked="" type="checkbox"/> Bit 1 - PhaseLossTrip	<input type="checkbox"/> Bit 9 - L3UnderCurrTrip
<input type="checkbox"/> Bit 2 - GroundFaultTrip	<input type="checkbox"/> Bit 10 - L1OverCurrentTrip
<input type="checkbox"/> Bit 3 - StallTrip	<input type="checkbox"/> Bit 11 - L2OverCurrentTrip
<input type="checkbox"/> Bit 4 - JamTrip	<input type="checkbox"/> Bit 12 - L3OverCurrentTrip
<input type="checkbox"/> Bit 5 - UnderloadTrip	<input type="checkbox"/> Bit 13 - L1LineLossTrip
<input type="checkbox"/> Bit 6 - CurrentImbalTrip	<input type="checkbox"/> Bit 14 - L2LineLossTrip
<input type="checkbox"/> Bit 7 - L1UnderCurrTrip	<input type="checkbox"/> Bit 15 - L3LineLossTrip

Properties

I/O

Parameters

FLASetting

0.50 Amps

TripClass

10

TripEnable

00000000 00000011

**IO Mapping**

- Select new IO Mapping definition from a provided list of definitions
- View individual input or output on/off states
- View impact of the individual Input/Output to the overall device status based on selected IO Mapping definition

**Unit Profile – E300 Parameters**

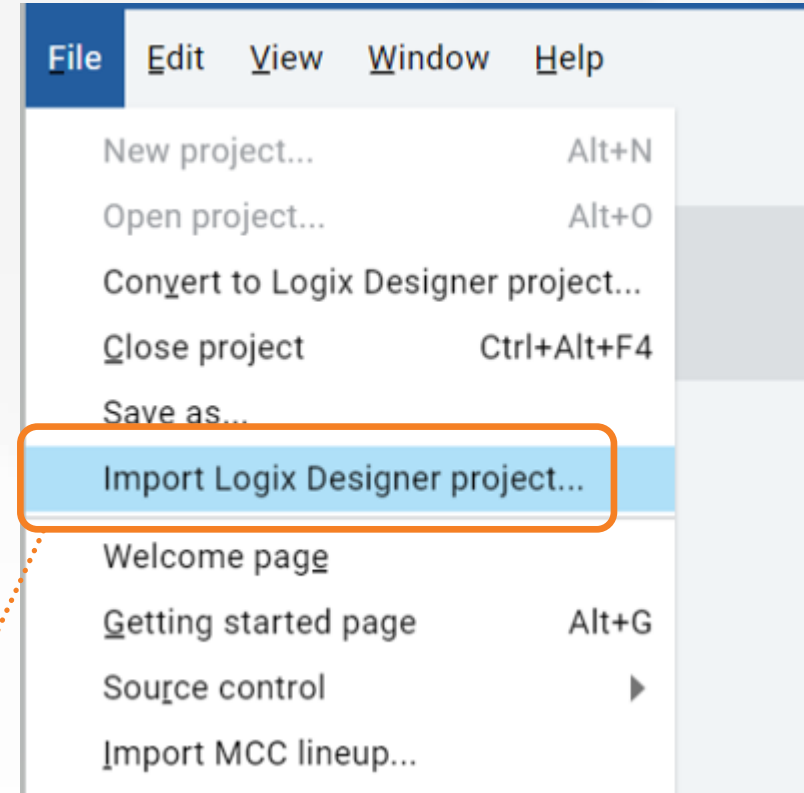
- View 8 most used E300 Parameters
- View current values of the 8 E300 Parameters

# Add content from a Studio 5000 Logix Designer® project

Currently supported during the import process:

- Controller (if controller not recognized by FactoryTalk® Design Studio™, you will receive a message on import accompanied by a log file)
- Ladder logic
- Tasks, programs, routines
- AOIs and UDTs (note that ALL data types are supported yet; and nested UDTs will not be resolved as expected)

Import code you have already created in Logix Designer application and start taking advantage of the value of cloud-based design



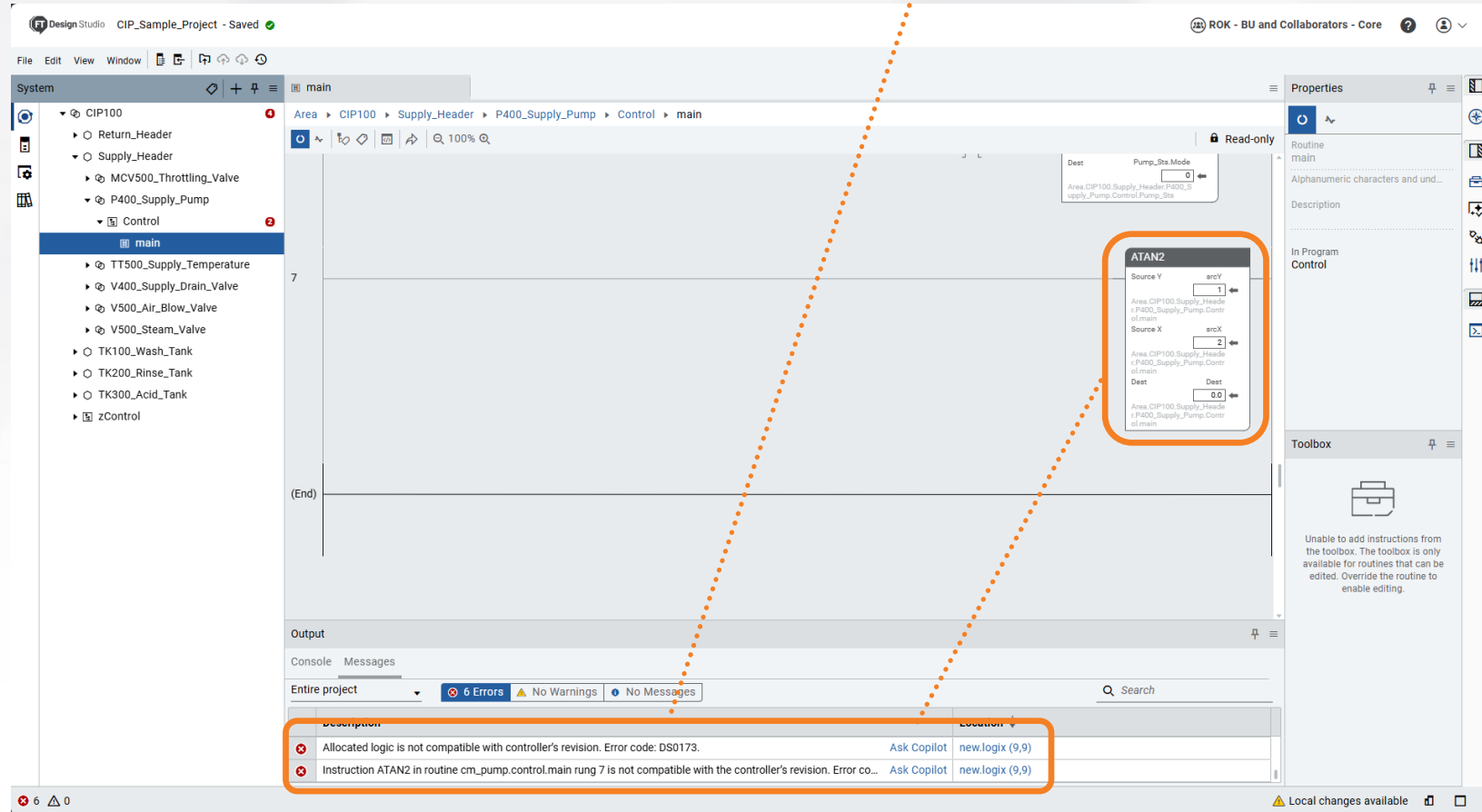
# Multiple revision support

A key aspect of a multi-controller design environment

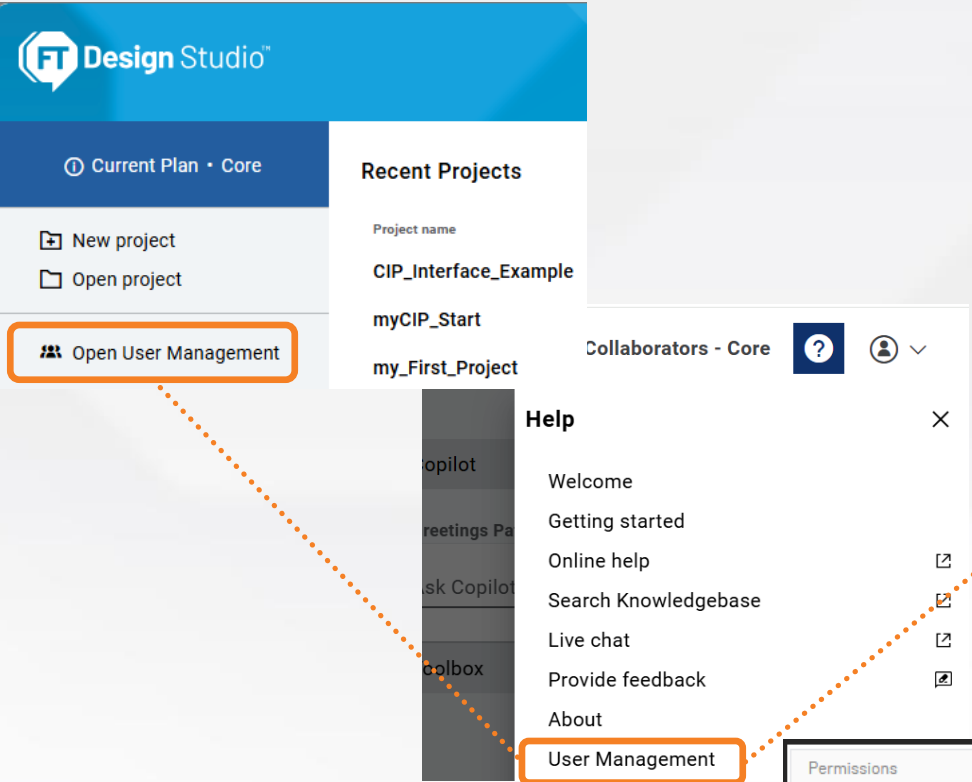
Develop, deploy, and manage multi-controller projects from the same ecosystem **without having to manage firmware in the controller.**

