PlantPAx Control Strategies

Allen-Bradley
by ROCKWELL AUTOMATION

Reference Manual

Original Instructions

Important User Information

Read this document and the documents listed in the additional resources section about installation, configuration, and operation of this equipment before you install, configure, operate, or maintain this product. Users are required to familiarize themselves with installation and wiring instructions in addition to requirements of all applicable codes, laws, and standards.

Activities including installation, adjustments, putting into service, use, assembly, disassembly, and maintenance are required to be carried out by suitably trained personnel in accordance with applicable code of practice.

If this equipment is used in a manner not specified by the manufacturer, the protection provided by the equipment may be impaired.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Throughout this manual, when necessary, we use notes to make you aware of safety considerations.

WARNING: Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.

ATTENTION: Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you identify a hazard, avoid a hazard, and recognize the consequence.

IMPORTANT Identifies information that is critical for successful application and understanding of the product.

These labels may also be on or inside the equipment to provide specific precautions.

SHOCK HAZARD: Labels may be on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.

BURN HAZARD: Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.

ARC FLASH HAZARD: Labels may be on or inside the equipment, for example, a motor control center, to alert people to potential Arc Flash. Arc Flash will cause severe injury or death. Wear proper Personal Protective Equipment (PPE). Follow ALL Regulatory requirements for safe work practices and for Personal Protective Equipment (PPE).

The following icon may appear in the text of this document.

Identifies information that is useful and can help to make a process easier to do or easier to understand.

Table of Contents

Internal Model Control (IMC) Control Strategy

Chapter 5

Modular Multivariable Control (MMC) Control Strategy

Chapter 6

Process Analog Input (PAI) Control Strategies

Chapter 7

Process Analog Dual Sensor Input (PAID) Control Strategies

Chapter 8

Process Analog Input Multi Sensor (PAIM) Control Strategies

Chapter 9

Process Analog Output (PAO) Control Strategies

Chapter 10

Process Boolean Logic (PBL) Control Strategy

Chapter 11

Process Discrete 2-, 3-, or 4-State Device (PD4SD) Control Strategy

Chapter 12

Process Deadband Controller (PDBC) Control Strategy

Chapter 13

Process Discrete Input (PDI) Control Strategy

Chapter 14

Chapter 15

Process Discrete Output (PDO) Control Strategies

Chapter 16

Process Dosing Flow Meter (PDOSEFM) Control Strategy

Chapter 17

Process Dosing Weigh Scale (PDOSEWS) Control Strategy

Chapter 18

Process Analog Fanout (PFO) Control Strategies

Chapter 19

Process High or Low Selector (PHLS) Control Strategies

Process Lead Lag Standby (PLLS) Control Strategy

Chapter 20

Process Motor (PMTR) Control Strategies

Chapter 21

Process n-Position (PNPOS) Control Strategy

Chapter 22

Process Proportional + Integral + Derivative (PPID) Basic Control Strategies

Chapter 23

Process Proportional + Integral + Derivative (PPID) with Cascade Control Strategies

Chapter 24

Process Proportional + Integral + Derivative (PPID) Feedforward Control Strategies

Chapter 25

Process Proportional + Integral + Derivative (PPID) Override Control Strategies

Chapter 26

Process Proportional + Integral + Derivative (PPID) Basic PPID with Process Analog Output (PAO) Control Strategies

Chapter 27

Process Proportional + Integral + Derivative (PPID) Ratio Control Strategies

Chapter 28

Chapter 29

Process Proportional + Integral + Derivative (PPID) Split Range Control Strategies

Chapter 30

Process Pressure/Temperature Compensated Flow (PPTC) Control Strategies

Chapter 31

Process Tank Strapping Table (PTST) Control Strategies

Chapter 32

Process Valve Hand Operated (PVLVHO) Control Strategy

Chapter 33

Process Valve Motor Operated (PVLVMO) Control Strategy

Chapter 34

Process Mix Proof Valve (PVLVMP) Control Strategy

Process Valve Solenoid Operated (PVLVSO) Control Strategy

Chapter 35

Process Variable Speed Drive (PVSD) Control Strategies

Chapter 36

Ramp/Soak (RMPS) Control Strategy

Chapter 37

Totalizer (TOT) Control Strategy

Chapter 38

Process Area Control Strategy

Chapter 39

Generic Equipment Module (EMGEN) Control Strategy

Chapter 40

Generic Equipment Phase (EPGEN) Control Strategy

Chapter 41

Process Unit Control Strategy

Chapter 42

About This Publication

The Rockwell Automation® Process Objects Library includes PlantPAx® control strategies to help reuse logic to save development time. Use this reference manual for guidance on when and how to use each control strategy.

Inclusive Terminology

Rockwell Automation recognizes that some of the terms that are currently used in our industry and in this publication are not in alignment with the movement toward inclusive language in technology. We are proactively collaborating with industry peers to find alternatives to such terms and making changes to our products and content. Please excuse the use of such terms in our content while we implement these changes.

Download Firmware, AOP, EDS, and Other Files

Download firmware, associated files (such as AOP, EDS, and DTM), and access product release notes from the Product Compatibility and Download Center at rok.auto/pcdc.

Additional Resources

These documents contain additional information concerning related products from Rockwell Automation. You can view or download publications at rok.auto/literature.

Resource	Description
PlantPAx Distributed Control System Configuration and Implementation User Manual, publication PROCES-UM100	Provides system guidelines and instructions to assist with the development of your PlantPAx system.
Rockwell Automation Library of Process Objects: HART Modules for PlantPAx DCS, publication PROCES-RM010	Provides details on the integration of HART devices into a PlantPAx system or Integrated Architecture®
Rockwell Automation Library of Process Objects, publication PROCES-RM200	Describes the Add-On Instructions, PlantPAx instructions, and associated faceplates that are available to develop applications.
Rockwell Automation Sequencer Object User Manual, publication PROCES-RM202	Provides an overview of how to use the Rockwell Automation® Sequencer Object (raP_Opr_Seq).
Power Device Library Reference Manual, publication DEVICE-RM100	Provides information on objects for discrete, velocity, motion, and PowerMonitor™ devices.
I/O Device Library Reference Manual, publication DEVICE-RM200	Provides information on objects for Rockwell Automation 1756, 1769, 1734, 1794, 1738, 1732E, 1719, 5069, 5094 I/O modules, including pre-configured status and diagnostic faceplates.
Advanced Process Control and Drives and Phase and Sequence Instruction Manual, publication 1756-RM006	Provides details about the available General, Motion, Process, and Drives instruction set for a Logix-based controller.

Notes:

Chapter 1

PlantPAx Control Strategies

The PlantPAx® control strategies are routines or programs that you import into your controller project. The PlantPAx control strategies are Function Block Diagrams or Ladder Diagrams that include pre-configured process instructions that represent common control and equipment scenarios in process automation. The PlantPAx control strategies have several preconfigured arrays and tags.

See the instruction online help for complete details on the instructions in the control strategies.

Library Prerequisites

On the Product Compatibility and Download Center (PCDC), download the latest versions of these libraries.

- Power Device Library
- I/O Device Library

How to use PlantPAx Control Strategies

You can import the PlantPAx control strategies into your project using Studio 5000 Logix Designer® or using Application Code Manager (ACM) plug-ins within Studio 5000 Logix Designer.

Import Method	Considerations
Import using Studio 5000 Logix Designer	• You can easily modify a source import file for each application: a. Import the standard routine. b. Modify the routine. c. Export the modified routine to a renamed control strategy for your application. • You must import individual routines one at a time (even when a single control strategy is comprised of multiple routines). • You **can** add routines while online with the controller. For more information, see Import with Studio 5000 Logix Designer on page 18.
Import using ACM plug-ins in Studio 5000 Logix Designer	• ACM process library includes a comprehensive set of PlantPAx control strategies plug-ins. • You can enter multiple control strategies at once (even when there are multiple routines per control strategy) • You can configure faceplate navigation at import. • It is difficult to modify source routines. • You **cannot** use the plug-in feature while Online with the controller. For more information, see Import Using Application Code Manager Plug-ins on page 20.

Import with Studio 5000 Logix Designer

The PlantPAx control strategies are provided as folders that contain one or more routines that can be imported into an appropriate program.

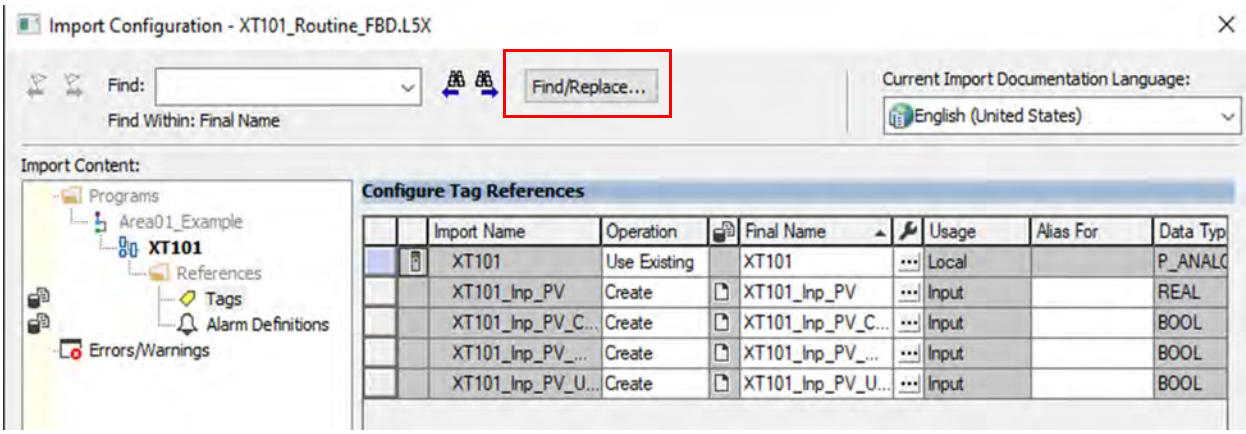
For information on how to import routines and programs, see the Logix 5000® Controllers Import/Export Project Components Programming Manual, publication 1756-PM019.

When the Import Configuration window opens:

1. Select the Tags folder.

   All tags in the control strategy have a default prefix, such as XT101.

2. Use the Find/Replace button to rename the prefix to match your site's tag naming convention.

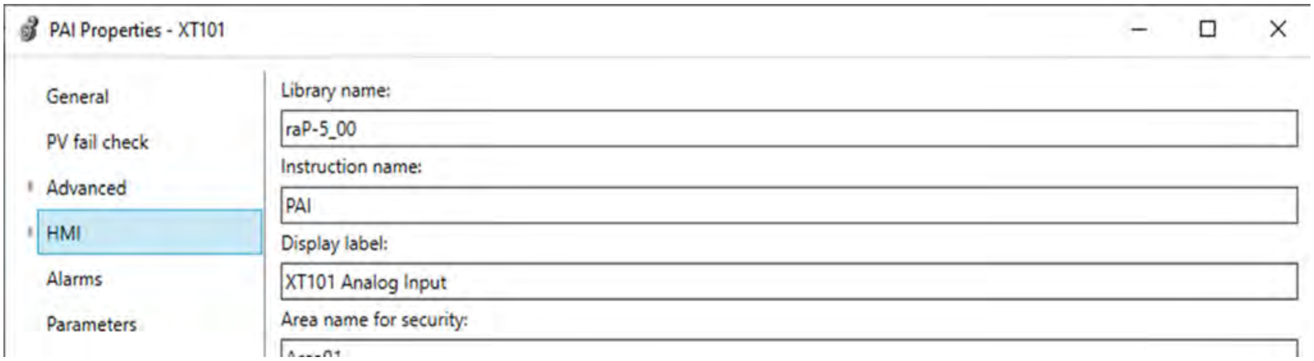


To configure the appropriate display label in the HMI:

1. Select the Properties dialog box for the instruction in the control strategy.

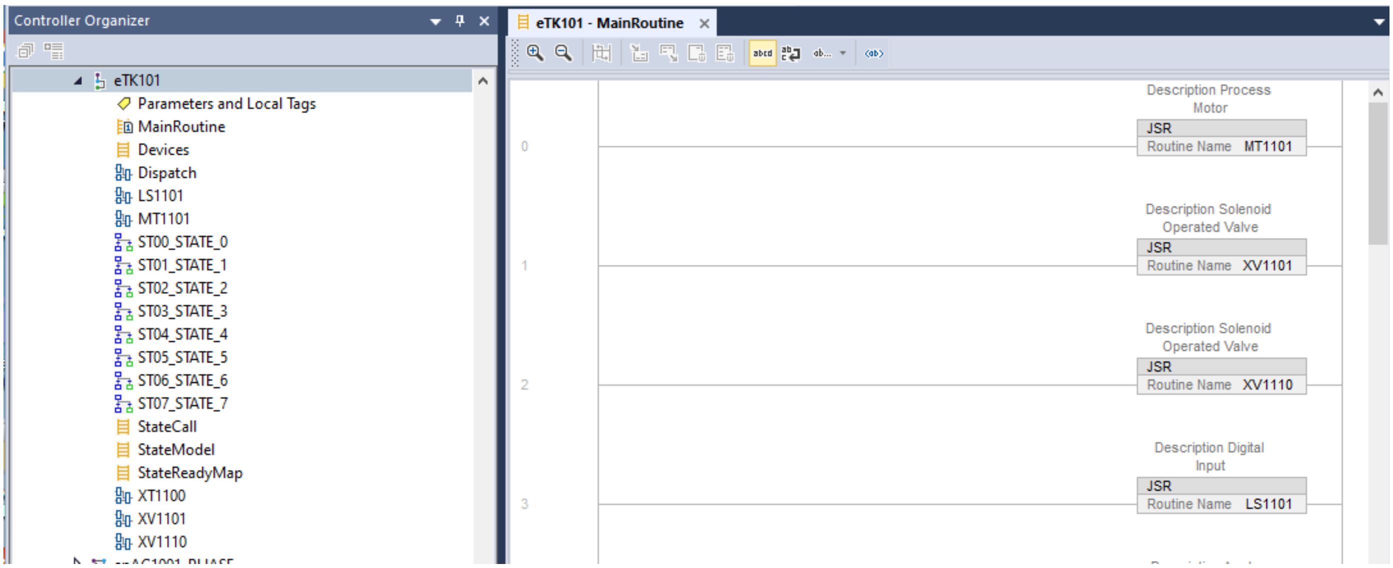
2. Open the HMI page.