- MRS helps prevent users from **deploying content** that a controller's firmware does not support
- MRS **provides feedback during code development** that assists users in making informed decisions on actions that will ultimately result in code validation errors.
- Currently supported:
  - ControlLogix® and CompactLogix® L8 controllers
  - Firmware versions V34, V35, V36, and V37

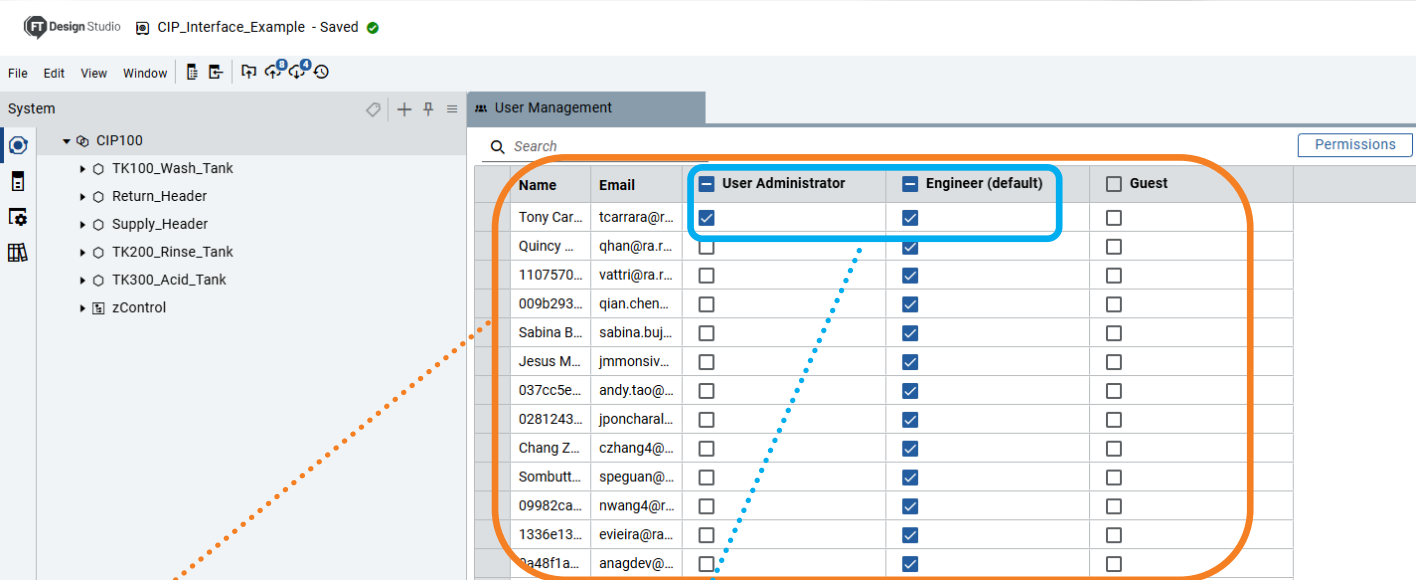
Users will be alerted when attempting to deploy code to a controller whose current firmware doesn't support certain instructions



# Role-based access control



Manage access to your projects by choosing the appropriate role for the user's job function.



The owner of the FactoryTalk® Hub™ organization will have both Admin and Engineer roles by default. If you are an admin at the FactoryTalk® Hub™ level, you will be an admin at the service level.

Permissions	Guest role	Engineer role	Administrator role
Create a project	X	X	
Copy / Save as		X	
Open a project	X	X	
Manage user roles			X
Version control Push		X	

User access by assigned role

The administrator role is responsible for managing permissions and access levels in the FactoryTalk® Hub™ organization.

# AI-powered design with the integrated Generative AI copilot

## Product guidance

PRODUCT GUIDANCE:  
Answers questions about  
product functionality and  
guidance on errors

ASK COPILOT:  
Use the Ask Copilot function in  
the output pain to help you  
quickly diagnose and mitigate  
errors in your project

The screenshot displays the FactoryTalk Design Studio interface. On the left, a project tree shows the hierarchy: CIP100 > PMP001 > control > main. The main workspace shows a ladder logic diagram with rungs for Start\_PB, Stop\_PB, Pump\_Run, Start\_Timer.DN, Alarm\_Timer.DN, and Alarm. On the right, the Copilot chat window is open, displaying a message from the user: "Can you please explain the value of a smart object". Below this, the Copilot provides a detailed explanation of a FactoryTalk Smart Object, including its purpose and key features. At the bottom, the Console Messages pane shows 11 errors, with the first three being related to invalid styles for COUNTER and TIMER, and the last three being related to an unavailable data type [STRING]. The 'Ask Copilot' button is highlighted in the console for each error.

Design Studio myCIP\_Start - Saved

File Edit View Window

System

Area: PMP001 > control > main

Start\_PB Area.PMP001.control.main

Stop\_PB Area.PMP001.control.main

Pump\_Run Area.PMP001.control.main

Start\_Timer.DN PMP001.control.main.Start\_Ti

Start\_Timer.DN PMP001.control.main.Start\_Ti

Alarm\_Timer.DN PMP001.control.main.Alarm\_Ti

Pump\_Run Area.PMP001.control.main

Alarm Area.PMP001.control.main

TON

Timer Start\_Timer

Preset 3000

Accum 0

TON

Timer Alarm\_Timer

Preset 2000

Accum 0

Console Messages

Entire project 11 Errors No Warnings No Messages

Description	Location
'NULL_STYLE' is an invalid style for COUNTER. Please refer to the Online help content for valid styles. Error code: DS0190.	Sequencer.logix (22,18)
'NULL_STYLE' is an invalid style for TIMER. Please refer to the Online help content for valid styles. Error code: DS0190.	Sequencer.logix (196,18)
Data type [STRING] is unavailable for use. Error code: DS0138.	M_Dvc_Valve_Fdbk.logix (399,6)
Data type [STRING] is unavailable for use. Error code: DS0138.	M_Dvc_Valve_Fdbk.logix (438,6)
Data type [STRING] is unavailable for use. Error code: DS0138.	M_Dvc_Valve_Fdbk.logix (0,0)

Ask Copilot

Ask Copilot

Ask Copilot

Copilot

As your AI assistant, I fulfill requests to accelerate your efficiency in FactoryTalk Design Studio.

I'm powered by AI, so surprises and mistakes are possible. Make sure to verify suggestions and share feedback so that I can learn and improve.

Can you please explain the value of a smart object

A FactoryTalk Smart Object is a powerful element in FactoryTalk Design Studio used to model system elements and their relationships. Here's an explanation of its value:

**Key Features and Benefits:**

- Reusability:**
  - Smart Objects allow you to create reusable logic for system components. For example, you can create a Smart Object for a tank in a plant and reuse it for all tanks, avoiding the need to create individual logic for each tank.
- Containment:**
  - A Smart Object acts as a container that can hold:
    - Other Smart Objects (child objects).
    - Programs (representing executable behaviors).
    - Routines (specific logic within programs).
- Inheritance and Override:**
  - Smart Objects can inherit behaviors from definitions, enabling consistent logic across instances.
  - You can override elements (e.g., programs, routines, tag values) in a derived Smart Object to customize its behavior without altering the base definition.
- Instance-Specific Data:**
  - Each Smart Object instance can have unique data, such as:
    - Program parameters.
    - Routine parameters.
    - Local tags.
  - This allows flexibility in configuring individual instances while maintaining a shared definition.
- Improved System Modeling:**
  - Smart Objects enhance system modeling by representing real-world elements and their relationships in a structured and organized manner.
- Allocation to Controllers:**
  - Programs within Smart Objects can be allocated to controllers and tasks, enabling deployment to logic controllers.
- Support for IEC 61131-3 Standards:**
  - Smart Objects follow IEC 61131-3 rules for overriding elements, ensuring compatibility with industry standards.

**Types of Smart Objects:**

- Standalone Smart Objects:**
  - Created independently in the System Explorer.
  - Useful for testing functionality or creating single, non-reusable objects.

Ask Copilot



# AI-powered design with the integrated Generative AI copilot

## Project guidance and creation

**PROJECT GUIDANCE:**  
Answers questions about your project, including project and code summaries and device information

**PROJECT CREATION:**  
Use natural language prompts to create project content, generate ladder logic code, task creation, and add comments to existing code

Design Studio myCIP - Saved

This operating system and browser are not supported. For full support use FactoryTalk Design Studio in Google Chrome on Windows.

File Edit View Window

System

- ▼ CIP100
  - TK100\_Wash\_Tank
  - Return\_Header
  - Supply\_Header
  - TK200\_Rinse\_Tank
    - V201\_Outlet\_Valve
    - CT200\_Conductivity
    - LT200\_Level
    - P200\_Rinse\_Pump
    - Control
      - main
      - V200\_Inlet\_Valve
      - V200\_Water\_Valve
      - TK300\_Acid\_Tank
      - zControl
  - ▼ PMP001
    - Control
      - main

Area PMP001 Control main

Start\_PB Area.PMP001.Control.main

Stop\_PB Area.PMP001.Control.main

Pump\_Run Area.PMP001.Control.main

Start\_Timer DN Area.PMP001.Control.main

Start\_Timer DN Area.PMP001.Control.main

Alarm\_Timer DN Area.PMP001.Control.main

Pump\_Run Area.PMP001.Control.main

Alarm\_Timer DN Area.PMP001.Control.main

TON Area.PMP001.Control.main

TON Area.PMP001.Control.main

Copilot

- **XIO(\_Manual)**: The system is already in manual mode or transitioning to it.
- **XIO(PB\_Off)**: The PB\_Off push button must not be active (not pressed).
- **XIO(PB\_auto)**: The PB\_Auto push button must not be active (not pressed).
- **OTE(\_Manual)**: This output instruction sets the \_Manual tag to true, enabling manual mode.

**Summary:**

Manual mode is enabled when the PB\_Manual button is pressed, and neither the PB\_Off nor PB\_Auto buttons are active. This ensures that the system transitions to manual mode only when explicitly requested and not in conflict with other modes. Let me know if you need further clarification!

Using definitions from my library, create nested Smart Objects for a conductivity transmitter and a rinse pump under the CIP100 Rinse Tank.

The nested Smart Objects have been successfully created under the CIP100 Rinse Tank:

- Conductivity Transmitter:**
  - Name: CT200\_Conductivity
  - Description: "Conductivity Transmitter for Rinse Tank"
- Rinse Pump:**
  - Name: P200\_Rinse\_Pump
  - Description: "Rinse Pump for Rinse Tank"

Let me know if you need further assistance or additional configurations!

Create a standalone pump Smart Object named PMP001. Under PMP001, create a program named Control. Create a routine with simple motor control with start, stop and alarm reset using Rockwell examples. Start the pump after a 3-second maintained press. Trip an alarm when the pump does not start 2 seconds after the delay.

Would you like me to overwrite the existing routine or create a new routine with a different name?

overwrite

The routine main under the program Control in the smart object PMP001 has been successfully overwritten with the new logic. Let me know if you need further assistance!

Ask Copilot

Use natural language prompts to ask the Generative AI Copilot create library definitions such as smart objects, AOI, and UDTs.

# AI-powered design with the integrated Generative AI copilot

## Task creation

Use the Generative AI copilot to create tasks with specific properties and parameters.

Move programs from one task to another, then ask copilot to summarize the changes that were made in a tabular format.

Easily drag-and-drop that information into an excel spreadsheet for easy desktop access.

Copilot

• If you need further breakdown or have a specific format in mind, let me know!

can you create a second task under the ControlLogix84\_slot3 device. make it a higher priority, with a period of 50ms, and a watchdog of 75ms

A second task named HighPriorityTask has been created under the ControlLogix84\_slot3 device. This new task is configured as follows:

- Priority: 1 (higher than the default task)
- Period: 50 ms
- Watchdog: 75 ms
- Description: High priority 50ms task with 75ms watchdog for ControlLogix84\_slot3

If you need to allocate programs or smart objects to this new task, or want to adjust any settings, please let me know!

please regenerate the table that summarizes the list of controllers, apps, and tasks, all organized in a table, along with the main properties

Here is a summary table of all controllers, their associated controller apps, and tasks, including main properties:


Controller Name	Catalog Number	Controller App Name	Task Name	Task Type	Period (ms)	Priority	Watchdog (ms)	Description
CLX_CIP100	1756-L85E	LogixApp_CLX_CIP100	DefaultTask	PERIODIC	250	10	500	
CMX_a	5069-L340ER	LogixApp_CMX_a	DefaultTask	PERIODIC	250	10	500	
CM_b	5069-L340ER	LogixApp_CM_b	DefaultTask	PERIODIC	250	10	500	
ControlLogix84_slot3	1756-L84E	LogixApp_ControlLogix84_slot3	DefaultTask	PERIODIC	100	10	250	

Ask Copilot

Controller Name	Catalog Number	Major	Minor	Controller App Name	Task Name	Task Type	Period (ms)	Priority	Watchdog (ms)	Inhibit	Description
CLX_CIP100	1756-L85E	37	1	LogixApp_CLX_CIP100	DefaultTask	PERIODIC	250	10	500	FALSE	None
CMX_a	5069-L340ER	36	1	LogixApp_CMX_a	DefaultTask	PERIODIC	250	10	500	FALSE	None
CM_b	5069-L340ER	36	1	LogixApp_CM_b	DefaultTask	PERIODIC	250	10	500	FALSE	None
ControlLogix84_slot3	1756-L84E	37	1	LogixApp_ControlLogix84_slot3	DefaultTask	PERIODIC	100	10	250	FALSE	Area.CIP100.TK100_Wash_Tank.CT100_Co Area.CIP100.TK100_Wash_Tank.CT100_Co Area.CIP100.TK100_Wash_Tank.LT100_Lev Area.CIP100.TK100_Wash_Tank.LT100_Lev Area.CIP100.TK100_Wash_Tank.TK100_Wi Area.CIP100.TK100_Wash_Tank.TK100_Wi Area.CIP100.TK100_Wash_Tank.TK100_Wi Area.CIP100.TK100_Wash_Tank.TK100_Wi Area.CIP100.TK100_Wash_Tank.TK100_Wi Area.CIP100.TK100_Wash_Tank.V100_Wat Area.CIP100.TK100_Wash_Tank.V101_Outl Area.CIP100.zControl
ControlLogix84_slot3	1756-L84E	37	1	LogixApp_ControlLogix84_slot3	HighPriorityTask	PERIODIC	50	1	75	FALSE	High priority 50ms task with 75ms watchdog Area.CIP100.TK100_Wash_Tank.P100_Cau Area.CIP100.TK100_Wash_Tank.P100_Cau

PUBLIC • Copyright ©2025 Rockwell Automation, Inc.

49

 **Rockwell Automation**

# AI-powered design with the integrated Generative AI copilot

## Chat retention

When you are working in a project, then close and come back at a later date, previous copilot interactions are retained and appear in the copilot interface.

**Copilot**

- Each arrow (→) represents a parameter connection from the source (left) to the target (right).

If you need a more detailed or graphical diagram, please specify the format or tool you would like to use!

Please explain error DS0091: "Invalid argument format. Assign one of these valid formats: [Array, First Array Element]."

**Error DS0091: "Invalid argument format. Assign one of these valid formats: [Array, First Array Element]."**

**Explanation:**  
This error occurs in FactoryTalk Design Studio when you use an instruction that expects an array or an array element as its argument, but you provide a value in the wr requires an array, you must pass a tag defined as an array (e.g., MyArray), or if it requires a single element, you must pass a specific element (e.g., MyArray[0]).

**How to resolve:**

- Check the instruction's documentation to see what format is required for its argument.
- If the instruction requires an array, use a tag that is defined as an array.
- If the instruction requires a single element, use the correct array element (e.g., MyArray[0]).

**Summary:**  
Make sure the argument you assign matches the required format:

- Use an array tag if an array is required.
- Use a specific array element if a single element is required.

If you need help identifying the correct format for a specific instruction or tag, let me know!  
(Source: FactoryTalk Design Studio Knowledgebase)

Ask Copilot

Design Studio myCIP\_Start\_RSS - Saved

File Edit View Window 100% 100%

System CIP100 TK100\_Wash\_Tank CT100\_Conductivity LT100\_Level P100\_Caustic\_Pump P100\_Caustic\_Pump\_01 Control main

Area CIP100 TK100\_Wash\_Tank P100\_Caustic\_Pump\_01 Control extern\_routine

0

**Copilot**

can you explain to me what does the tag Pump\_DO\_ChPt do?

The tag **Pump\_DO\_ChPt** is an output tag of type **BOOL** (Boolean) in the program Area.CIP100.TK100\_Wash\_Tank.P100\_Caustic\_Pump.Control. As an output, it is typically used to represent the digital output command to the caustic pump—most likely to turn the pump on or off.

- Usage:** Output (BOOL)
- Purpose:** It is intended to be used as the digital output channel (DO\_ChPt) for controlling the pump hardware, such as energizing a relay or actuator to start or stop the pump.