3. Edit the default display label to provide an appropriate label for the operator interface components.
   - As a best practice, use a consistent labeling methodology throughout all projects that are part of the system. You could use the exact tag name, or use a more readable format.
   - If you do not use the exact tag name, then the display label should generally align with ANSI/ISA-5.1-2022 naming standards.



Add Main Routine Code to Execute the Imported Control Strategies

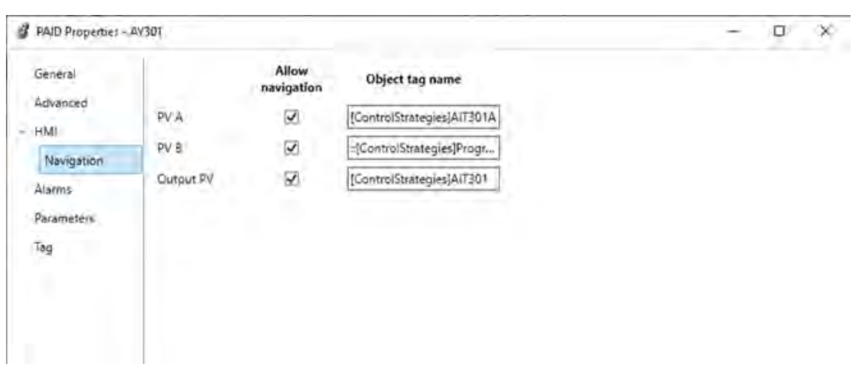
Add JSR instructions that reference the imported control strategy to the Main Routine to execute the new control strategy routines. For example:



HMI Navigation

The process instructions in the PlantPAx control strategies support HMI navigation to other instructions in the same control strategy. To leverage this capability, you only need to specify specify the appropriate controller-scoped or program-scoped tag.

On the process instruction, select Properties > HMI> Navigation and enter the tags for the control strategy objects that you want to allow navigation to.



This example shows both controller-scoped and program-scoped tags, but you can use either for each option. The syntax for tag type is as follows:

Tag Type	Syntax
Controller scope	[TOPIC]TagName For example: [ControlStrategies]AIT301A
Program scope	::[TOPIC]Program:ProgramName.TagName For example: ::[ControlStrategies]Program:CS_PAID.AIT301B

Import Using Application Code Manager Plug-ins

The Application Code Manager (ACM) process library includes a comprehensive set of PlantPAx control strategies plug-ins for you to use in your controller projects. Follow your project plan (the spreadsheet with your devices and tags) as you add PlantPAx control strategies for devices (motors, valves, drives, and so forth) to the Studio 5000 Logix Designer application project file.

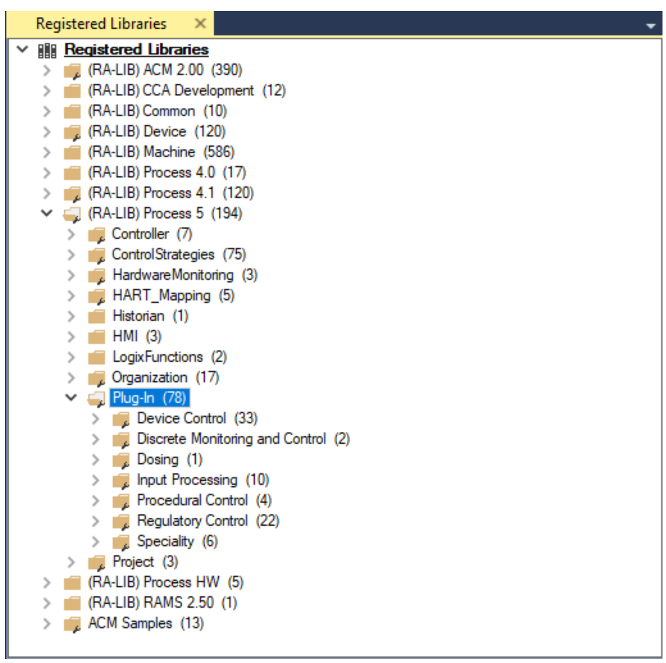
For information on Application Code Manager, see Application Code Manager User Manual, publication LOGIX-UM003.

IMPORTANT	• You can use ACM and ACM plug-ins to add PlantPAx control strategies only when you are **offline** with the controller. • The Library Object Import Wizard can import one or more control strategies at a time. • When adding multiple PlantPAx control strategies of the same type, rename each instance to a unique name.

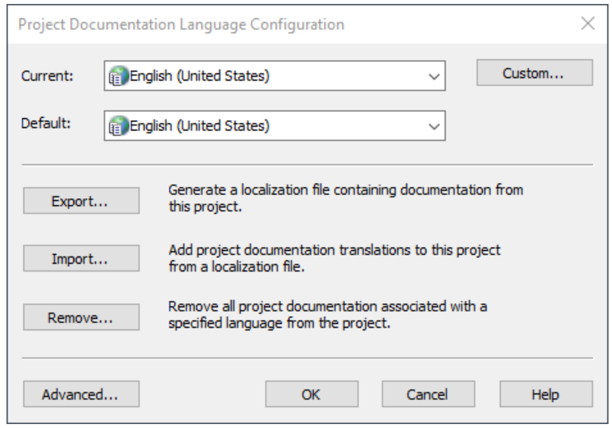
Prerequisites

Before you can use plug-ins with the Import Library Objects in Studio 5000 Logix Designer, you must do the following:

1. Make sure Application Code Manager is installed on the workstation that has Studio 5000 Logix Designer.
2. Make sure the Application Code Manager Process Library is registered in ACM.



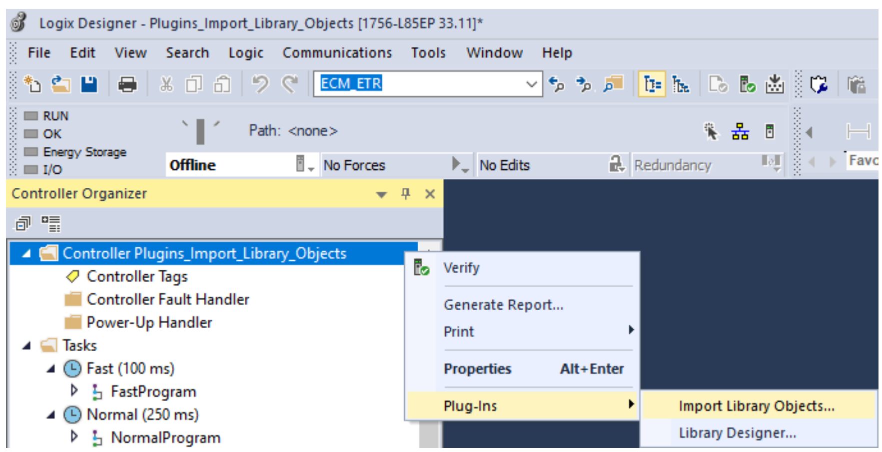
3. In Studio 5000 Logix Designer, go to Tools > Documentation Languages. Make sure the Project Documentation Language Configuration Default is set to English (United States).



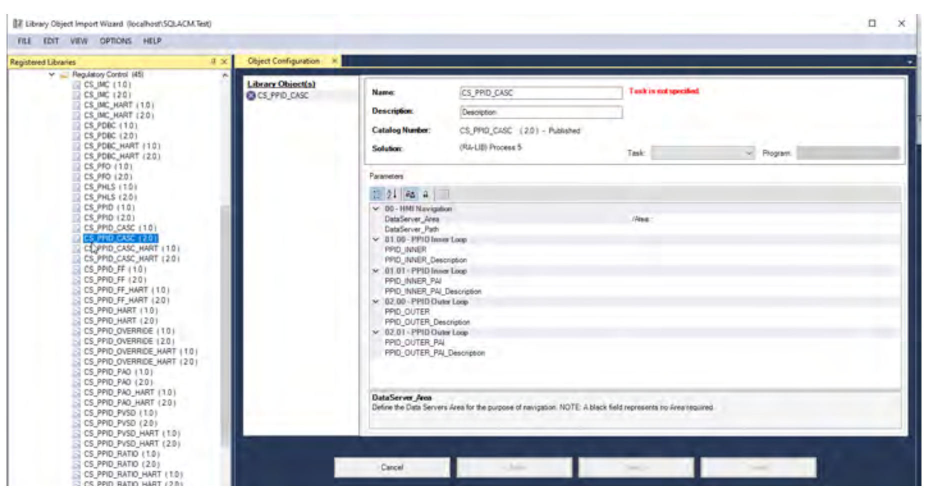
Import Library Objects (Offline Only)

This example workflow shows how to use the Library Object Import Wizard to add two PlantPAx control strategies (CS_PPID_CASC and CS_PVLVSO) into the Logix Designer Project (ACD) file.

1. Open a Studio 5000 Logix Designer Project (.ACD) file.
2. Right-click on the Controller, and navigate to Plug-Ins > Import Library Objects… to launch the Library Object Import Wizard.



3. In the Library Object Import Wizard, expand the tree to (RA-LIB) Process 5 > Plug-In > Regulatory Control.
4. Double-click CS_PPID_CASC to add to Library Object(s).



5. Under Library Object(s), click CS_PPID_CASC and configure the Task and Program. Configure the parameters required for the CS_PPID_CASC control strategies as indicated in Table 1.

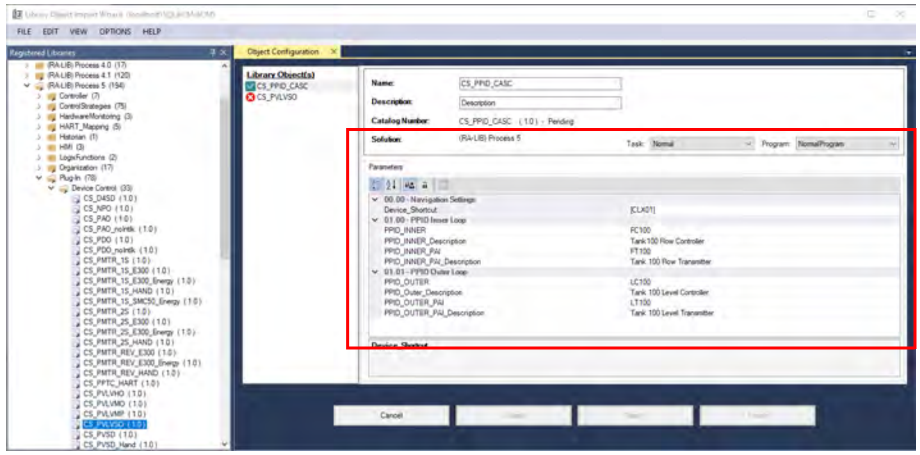
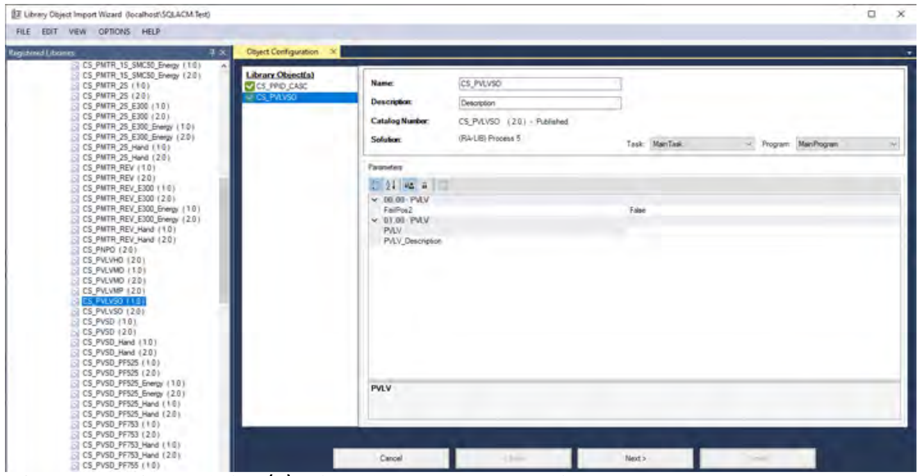


Table 1 - CS_PPID_CASC Parameters

00.00 – Navigation Settings		
Device_Shortcut	[CLX01]	Example - [TOPIC] or /Area/DATA::[TOPIC]
01.00 - PPID Inner Loop		
PPID_INNER	FC100	Inner Loop Controller Tag
PPID_INNER_Description	Tank 100 Flow Controller	Inner Loop Controller Tag Description
PPID_INNER_PAI	FT100	Inner Loop Analog Input Tag
PPID_INNER_PAI_Description	Tank 100 Flow Transmitter	Inner Loop Analog Input Tag Description
01.01 – PPID Outer Loop		
PPID_OUTER	LC100	Outer Loop Controller Tag
PPID_OUTER_Description	Tank 100 Level Controller	Outer Loop Controller Tag Description
PPID_OUTER_PAI	LT100	Outer Loop Analog Input Tag
PPID_OUTER_PAI_Description	Tank 100 Level Transmitter	Outer Loop Analog Input Tag Description

6. In the Library Object Import Wizard, expand the tree to (RA-LIB) Process 5 > Plug-In > Device Control.
7. Double-click CS_PVLVSO to add to Library Object(s).