If you need more details about how this tag is used in the logic or want to see where it is connected in the system, let me know!

what tags affect the Pump\_DO\_ChPt?

The tag **Pump\_DO\_ChPt** is controlled in the main routine of the program Area.CIP100.TK100\_Wash\_Tank.P100\_Caustic\_Pump.Control. The logic that affects this output is:

```
[XIC(_auto) XIC(Pump_Cmds.on), XIC(_manual)] OTE(Pump_DO_ChPt);
```

This means **Pump\_DO\_ChPt** will be energized (set to 1/ON) if either of the following conditions is true:

1. **\_auto** is true AND **Pump\_Cmds.on** is true
2. **\_manual** is true

**Tags that affect Pump\_DO\_ChPt:**

- \_auto** (local BOOL tag)
- Pump\_Cmds.on** (member of the input structure Pump\_Cmds)
- \_manual** (local BOOL tag)

Chat retention delivers seamless continuity and context awareness.

# AI-powered design with the integrated Generative AI copilot

Create, modify, or delete parameter connections

Use the copilot to create new connections with existing/new parameters, modify connection parameter data types, direction, and more

Promote tags from LOCAL to INPUT or OUTPUT, delete parameters and view a list of parameter connections in tabular format.

Copilot

Greetings Patricia,  
As your AI assistant, I fulfill requests to accelerate your efficiency in FactoryTalk Design Studio.  
I'm powered by AI, so surprises and mistakes are possible. Make sure to verify suggestions and share feedback so that I can learn and improve.

show me a list of all parameter connections in a table

Here is a table listing all parameter connections in the project:

Source Parameter (From)	Usage/Type	Target Parameter (To)	Usa
Area.CIP100.TK100_Wash_Tank.CT100_Conductivity.Simulation.Control.Level	INPUT / REAL	Area.CIP100.TK100_Wash_Tank.LT100_Level.Control.AI.PV	OUTP IANAI
Area.CIP100.TK100_Wash_Tank.CT100_Conductivity.Simulation.Control.PV	INPUT / REAL	Area.CIP100.TK100_Wash_Tank.CT100_Conductivity.Control.AI.PV	OUTP IANAI
Area.CIP100.TK100_Wash_Tank.CT100_Conductivity.Simulation.Control.Sim_Enabled	OUTPUT / BOOL	Area.CIP100.TK100_Wash_Tank.CT100_Conductivity.Control.Sim_Enabled	INPU'
Area.CIP100.TK100_Wash_Tank.CT100_Conductivity.Simulation.Control.Sim_PV	OUTPUT / REAL	Area.CIP100.TK100_Wash_Tank.CT100_Conductivity.Control.AI_ChPt_Sim	INPU'
Area.CIP100.TK100_Wash_Tank.CT100_Conductivity.Simulation.Control.Subtract_Fn1	INPUT / BOOL	Area.CIP100.TK100_Wash_Tank.V100_Water_Valve.Control.Valve_Sts.Request_Done	OUTP IRESF
Area.CIP100.TK100_Wash_Tank.LT100_Level.Simulation.Control.Add_Fn	INPUT / BOOL	Area.CIP100.TK100_Wash_Tank.V100_Water_Valve.Control.Valve_Sts.Request_Done	OUTP IRESF
Area.CIP100.TK100_Wash_Tank.LT100_Level.Simulation.Control.PV	INPUT / REAL	Area.CIP100.TK100_Wash_Tank.LT100_Level.Control.AI.PV	OUTP IANAI
Area.CIP100.TK100_Wash_Tank.LT100_Level.Simulation.Control.Sim_Enabled	OUTPUT / BOOL	Area.CIP100.TK100_Wash_Tank.LT100_Level.Control.Sim_Enabled	INPU'
Area.CIP100.TK100_Wash_Tank.LT100_Level.Simulation.Control.Sim_PV	OUTPUT / REAL	Area.CIP100.TK100_Wash_Tank.LT100_Level.Control.AI_ChPt_Sim	INPU'
Area.CIP100.TK100_Wash_Tank.LT100_Level.Simulation.Control.Subtract_Fn1	INPUT / BOOL	Area.CIP100.TK100_Wash_Tank.V101_Outlet_Valve.Control.Valve_Sts.Request_Done	OUTP IRESF

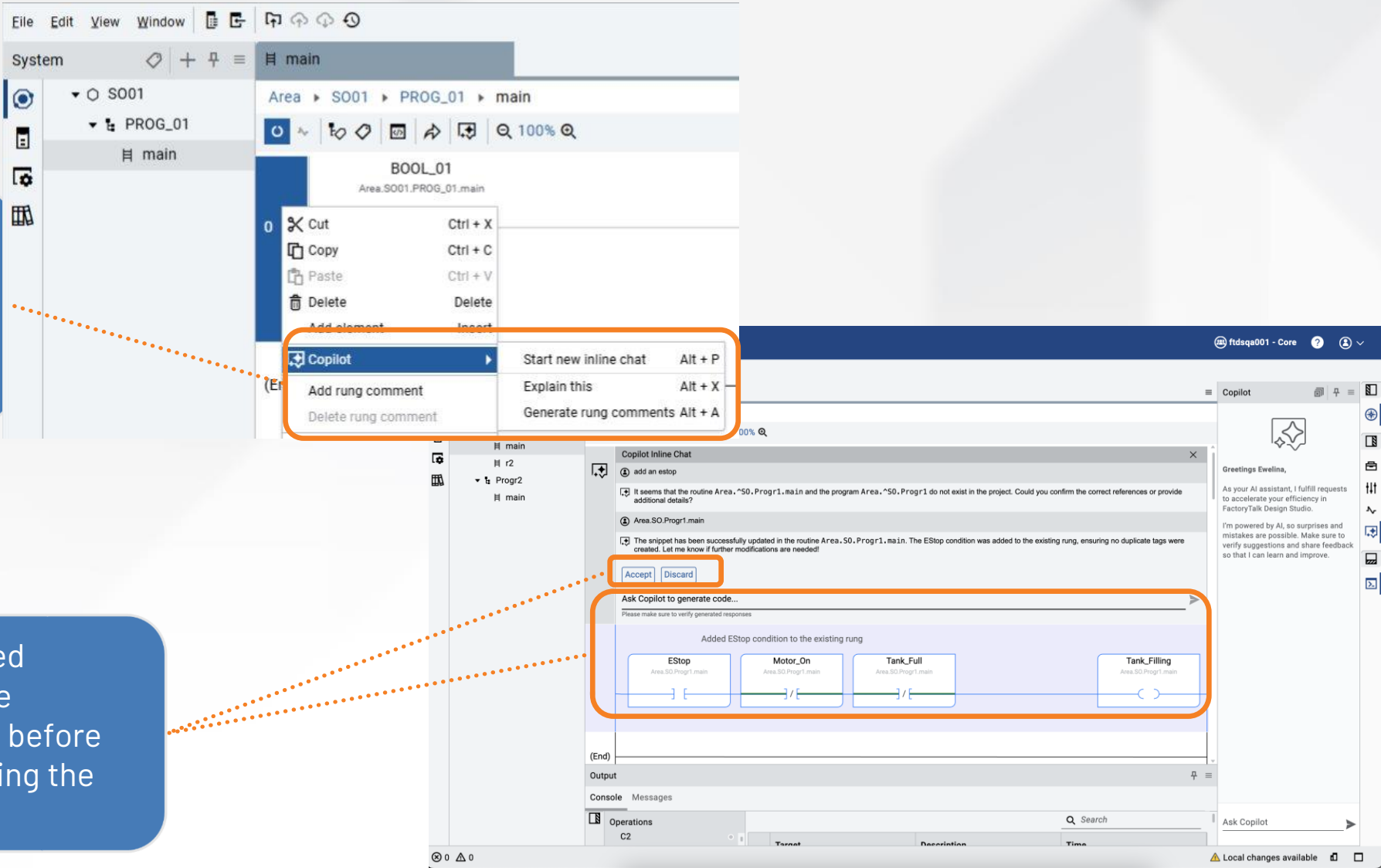
Ask Copilot

# AI-powered design with the integrated Generative AI copilot

Inline chat

Invoke the Generative AI assistant directly from the ladder editor for a more focused experience on the code you're actively working on

Preview the requested updates and continue iterating on the code before accepting or discarding the generated content



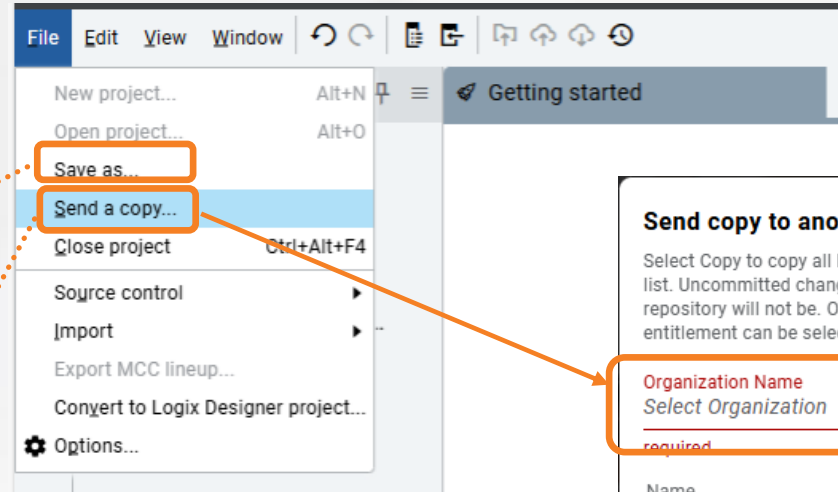


# Project transfer

Seamlessly move projects between organizations

File > Save as allows you to make a copy of your project.

File > Send a copy allows you to make a copy of your project and transfer it to another organization.



The organization where you want to transfer your project. This selection will show only organizations that have a VALID subscription of FTDS.

## Send copy to another organization

Select Copy to copy all local project changes and send to an organization from the list. Uncommitted changes will be included but changes not pulled from the remote repository will not be. Only organizations with an active FactoryTalk Design Studio entitlement can be selected.

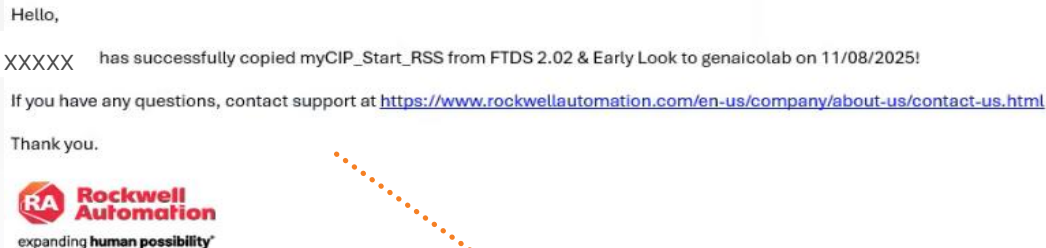
Organization Name  
Select Organization

Name  
myCIP\_Start\_RSS

Alphanumeric characters and underscores only

☐ Keep project history

Checking this option helps ensure that the project/change history will be retained and transferred with the project



The user transferring the project will receive an email from [FTDSTeam@rockwellautomation.com](mailto:FTDSTeam@rockwellautomation.com) confirming that the project has been successfully transferred. Any admins of that organization will also be copied

Project transfer is an especially useful capability for customers who work with delivery partners. Once design and commissioning are completed, SIs and OEMs can easily transfer the project for the operate and maintain part of the product / project lifecycle.



# AI-powered design demos (click on the video to watch)

## Product guidance



Demo Video

### WHAT YOU CAN DO

- Query online help topics from the chat panel
- Query error code documentation from the chat panel
- Ask Copilot how to resolve an error from the output pane

### TRY IT YOURSELF using these prompts:

- What new features are available?
- What devices are supported?
- How do I create a routine?
- Are motion instructions supported?

## Project guidance



Demo Video

### WHAT YOU CAN DO

- Gather information from your project
- Cross-reference tag and object usage
- Understand state and sequence logic

### TRY IT YOURSELF using these prompts:

- Summarize [smart-object-name].
- Explain the [routine-name] logic.
- What conditions prevent the [tag-name] from starting?

## Project creation



Demo Video

### WHAT YOU CAN DO

- Generate smart object singletons and instances
- Generate AOI definitions and instances
- Generate programs, routines, and PLC code
- Allocate and schedule programs

### TRY IT YOURSELF using this prompt:

- Create a pump Smart Object named PMP001. Under PMP001, create a program named Control. Update the main routine to create start and stop pump control logic using Rockwell examples. Start the pump after a 3-second maintained press. Trip an alarm when the pump does not start 2 seconds after the delay.

# Deploy code to controller(s)

Secure cloud-to-edge communications is now available for that end-to-end workflow of designing in the cloud and deploying to controllers in your automation system.

Select the deploy center from the menu and select a controller to deploy code to.

Set the controller path, then select DEPLOY.

Did YOU KNOW? You can deploy sections of code to different controllers in your system simultaneously.

Controller	Path	Deployment
<input type="checkbox"/> myCLX	Ethernet\...	

Target	Description	Time
myCLX	Deploy complete	02/06/2025, 04:50:56 PM
myCLX	Build complete	02/06/2025, 04:50:53 PM
myCLX	Starting deploy	02/06/2025, 04:50:49 PM

Controller	Path	Deployment
<input type="checkbox"/> CIP100_CLX	DemoEEC4\...	
<input type="checkbox"/> CIP200	DemoEEC4\...	

# Online monitor code and devices

After code deployment is completed, you can choose to go online with a controller in your system to monitor code and devices.

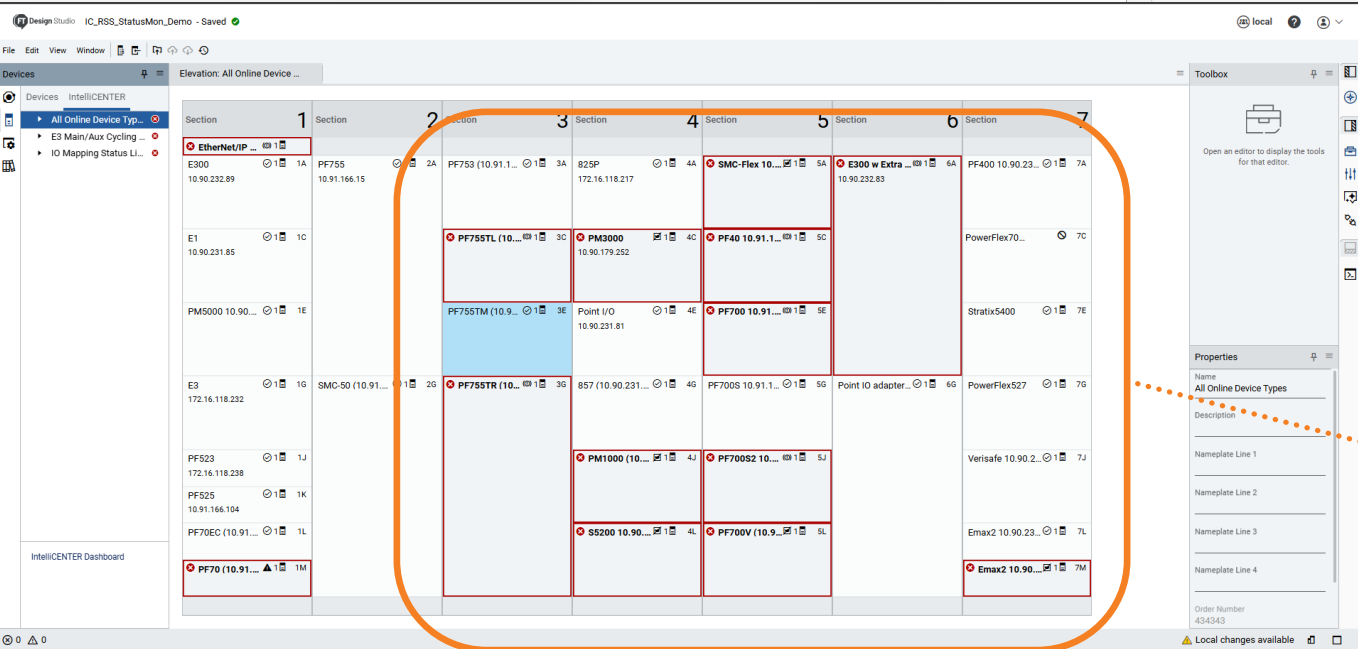
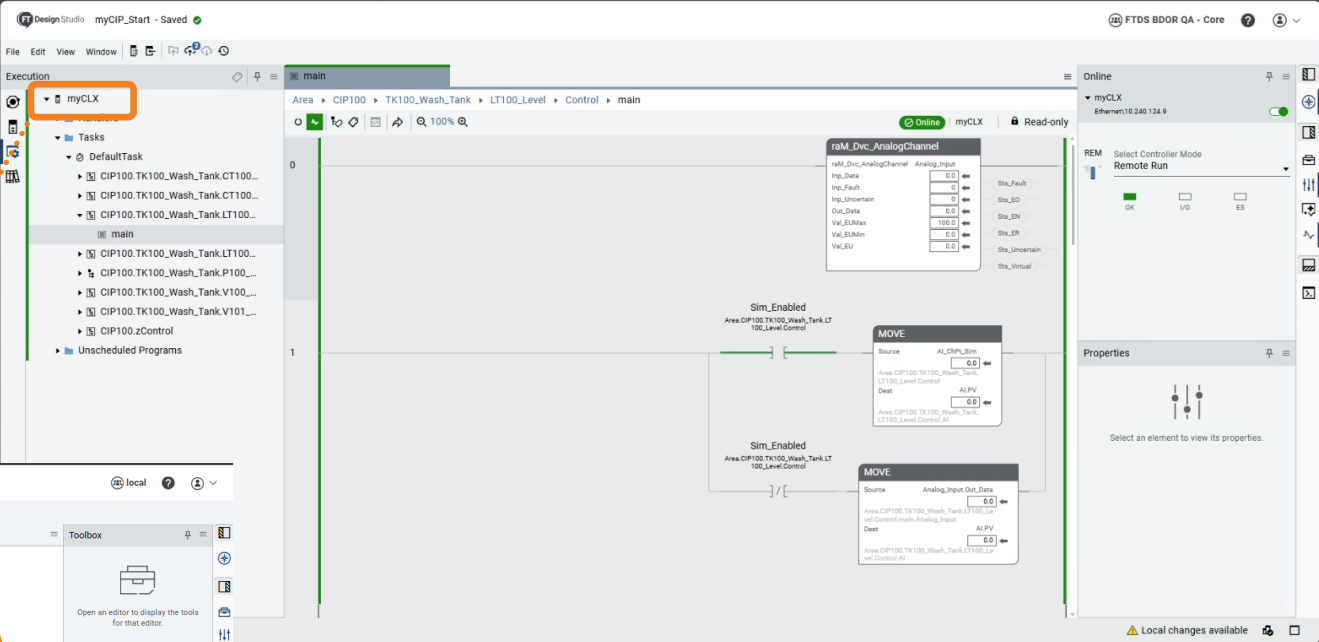
As of FactoryTalk® Design Studio™ 2.0 update, you can go online with 1 controller in your system to monitor code and devices.