8. Under Library Object(s), click CS_PVLVSO and configure the Task and Program. Configure the parameters that are required for the CS_PVLVSO control strategies as indicated in Table 2.

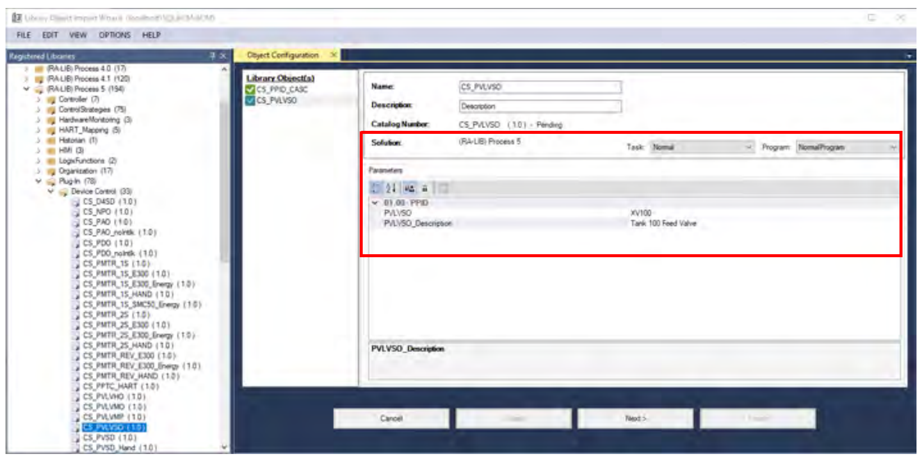
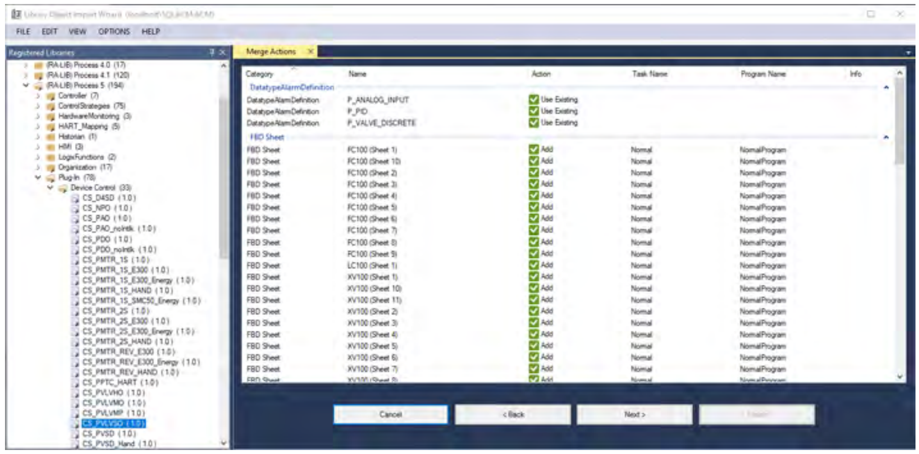


Table 2 - CS_PVLVSO Parameters

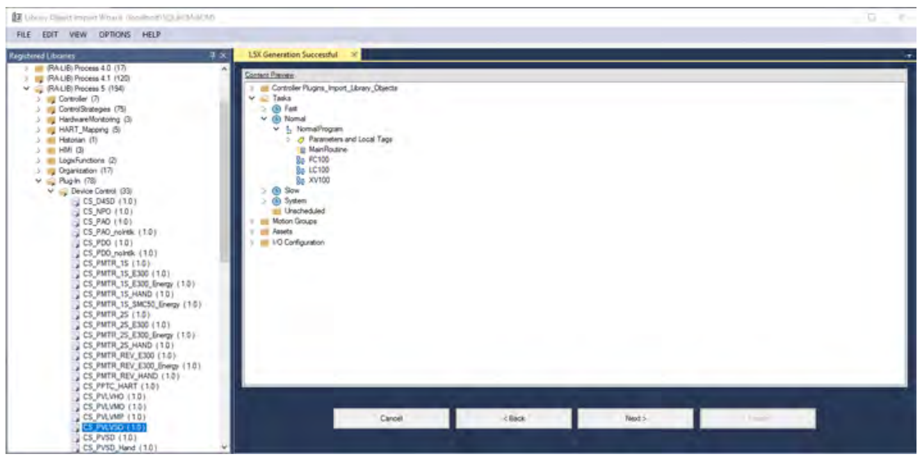
01.00 - PVLV		
PVLVSO	XV100	Valve Tag
PVLVSO_Description	Tank 100 Feed Valve	Valve Tag Description

9. In the Library Object Import Wizard, click Next.

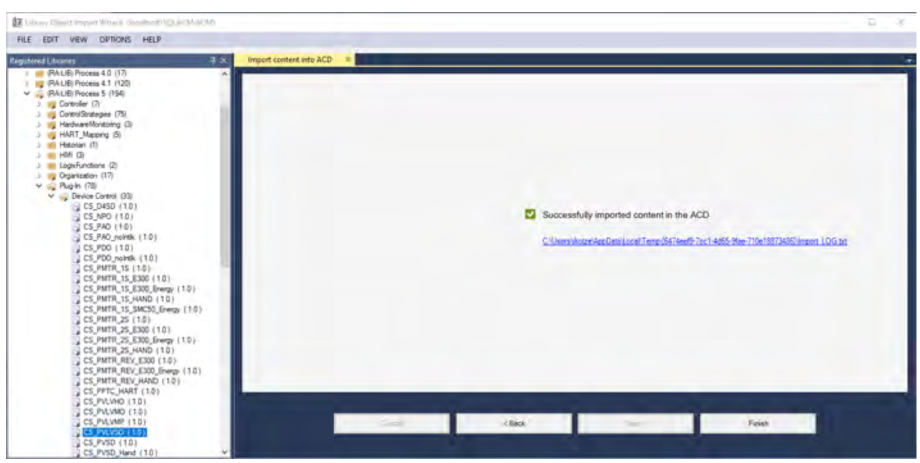
10. Review the Merge Actions screen, and click Next.



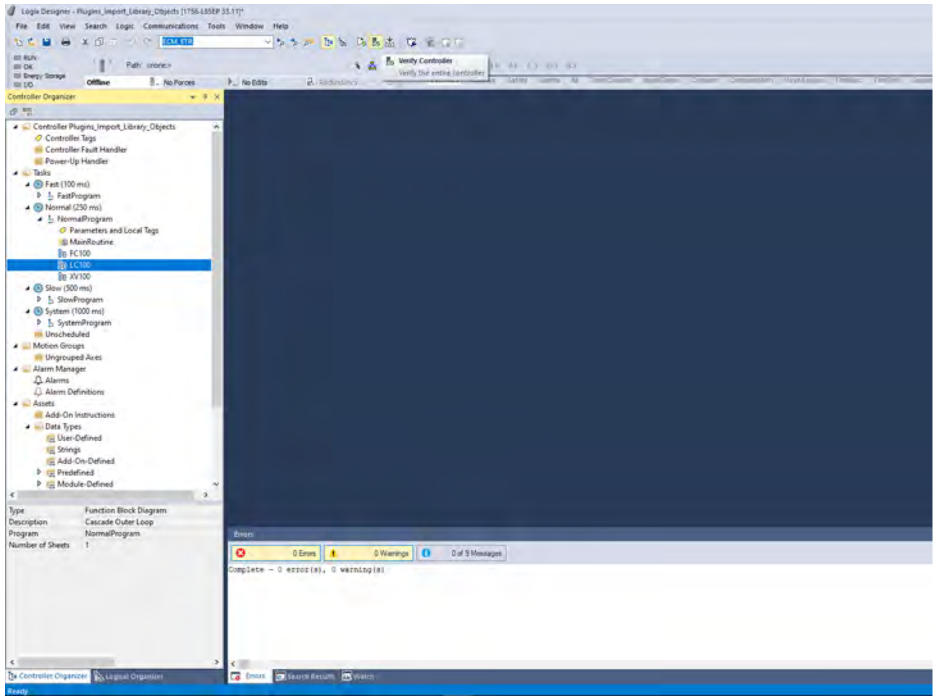
11. Review the L5X Generation Successful screen (expand tasks and programs), and click Next.



12. Review the Import content into ACD screen. Confirm that the content was imported successfully. Click Finish.



13. In Studio 5000 Logix Designer, use the Verify Controller feature to confirm that the control strategies were added to the Logix Designer Project without creating additional errors.



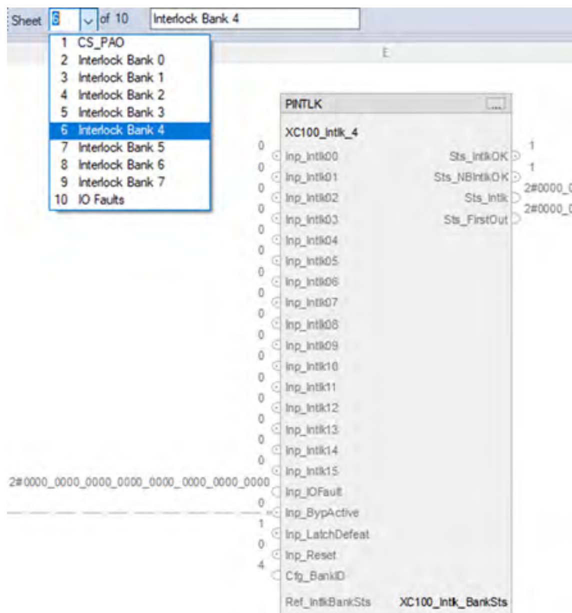
Interlock Options

The Process Interlocks (PINTLK) instruction prevents equipment from starting or being energized. Interlocks are always evaluated to de-energize equipment. For permissive conditions that must be made true to start the equipment, but are ignored once the equipment is running, use the Process Permissive (PPERM) instruction.

In each PlantPAx control strategy that has interlocks, there are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default.

If your project runs into memory constraints, you can remove unused code, such as unused interlock banks. Remove the interlock banks in order of the last to the first bank.

If you edit or add interlock sheets, make sure the PINTLK Cfg_BankID value matches the number of the interlock bank sheet. For example, Interlock Bank Sheet 4 has a PINTLK instruction where the Cfg_BankID is also 4.



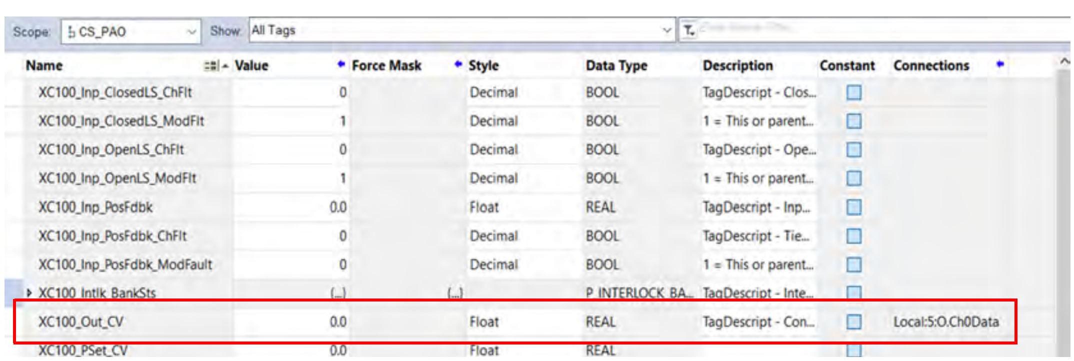
ACM creates the interlock banks sheets that you need based on your settings when you create your application.

For more information, see the online help for the PINTLK instruction.

I/O Connections

The PlantPAx control strategies have preconfigured program connections for the input and output values for the process instruction in the control strategies. These input and output values are program-scoped tags in the Parameters and Local Tags for the control strategy (not controller-scoped tags).

For example, in the PAO control strategy, the output CV is a program connection to a channel on the module.



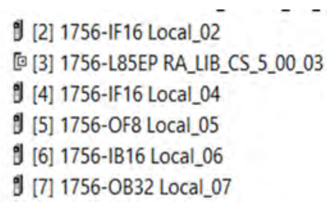
Scope: CS_PAO Show: All Tags

Name	Value	Force Mask	Style	Data Type	Description	Constant	Connections
XC100_Inp_ClosedLS_ChFlt	0		Decimal	BOOL	TagDescript - Clos...		
XC100_Inp_ClosedLS_ModFlt	1		Decimal	BOOL	1 = This or parent...		
XC100_Inp_OpenLS_ChFlt	0		Decimal	BOOL	TagDescript - Ope...		
XC100_Inp_OpenLS_ModFlt	1		Decimal	BOOL	1 = This or parent...		
XC100_Inp_PosFdbk	0.0		Float	REAL	TagDescript - Inp...		
XC100_Inp_PosFdbk_ChFlt	0		Decimal	BOOL	TagDescript - Tie...		
XC100_Inp_PosFdbk_ModFault	0		Decimal	BOOL	1 = This or parent...		
XC100_Intlk_BankSts	{...}	{...}		P_INTERLOCK_BA...	TagDescript - Inte...		
XC100_Out_CV	0.0		Float	REAL	TagDescript - Con...		Local:5:O.Ch0Data
XC100_PSet_CV	0.0		Float	REAL			

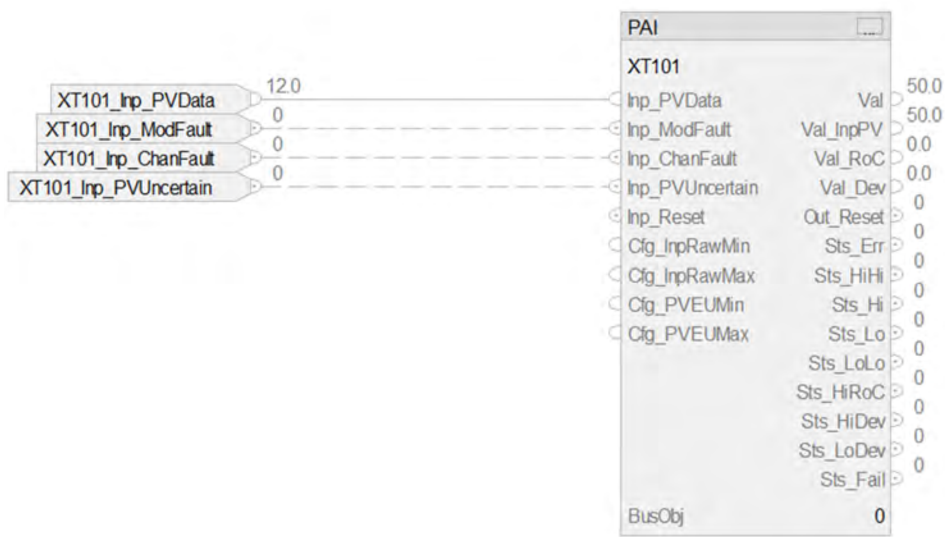
Map Device Tags to Input Data

In each PlantPAx control strategy, inputs to the main instructions are preconfigured to map to similar locations for input modules.

For example, a PAI control strategy for this 1756-IF16 analog input module in slot 2:



Has this logic:



And the inputs map as follows:

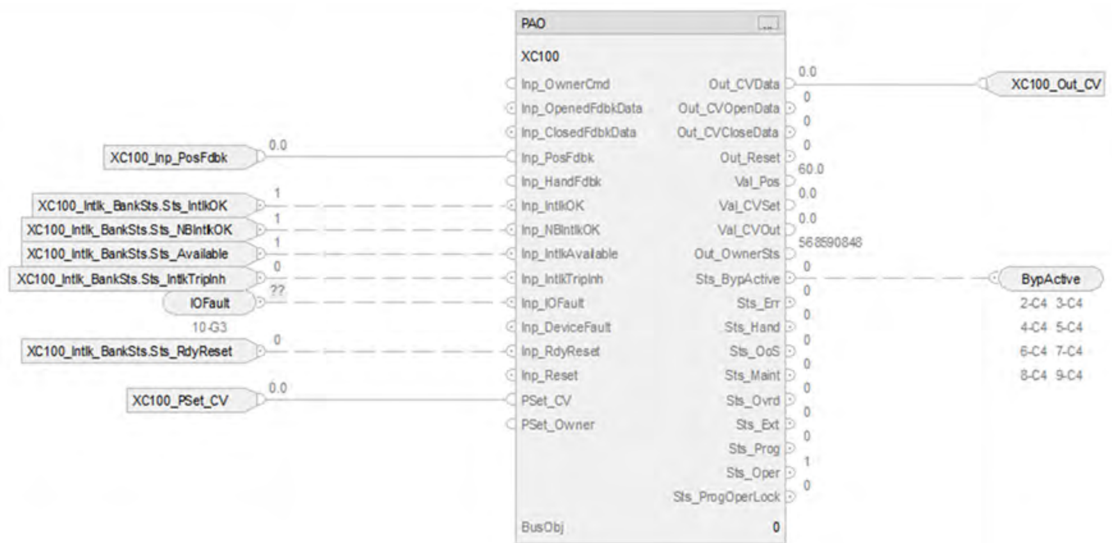
Input	Description
XT101_Inp_PVData	Process variable input (program-scoped tag) Source: sensor or input Program connection to Local:2:I.Ch0Data
XT101_Inp_ModFault	Controller-scoped tag Local_02.Sts_IOFault output from raP_Dvc_LgxModuleSts block for Local_02
XT101_Inp_ChanFault	Controller-scoped tag Local:2:I.CH0Fault directly from AB:1756_IF16_Float_No_Alm:I:0 module tag
XT101_Inp_PVUncertain	Controller-scoped tag Local_02.Sts_AnyChanUncertain output from raP_Dvc_LgxModuleSts block for Local_02

Map Tags to Output Data

In each PlantPAx control strategy, the output from the main instructions is preconfigured to map to similar locations for output modules.

For example, a PAO control strategy for this 1756-OF8 analog output module:

Has this logic:



And the output maps as follows:

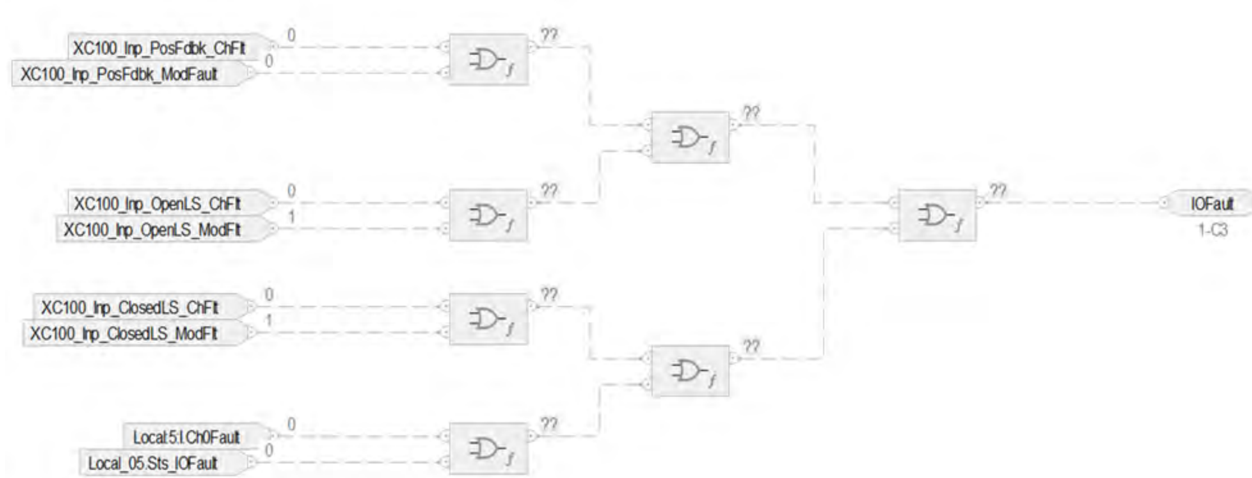
Input	Description
XC100_Out_CV	Control variable output (program-scoped tag) Program connection to Local:5:0.Ch0Data

Map I/O Faults

Fault data for output modules all wire to the IOFault reference on the associated I/O Fault sheet in the PlantPAx control strategy.

For example, a PAO control strategy for this 1756-OF8 analog input module:

Has this logic:



Input	Device Input Tag
XC100_Inp_PosFdbk_ChFault	Program-scoped tag
XC100_Inp_PosFdbk_ModFault	Program-scoped tag
XC100_Inp_OpenLS_ChFlt	Program-scoped tag
XC100_Inp_OpenLS_ModFlt	Program-scoped tag
XC100_Inp_ClosedLS_ChFlt	Program-scoped tag
XC100_Inp_ClosedLS_ModFlt	Program-scoped tag
Local5:I.Ch0Fault	Controller-scoped tag directly from Local:5:I.CH0Fault within AB:1756_OF8_Float:I:0 module tag
Local_05.Sts_IOFault	Controller-scoped tag Local_05.Sts.IOFault From raP_Dvc_LgxModuleSts block for Local_05

The program-scoped tags are preconfigured in the PlantPAx control strategy and must be mapped to the appropriate I/O points.

Notes:

HART Integration

HART Data

The PlantPAx® control strategies that use HART data use a Process Analog HART (PAH) instruction to provide input to a Process Analog Input (PAI) instruction. For more information, see the PAI Control Strategy: CS_PAI_HART Sheet on page 72.

The examples in this chapter use Application Code Manager (ACM) to enable more efficient project development with libraries of reusable code. Application Code Manager creates modular objects with customizable configuration parameters using the reusable content. Application Code Manager can also create the associated visualization, historical, and alarming elements for a project.

PAH Configuration Considerations

Operand	Type	Description
PlantPAx control	P_ANALOG_HART	• Instance of data structure (backing tag) required for proper operation of instruction.
Ref_HARTData	PAX_HART_DEVICE:I:0	• Required data type: HART data from the I/O module assembly. • Select the HART device in your Controller Organizer. The device must support the PAxDevice data type: IOTreeObject:I.PAxDevice
Ref_DiagTable	P_HART_CODE_DESC_STATUS[2]	• Lookup table for diagnostic bit number (to message and status). • Select the correct table for your HART device; see table below.
Ref_UnitsTable	RAC_CODE_DESCRIPTION[2]	• Lookup table for units of measure code (to units text). Select _HART_EUTable_Generic.

Fully Integrated HART with FLEX 5000 I/O

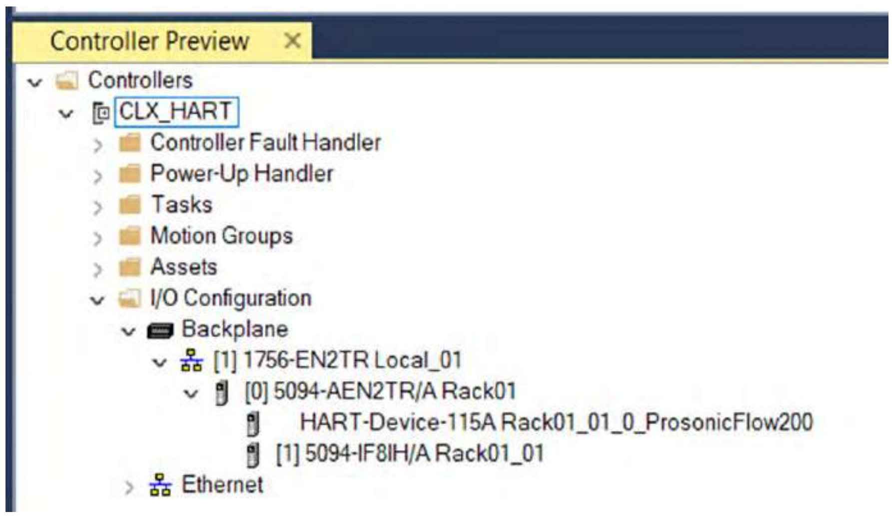
Highly-integrated HART uses a PlantPAx data type in the process controller for use with FLEX 5000® modules:

- Configuration of devices within the I/O Configuration tree (no Add-On Instruction needed)
- Device diagnostics automatically propagate to the controller project

Integrate FLEX 5000 I/O with HART Device Using PAH And PAI Instructions

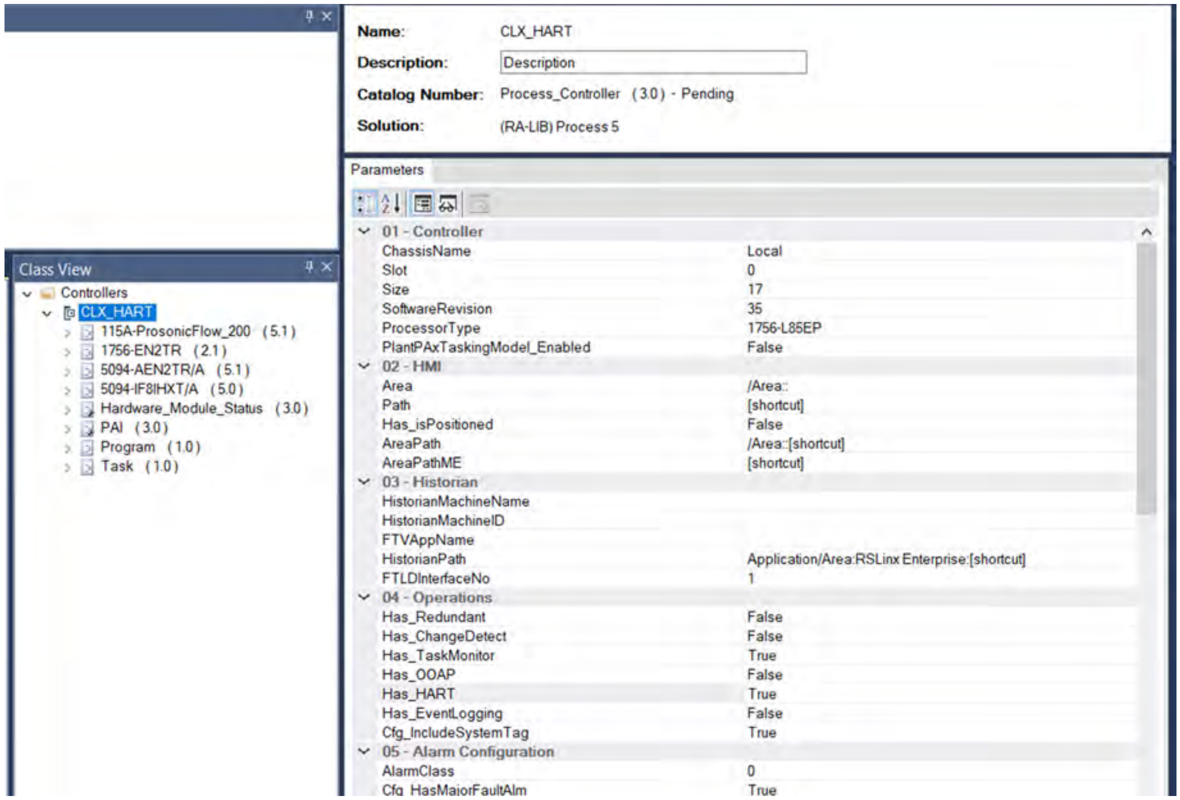
In this example, the ACM project contains:

- ControlLogix Process controller
- 1756-EN2TR communication module
- 5094-AEN2TR communication module for FLEX 5000 I/O connectivity
- 5094-IF8IH HART analog input module with an Endress+Hauser
- ProsonicFlow 200 instrument connected to Channel 0