Choose a controller to go online with after deployment ends:

- ☒ myCLX
- ☐ Go offline with all

Help

Ok

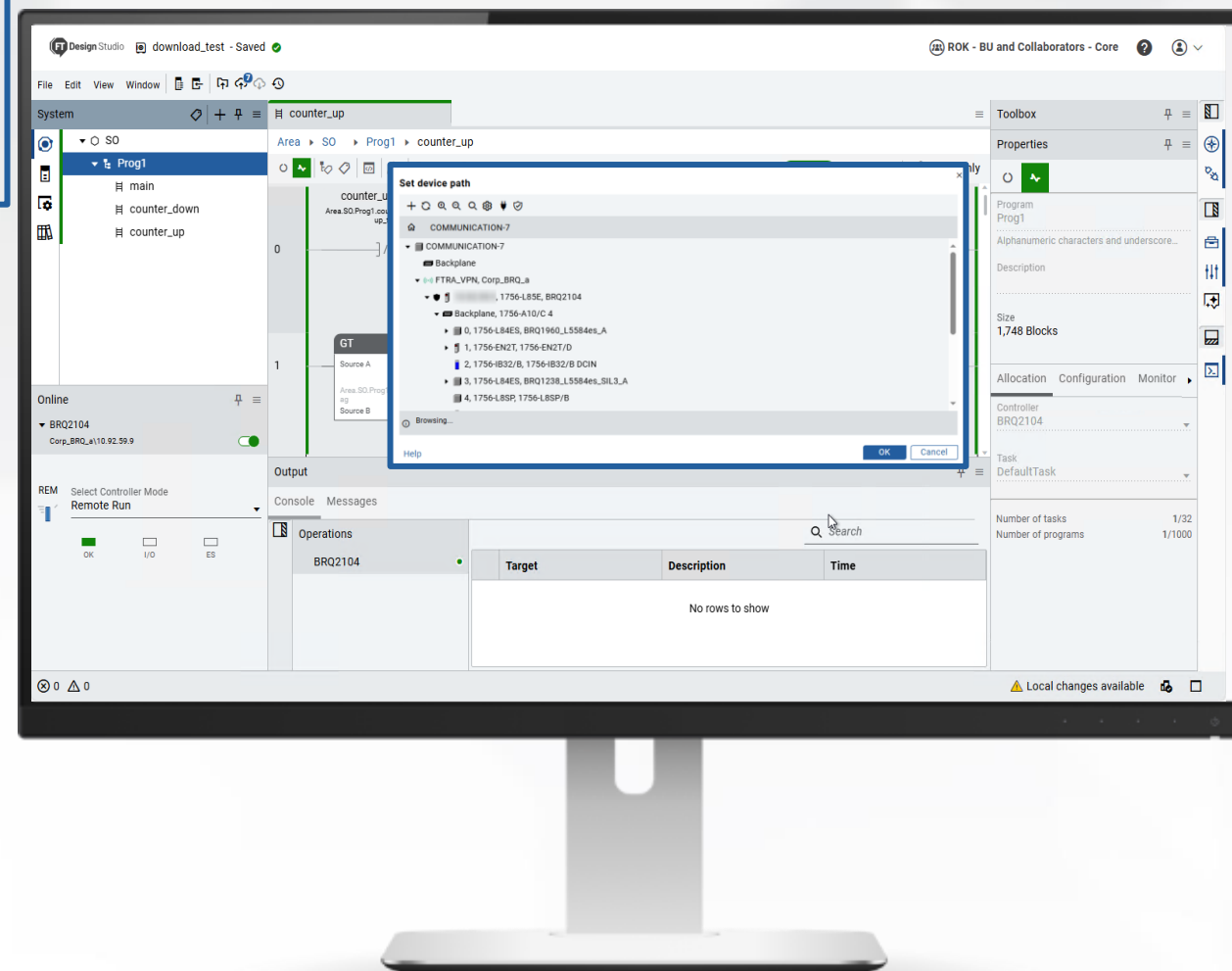


Status of devices in an IntelliCENTER® lineup can now be monitored as well. A quick look of those devices will tell you if they are ready, faulted, or have active warning indicators.

# Cloud-to-controller comms using FactoryTalk Remote Access

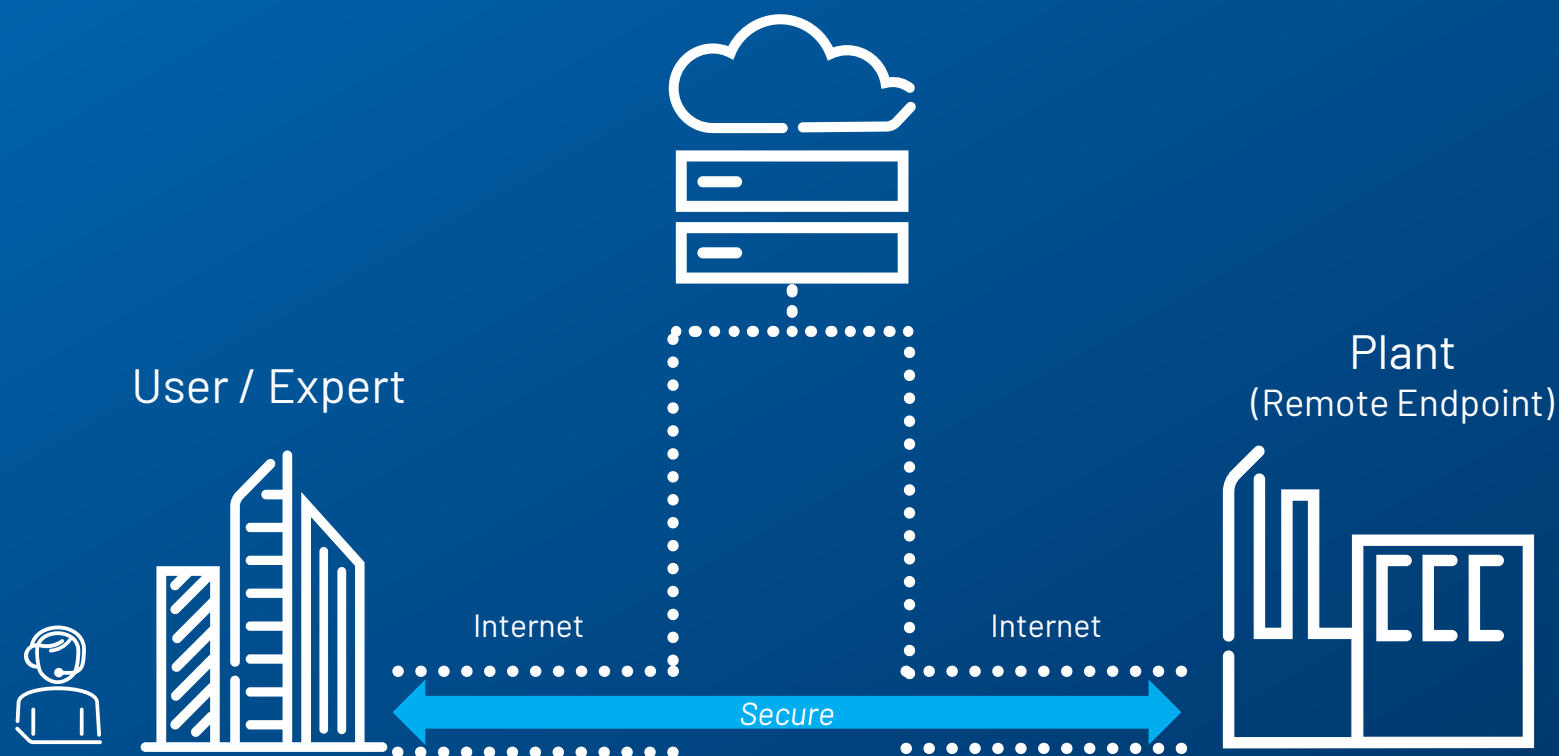
Configure VPN endpoint in FactoryTalk Remote Access (FTRA) to communicate from FactoryTalk® Design Studio™ to on-prem devices