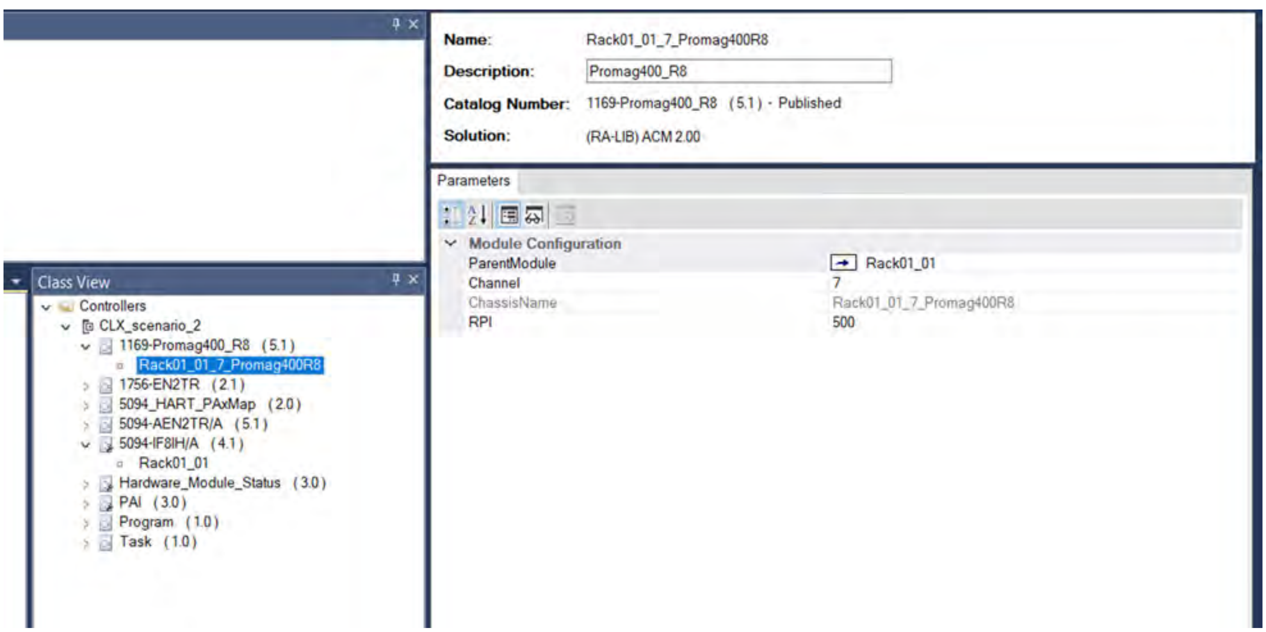


IMPORTANT	When you add multiple EtherNet/IP communication modules to an ACM project, remember to enter a unique IP address for each module.

1. Configure the process controller for parameters you need for your application, and make sure to set Has_HART.

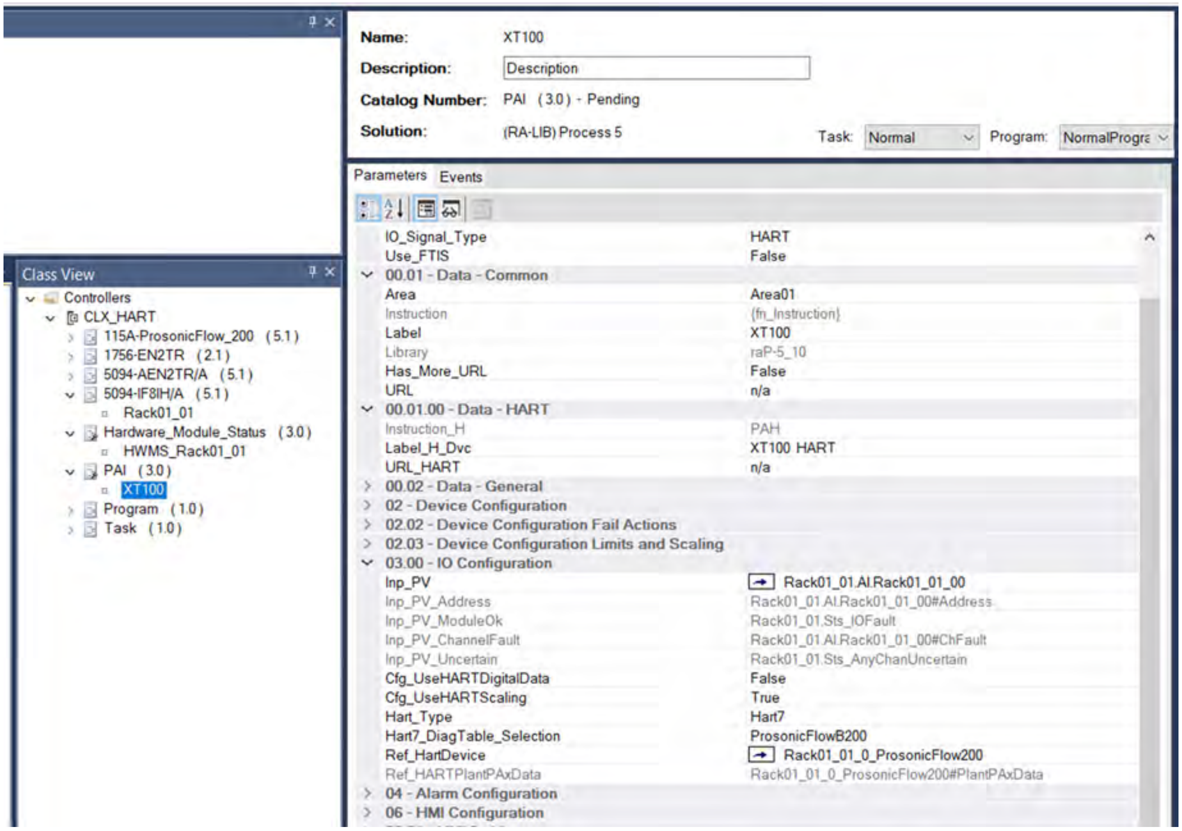


2. When you add the HART instrument, configure the ParentModule Parameter to the 1756-IF8IH module in Rack01.



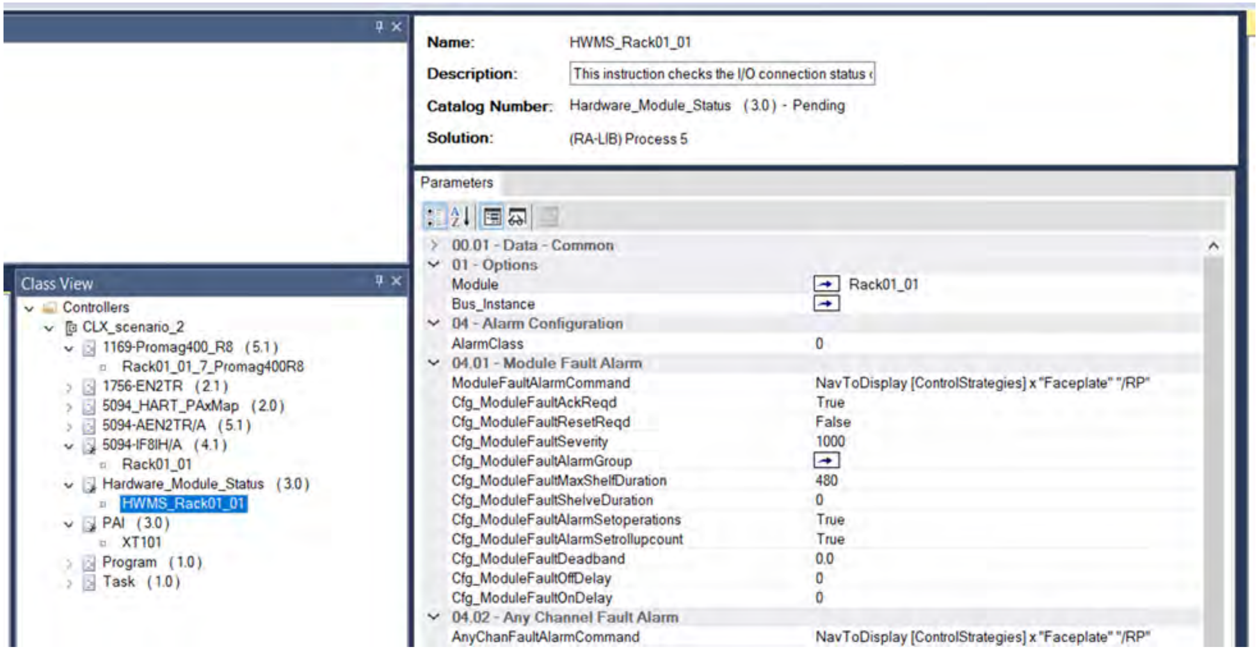
3. From the Process library > Control Strategies > Input Processing folder, add a PAI instance for the analog input module and configure these parameters in the I/O Configuration section.

IMPORTANT	You must create an individual PAI instance for each input module in your application.

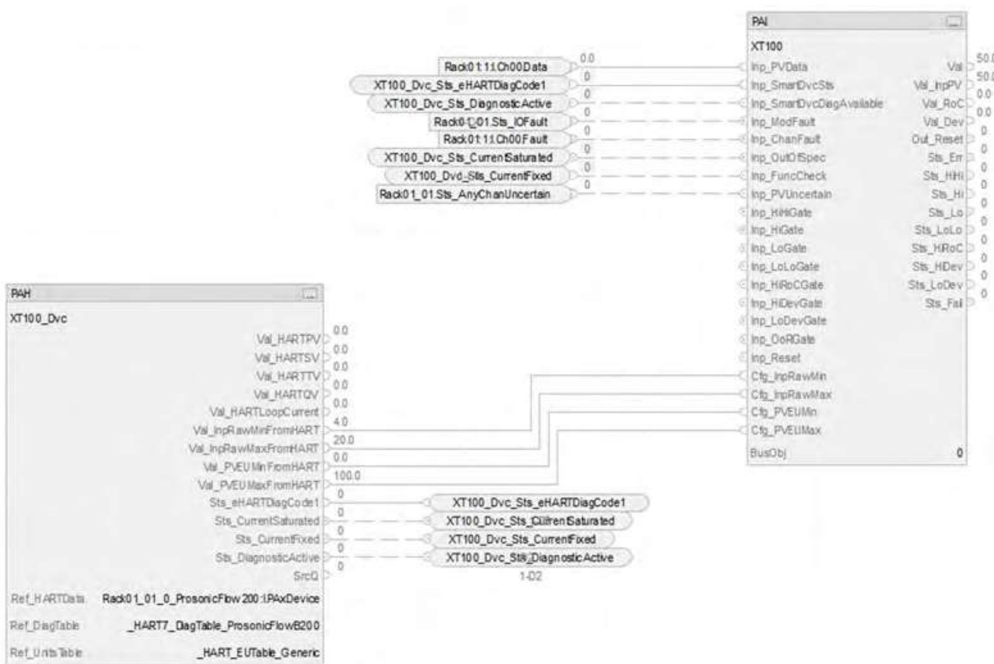


ACM Parameter	Usage
Task Program	Assign a Task and Program for the PAI control strategy.
IO_Signal_Type	HART
Inp_PV	Connect to the channel of the I/O module that the instrument is connected to.
Cfg_ UseHARTDigitalData	Not applicable, leave at default value.
Cfg_UseHARTScaling	Set to True if you want to connect the scaling parameters from the PAH module.
Hart_Type	Select the HART protocol revision (Generic, Hart, Hart5, Hart6 orHart7).
Hart7_DiagTable_Selection	Select the relevant Diag Table value for the instrument.
Ref_HartDevice	Connect to the instrument.

4. From the Process library > Hardware Monitoring > Specialty folder, add a Hardware_Module_Status object and configure the Module parameter for the 1756-IF8IH module.