- FactoryTalk Remote Access must be allocated to the same organization as FactoryTalk® Design Studio™ (FTDS)
- FTDS uses one FTRA connection per endpoint and requires only 1 entitlement
- Endpoint configuration is visible to all users and all projects in the organization
- There is no hard limit to the number of users using a single endpoint
- There is no hard limit to the number of users online with a controller



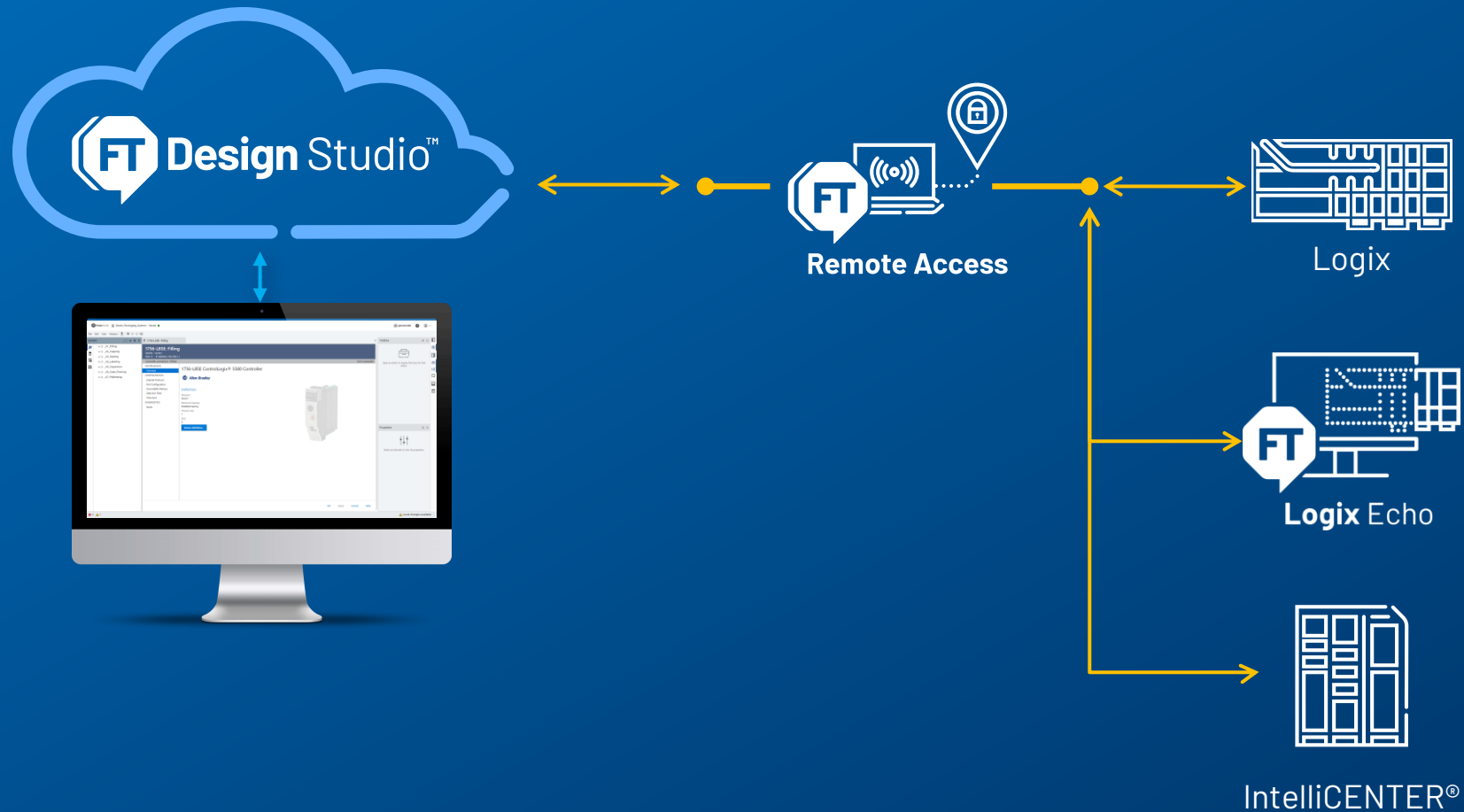
# Secure cloud-to-edge communications

Delivering secure communications over the internet to enable performance and security for on-demand remote assistance, installation, programming, troubleshooting, and maintenance of remote automation systems and applications.



# Connecting virtual design to edge devices

FactoryTalk® Design Studio™ cloud-to-edge communication





# FactoryTalk Remote Access endpoint options

A FactoryTalk® Remote Access™ Runtime entitlement is provided<sup>1</sup>

## OPTIXPANEL™ COMPACT

- Ships with FactoryTalk® Remote Access™ Runtime Basic entitlement
- Upgradeable to Runtime Pro entitlement

## OPTIXPANEL™ STANDARD

- Ships with FactoryTalk® Remote Access™ Runtime Pro entitlement



## EMBEDDED EDGE COMPUTE™

- Ships with FactoryTalk® Remote Access™ Runtime Pro entitlement
- Ships with FactoryTalk® Optix™ Runtime Xtra-Small entitlement
  - Upgradable to FactoryTalk® Optix™ Runtime Small - XL



## ASEM™ 6300 INDUSTRIAL PC<sup>2</sup>

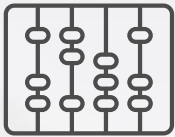
- Includes a FactoryTalk® Remote Access™ Runtime Basic entitlement
- Available on all ASEM™ 6300 iPCs, new and existing purchases.



<sup>1</sup>Remote access is not enabled or turned on by default, use the device's system manager to configure and enable remote access

<sup>2</sup>Download the FactoryTalk® Remote Access™ basic runtime from the [Product Compatibility & Download Center from Rockwell Automation](#)

## 2.02 update

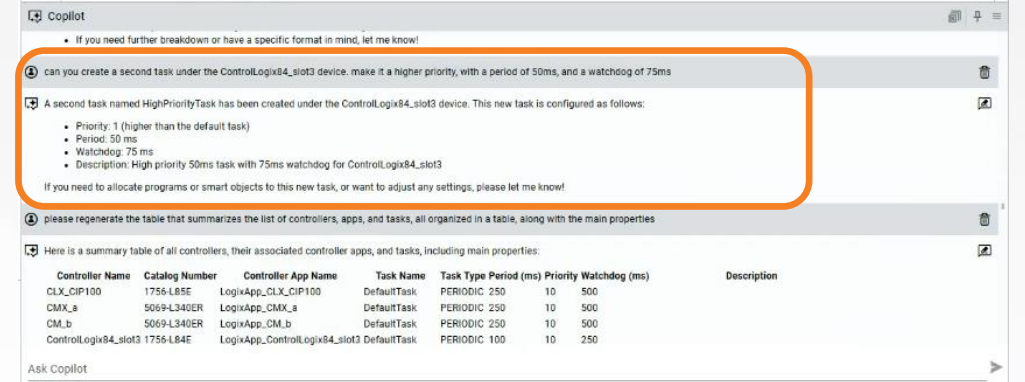
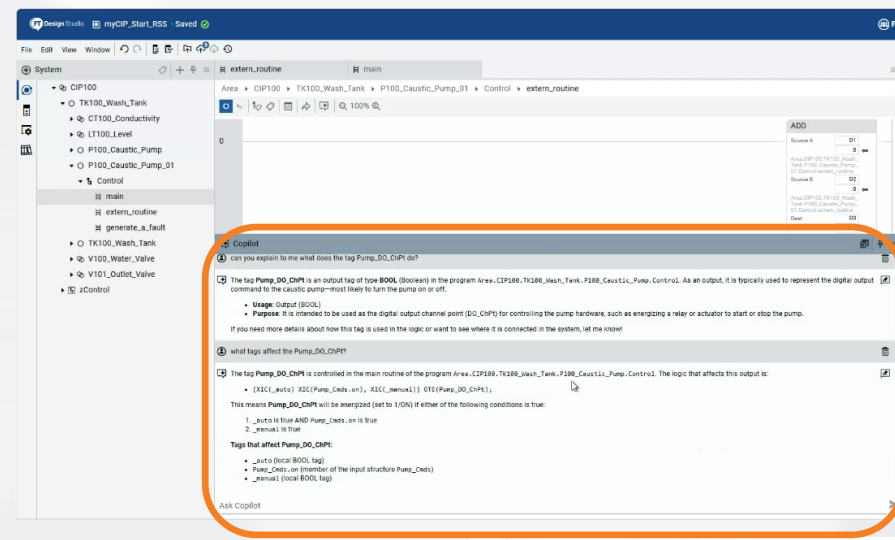


## AI-powered design and user experience enhancements

Updates to the integrated Generative AI copilot (task creation, chat history retention, parameter connections, issue VCS commands) deliver capabilities to improve the design experience.

Other updates:

- 843E encoders and Stratix® 5800 device support
- IntelliCENTER® I/O mapping and unit profile updates
- Performance and usability updates including undo/redo and secure project transfer between organizations



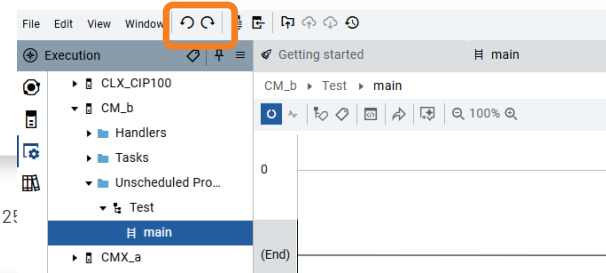
### Add devices to project

Show devices from

- ☒ Catalog
- ☐ Project

Category filter

Search	Catalog Number	Category	Description
	843E-MIPxxBAX	Encoder	Multi-turn Absolute Encoder
	843E-SIPxxBAX	Encoder	Single-turn Absolute Encoder



- ☒ Catalog
  - ☐ Project
- Category filter
- ☐ Analog
  - ☐ Communication
  - ☐ Controller
  - ☐ Digital
  - ☐ Encoder
  - ☐ Input
  - ☐ Miscellaneous

Catalog Number	Category	Description
<input type="checkbox"/> 1783-MMS10	Communication	Stratix 5800 10 Port Expandable Base
<input type="checkbox"/> 1783-MMS10B	Communication	Stratix 5800 10 Port Non-Expandable Base
<input type="checkbox"/> 1783-MMS10BE	Communication	Stratix 5800 10 Port Non-Expandable Base PoE
<input type="checkbox"/> 1783-MMS10E	Communication	Stratix 5800 10 Port Expandable Base PoE
<input type="checkbox"/> 1783-MMS10EA	Communication	Stratix 5800 10 Port Expandable Base PoE Advanced
<input type="checkbox"/> 1783-MMS10EAR	Communication	Stratix 5800 10 Port Expandable Base PoE Advance...
<input type="checkbox"/> 1783-MMS10ER	Communication	Stratix 5800 10 Port Expandable Base PoE Layer 3
<input type="checkbox"/> 1783-MMS10R	Communication	Stratix 5800 10 Port Expandable Base Layer 3



**The 2.02.01 update delivers more programming options!**



**Addition of another IEC 61131-3 programming language**

Structured Text is now available in the FTDS environment, providing users programming options when designing their automation system.

Structured text is ideal for:

- complex logic operations
- mathematical computations
- data handling and manipulation

Copilot also supports Structured Text, creating and deleting routines, creating code, changing routine types, and more based on user's natural language prompts.

**New routine...**

Name  
ST\_routine

Alphanumeric characters and underscores o...

Description

Select as Main Routine  
☐ Off

Select as Fault Routine  
☐ Off

Routine Type  
Ladder Diagram  
Structured Text

☒ Open New Routine

Help Create Cancel

HTX300

MainProgram

main

HTX\_Controls

ST\_Interlocks

```
File Edit View Window 100%
System HTX300 MainProgram ST_Interlocks
Area HTX300 MainProgram ST_Interlocks
1 routine st Heat_Exchange.MainProgram.ST_Interlocks() {
2   tag BOOL HTX_OKtoHeat {
3     desc := "Heat Exchanger is ready to start the heaters";
4   };
5   tag BOOL HTX_EStop_Circuit_OK {
6     desc := "Heat Exchanger E-Stop Circuit OK";
7   };
8   tag BOOL Main_EStop_Circuit_OK {
9     desc := "Main Machine E-Stop Circuit OK";
10  };
11  IF (HTX_OKtoHeat && HTX_EStop_Circuit_OK && Main_EStop_Circuit_OK) then // If all interlocks are satisfied
12    HTX_Interlocks := 1;
13  else
14    HTX_Interlocks := 0;
15  end_if;
16 }
17
18 }
```

AVAILABLE  
9/13/25



## INTEGRATED GENERATIVE AI

Drive breakthrough productivity using the integrated copilot for code generation, code explanation, troubleshooting, and product guidance

## AVAILABLE ANYTIME, ANYWHERE

- Browser-based, no download/install
- FactoryTalk® Hub™ integrated
- SaaS delivered, no patching/updating

## INTEGRATED VERSION CONTROL

- Project change history
- Text-based change comparison

## MULTI-USER COLLABORATION

- Shared projects & local projects
- Auto-merge changes from other users
- Detect & resolve conflicts

## CLOUD-TO-EDGE COMMUNICATION

- Secure end-to-end workflow for design to deploy
- Use FactoryTalk® Remote access to download to FactoryTalk® Logix Echo, ControlLogix®, or CompactLogix® controllers



## LOGIX PROGRAMMING

- Ladder Diagram Editor & Tag Editor
- Text-based Code Editor
- Routine-scoped tags

## MULTI-CONTROLLER SUPPORT

- Logical system modeling
- Hardware abstracted design
- Multiple controllers, single project

## INTEGRATED LIBRARY MANAGEMENT

- FactoryTalk® Smart Objects, AOIs, UDTs
- Definition and instance inheritance

# Learn more

## FACTORYTALK HUB

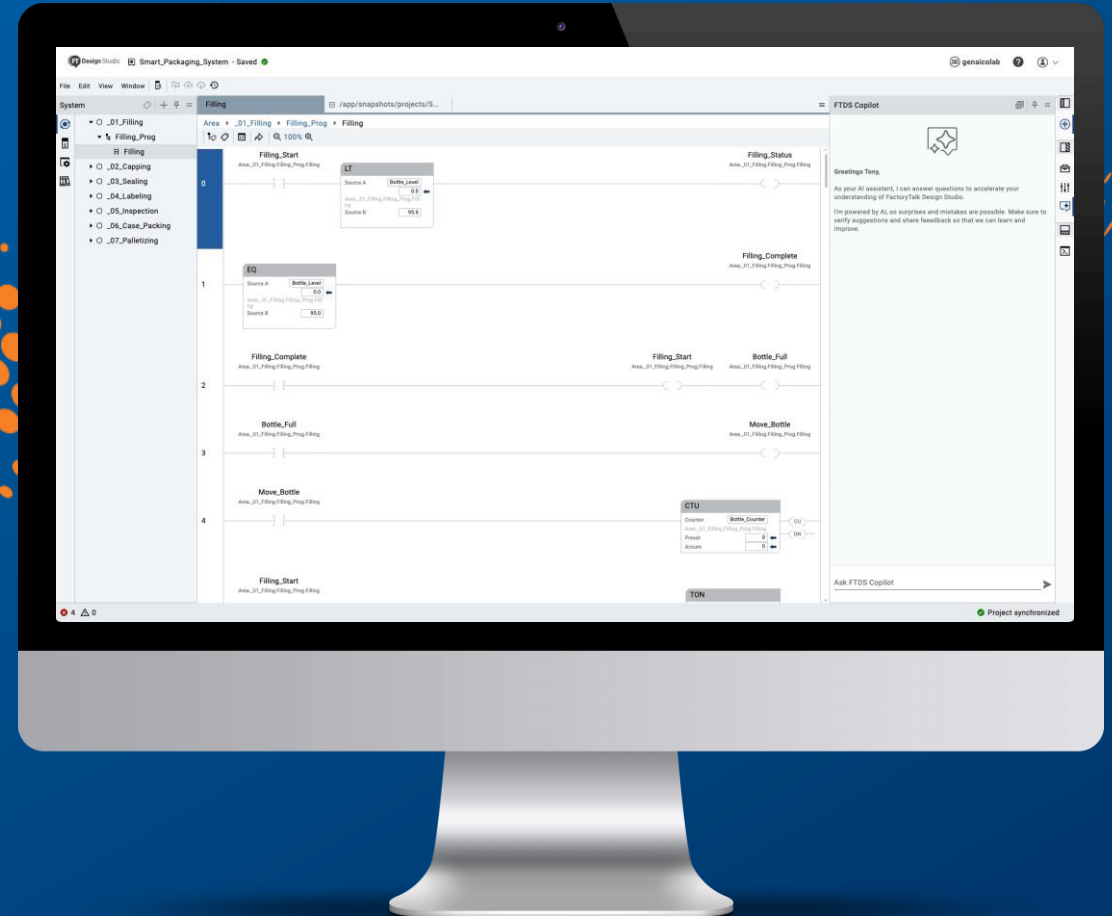
- Home page [HERE](#)
- Get started [HERE](#)

## FACTORYTALK DESIGN STUDIO

- Learn more [HERE](#)
- Get an entitlement [HERE](#)
- Get a NO COST 90-day trial [HERE](#)
- Webinar showcasing FTDS 2.0 capabilities [HERE](#)
- Setup cloud-to-controller communications [HERE](#)
- View the ordering guide [HERE](#)
- Internal content that can be shared with customers:
  - View commercial presentation [HERE](#)
  - Get started [HERE](#)

## FACTORYTALK REMOTE ACCESS

- Learn more [HERE](#)
- View the ordering guide [HERE](#)
- Get an entitlement [HERE](#)
- View webinar [HERE](#)
- Setup cloud-to-controller communications [HERE](#)



# Thank You



[www.rockwellautomation.com](http://www.rockwellautomation.com)



**Rockwell  
Automation**

expanding human possibility®