5. Generate the controller ACD file.

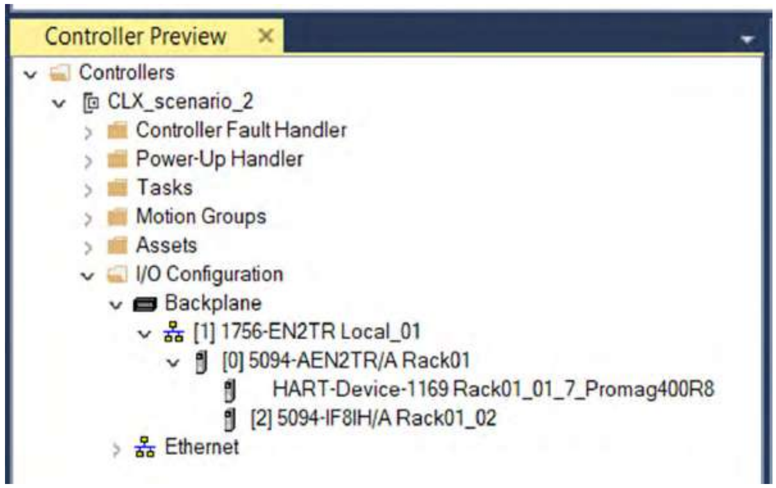


Integrate FLEX 5000 with HART Device via PV, SV, TV, or QV Values

In this example, the ACM project contains:

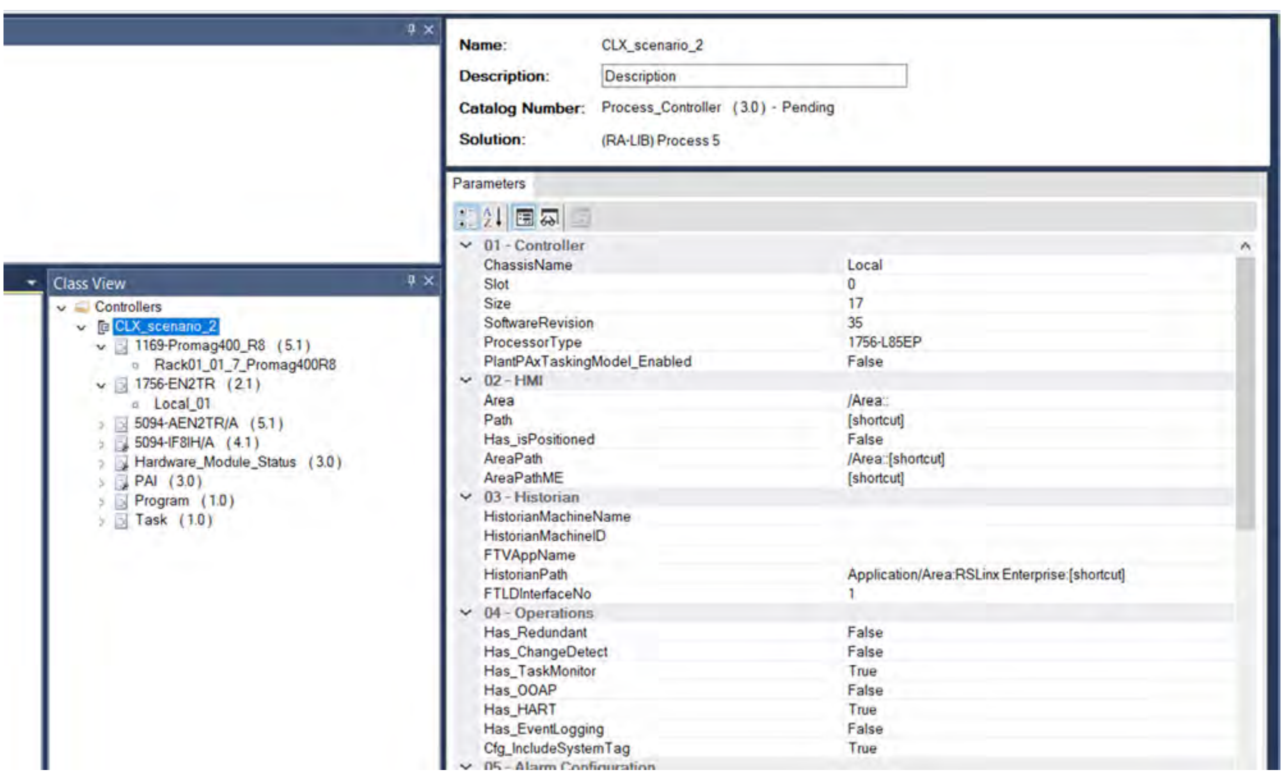
- ControlLogix Process controller
- 1756-EN2TR communication module
- 5094-AEN2TR communication module for FLEX 5000 I/O connectivity
- 5094-IF8IH HART analog input module with an Endress+Hauser
- Promag 400 revision 8 instrument connected to Channel 7

IMPORTANT	When you add multiple EtherNet/IP communication modules to an ACM project, remember to enter a unique IP address for each module.

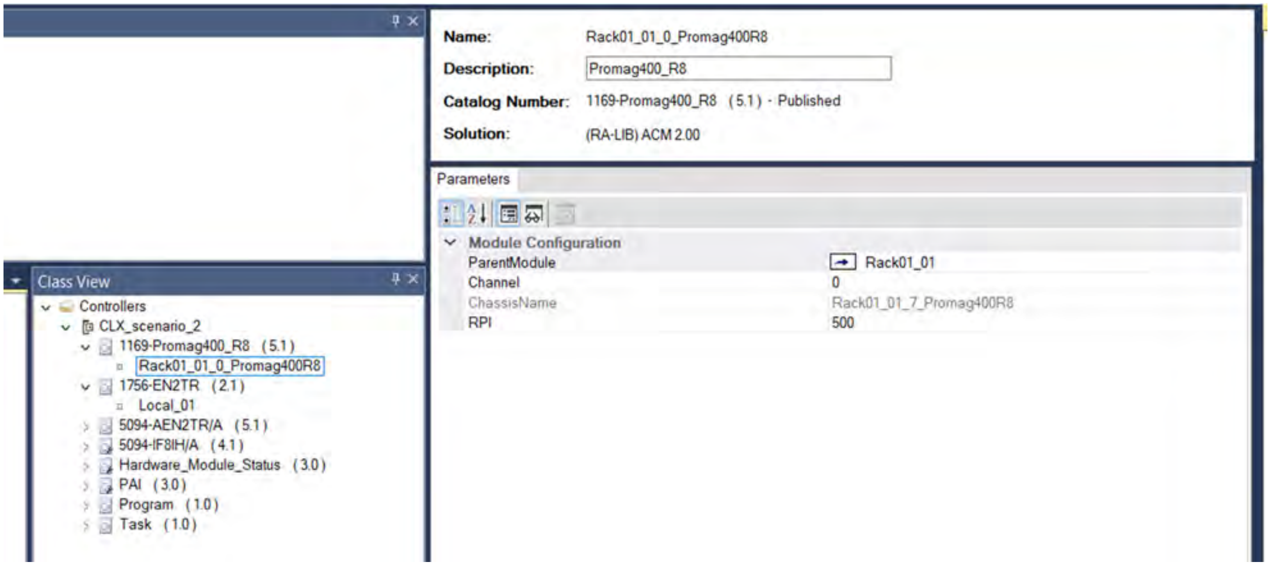


Add the devices to the ACM project and configure parameters as needed.

1. Configure the process controller for parameters you need for your application, and make sure to set Has_HART.

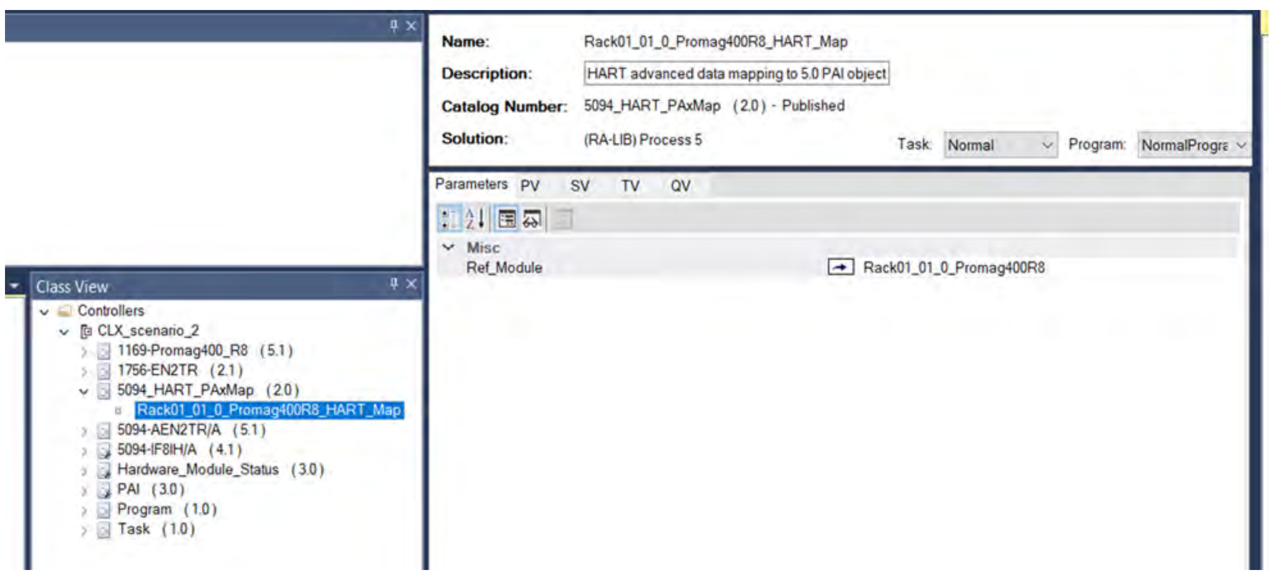


2. When you add the HART instrument, configure the ParentModule Parameter to the 1756-IF8IH module in Rack01.

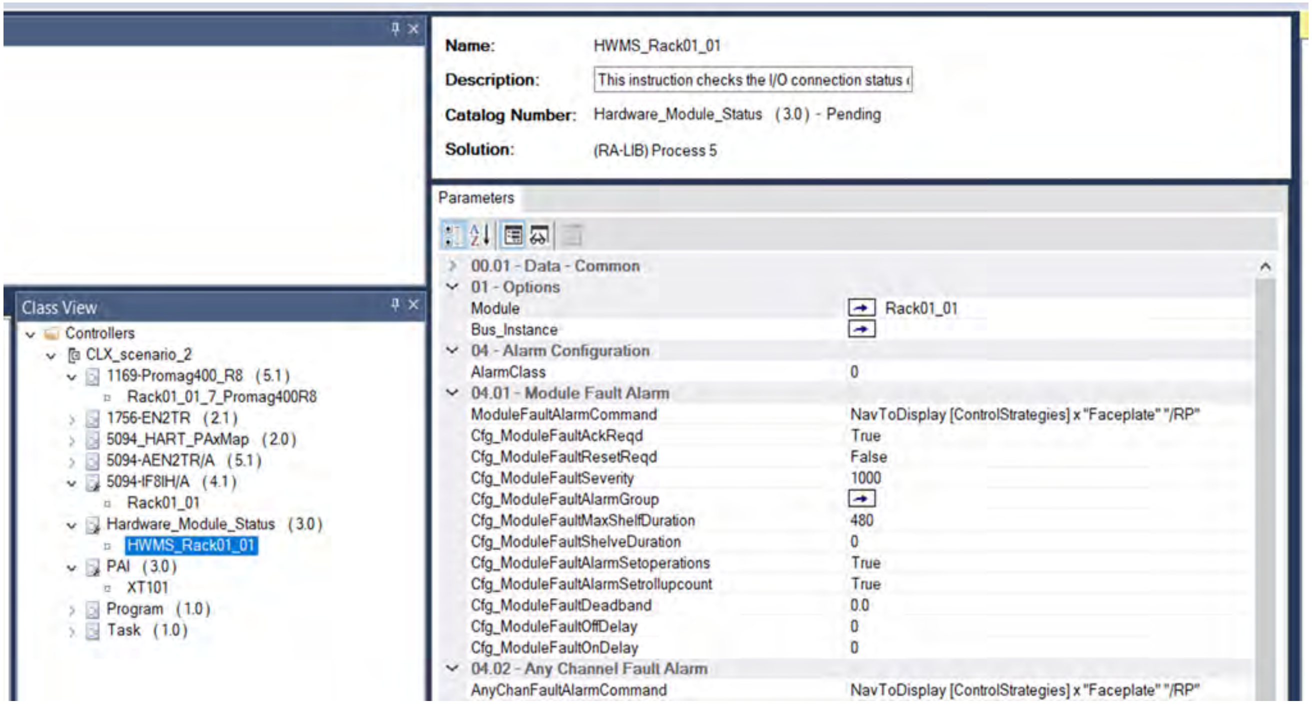


3. From the Process library > HART_Mapping > HART IO Card Mapping, create an instance of the 5094_HART_PAxMap and connect to the Promag 400 revision 8 instrument.

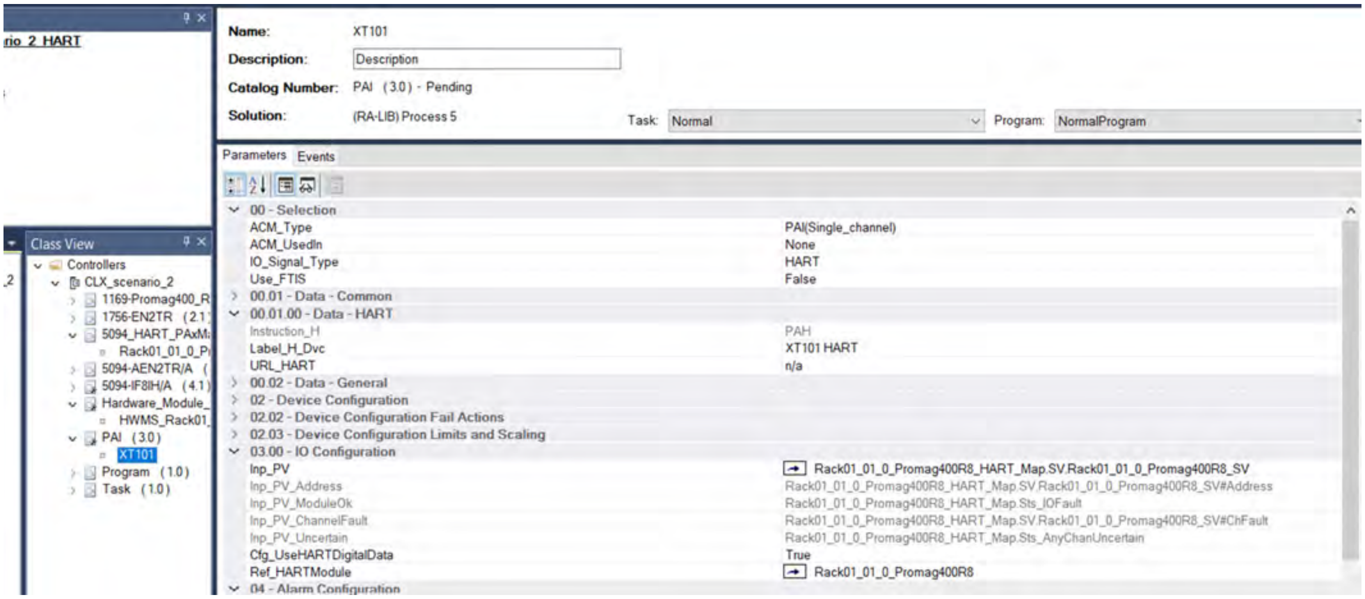
You need an instance of the library for each extra signal you want (PV, SV, TV, or QV).



4. From the Process library > Hardware Monitoring > Specialty folder, add a Hardware_Module_Status and configure the Module parameter for the 1756-IF8IH module.



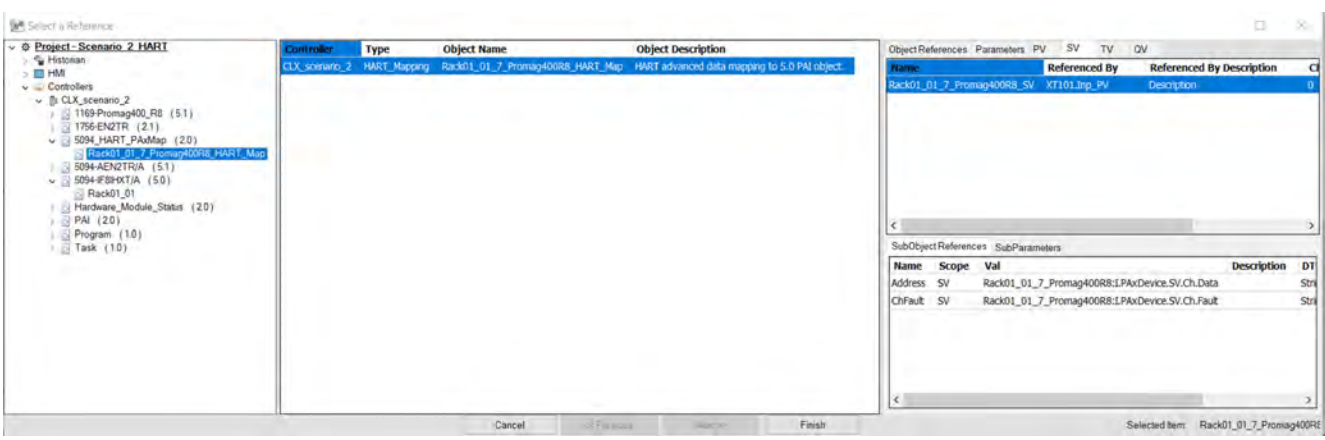
5. From the Process library > Control Strategies > Input Processing folder, add a PAI instance for the analog input module and configure these parameters in the I/O Configuration section.



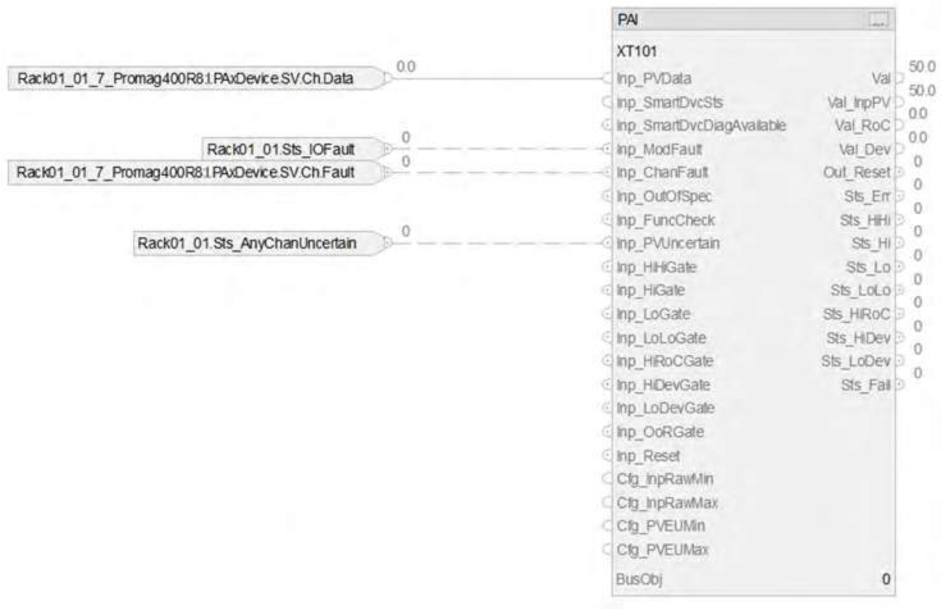
ACM Configuration for the PAI Instruction

ACM Parameter	Usage
Task Program	Assign a Task and Program for the PAI control strategy.
IO_Signal_Type	HART
Inp_PV	Set this reference to the PV, SV, TV, or QV of the 5094_HART_MapIO object that was created for the HART device
Cfg_ UseHARTDigitalData	Set to True
Ref_HARTModule	Set this reference to the 5094 module that the instrument is connected to

This example shows the SV value as the selection for the Inp_PV connection.

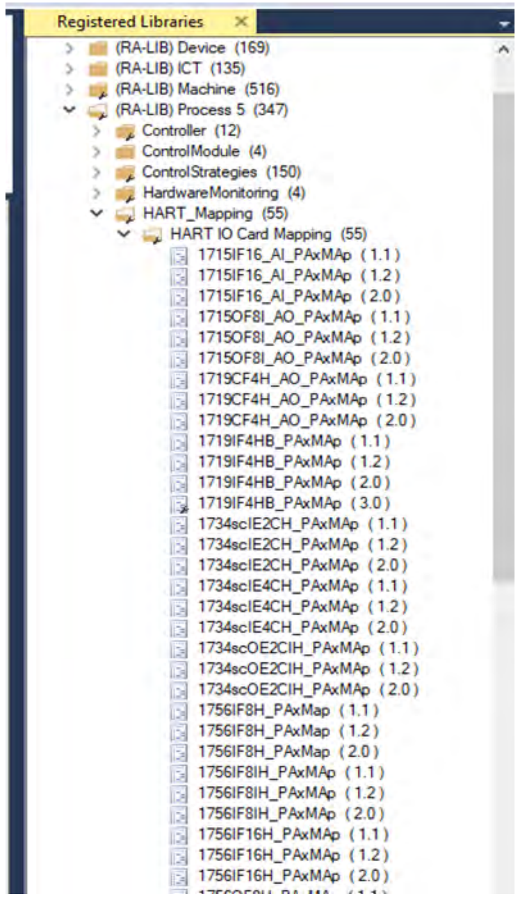


6. Generate the controller ACD file.



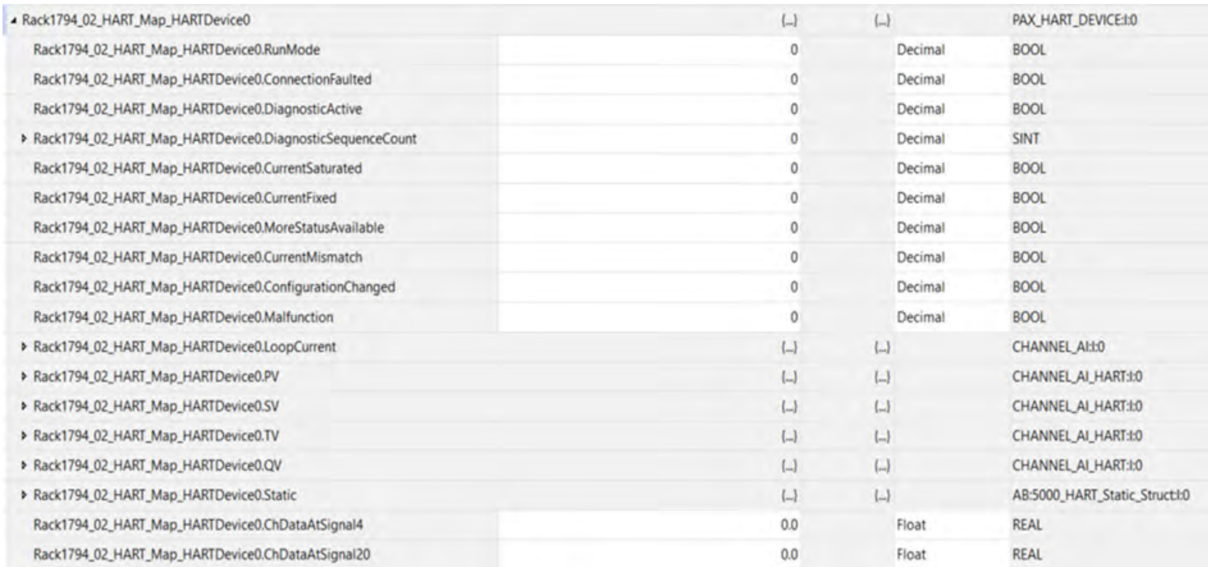
Integrate Other HART Modules with Process Controller

HART modules for other I/O platforms must be used with PlantPAx 5.0 instructions in a different way than FLEX 5000 modules. There is a HART-mapping ACM library for each HART I/O module in the HART_Mapping > HART IO Card Mapping folder.

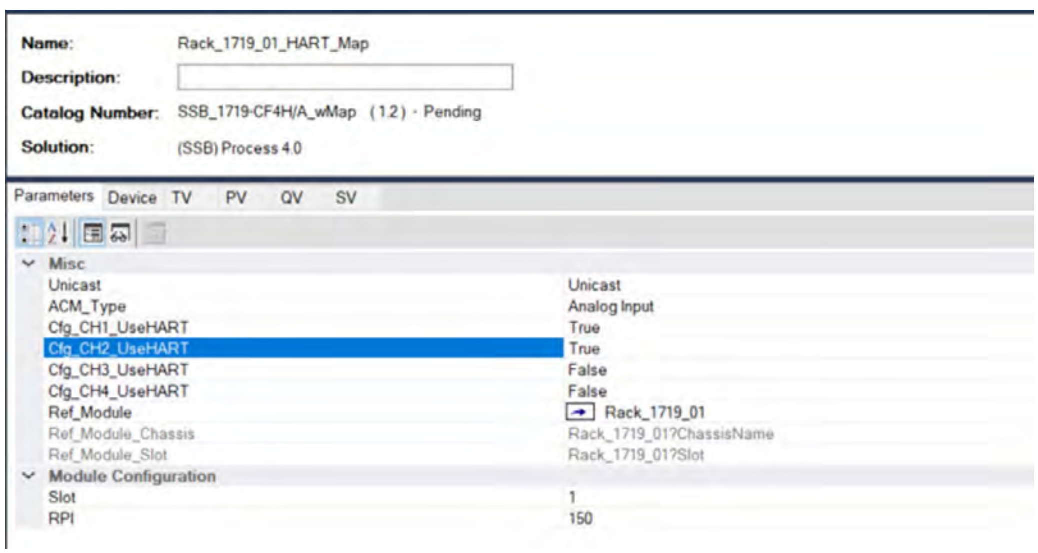


Each HART-mapping library has the following features:

- For each I/O Module, you can connect to a HART device that is connected to each channel.
- The HART device information is mapped into a standard data type PAX_HART_DEVICE:I:0



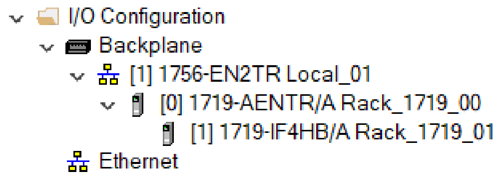
- For each channel, you can map any of the HART Digital Variables (PV, SV, TV, QV) to a PAI module.
- For each Channel of the HART module, you can connect to either the Device, PV, SV, TV, and QV (displayed as sub-objects for each mapping library).



Map HART Device to PAH from Non-FLEX 5000 I/O

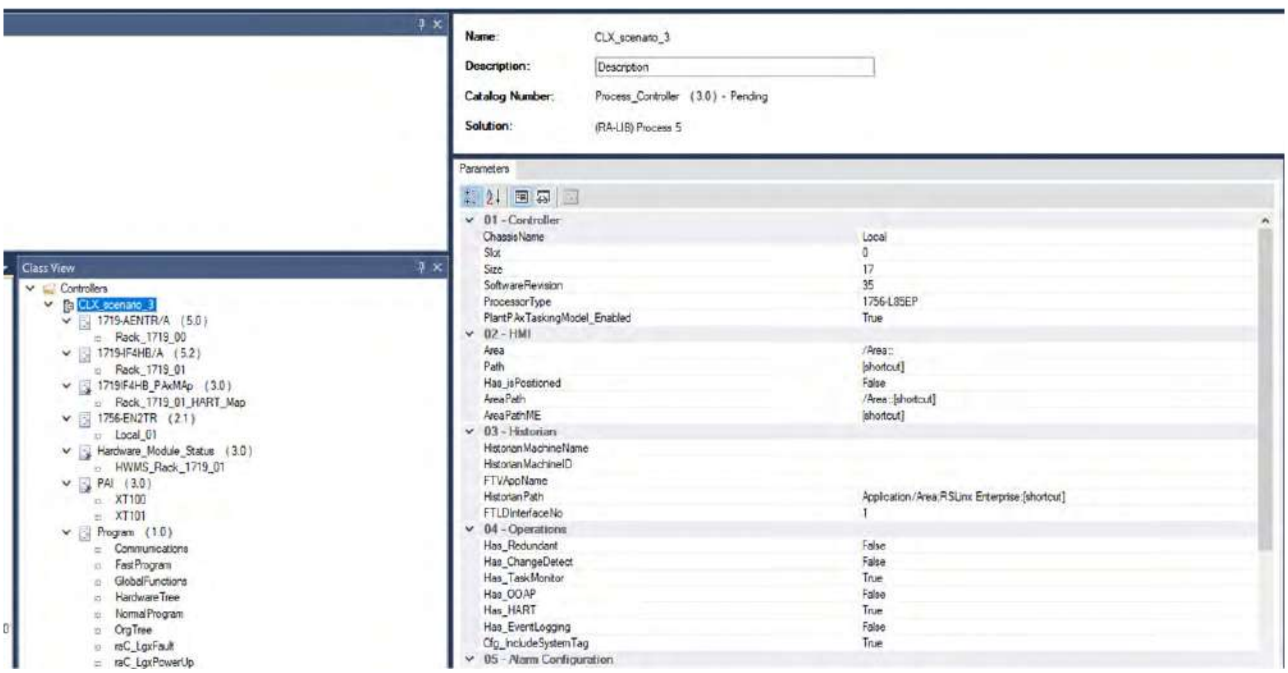
In this example, the ACM project contains:

- ControlLogix Process controller
- 1756-EN2TR communication module
- 1719-AENTR communication module connected to a 1719-IF4HB HART module
- Endress+Hauser
- ProsonicFlow 200 instrument connected to channel 1 of the 1719-IF4HB module
- Endress+Hauser
- Promag revision 9 instrument connected to channel 4 of the 1719-IF4HB module

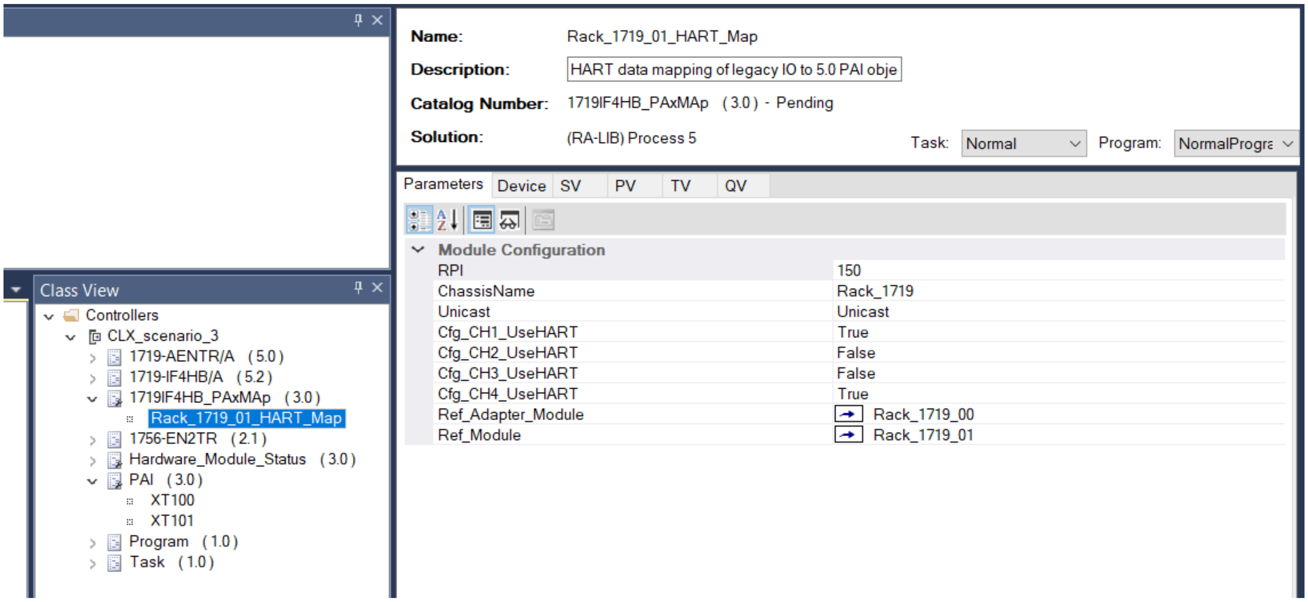


IMPORTANT	When you add the 1719-AENTR module to the ACM project, specify a unique rack name and IP address for the module.

1. Configure the process controller for parameters you need for your application, and make sure to set Has_HART.

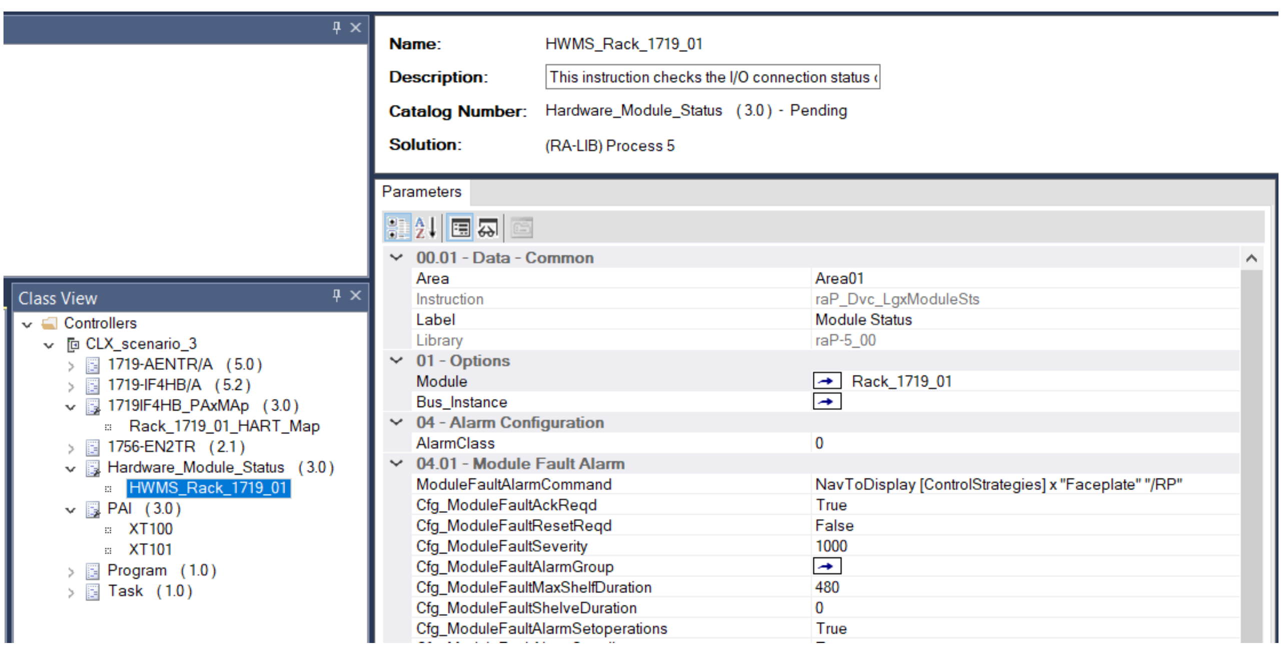


2. From the Process library > HART_Mapping > HART IO Card Mapping, create an instance of the 1719-IF4HB_PAxMap and connect to the 1719-IF4HB module.

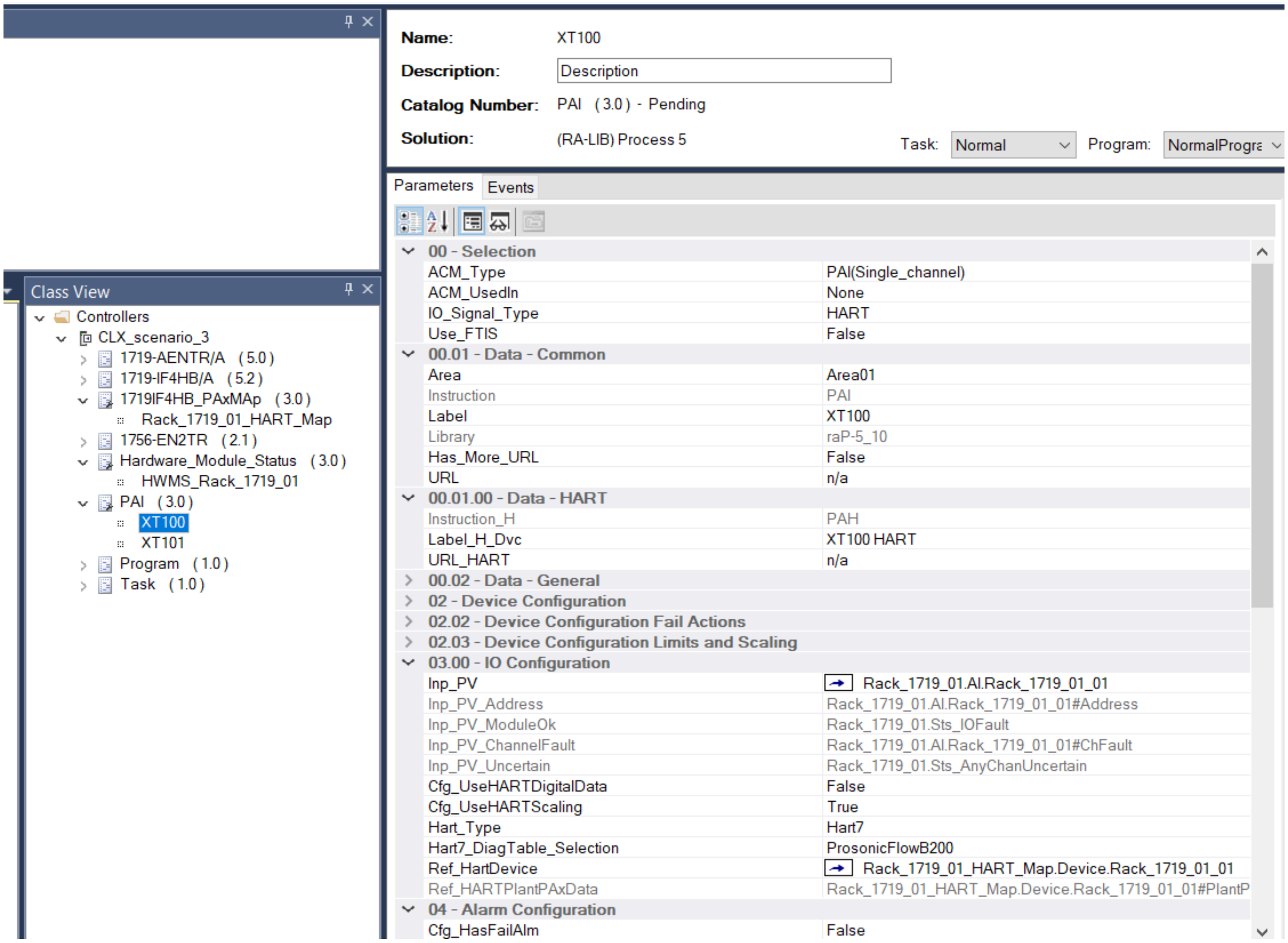


ACM Parameter	Usage
Cfg_CH1_UseHART	Set to True if a HART device is connected to this channel
Cfg_CH2_UseHART	Set to True if a HART device is connected to this channel
Cfg_CH3_UseHART	Set to True if a HART device is connected to this channel
Cfg_CH4_UseHART	Set to True if a HART device is connected to this channel
Ref_Adapter_Module	Select the EtherNet/IP adapter module connected the HART I/O module
Ref_Module	Select the HART I/O module

3. From the Process library > Hardware Monitoring > Specialty folder, add a Hardware_Module_Status and configure the Module parameter for the 1719-IF4HB module.



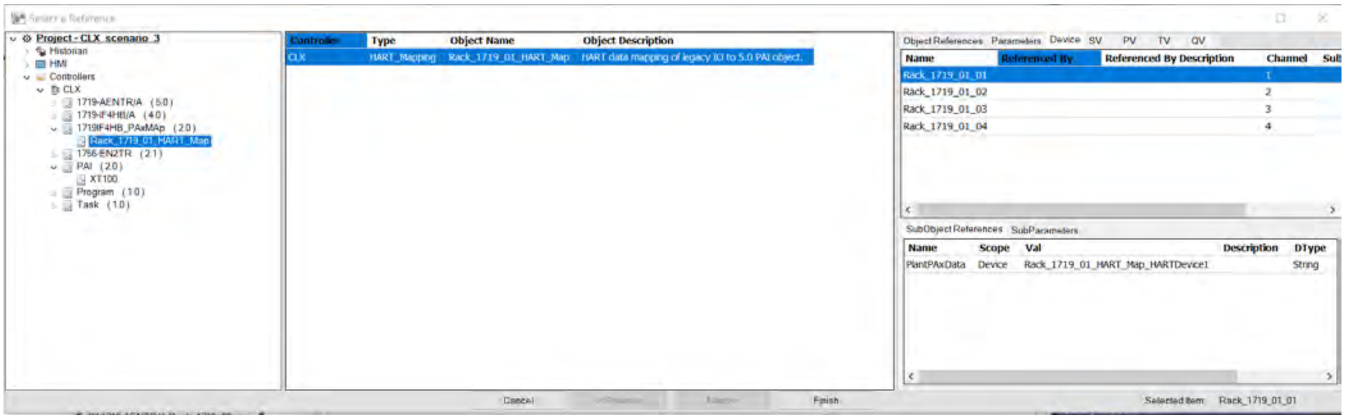
4. From the Process library > Control Strategies > Input Processing folder, add a PAI instance for each instrument and configure these parameters in the I/O Configuration section.



ACM Configuration for the PAI Instruction

ACM Parameter	Usage
Task Program	Assign a Task and Program for each PAI control strategy.
IO_Signal	HART
Inp_PV	Connect to the channel of the I/O module that the instrument is connected to.
Cfg_UseHARTDigitalData	Not applicable, leave at default value
Hart_Type	Select the HART protocol revision (Generic, Hart, Hart5, Hart6 orHart7)
Cfg_UseHARTScaling	Set to False
Hart7_DiagTable_Selection	Select the relevant Diag Table value for the instrument.
Ref_HartDevice	Connect to the mapping library in ACM and on the Device tab select the correct channel

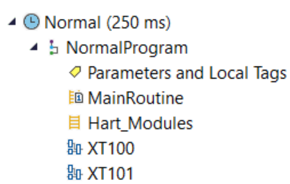
This example of the first PAI instance (XT100) shows the Ref_HartDevice for the Prosonic 200 instrument connected to channel 1 of the 1719-IF4HB module



This example of the second PAI instance (XT101) shows the Promag revision 9 instrument connected to channel 4 of the 1719-IF48H module.

5. Generate the controller ACD file.

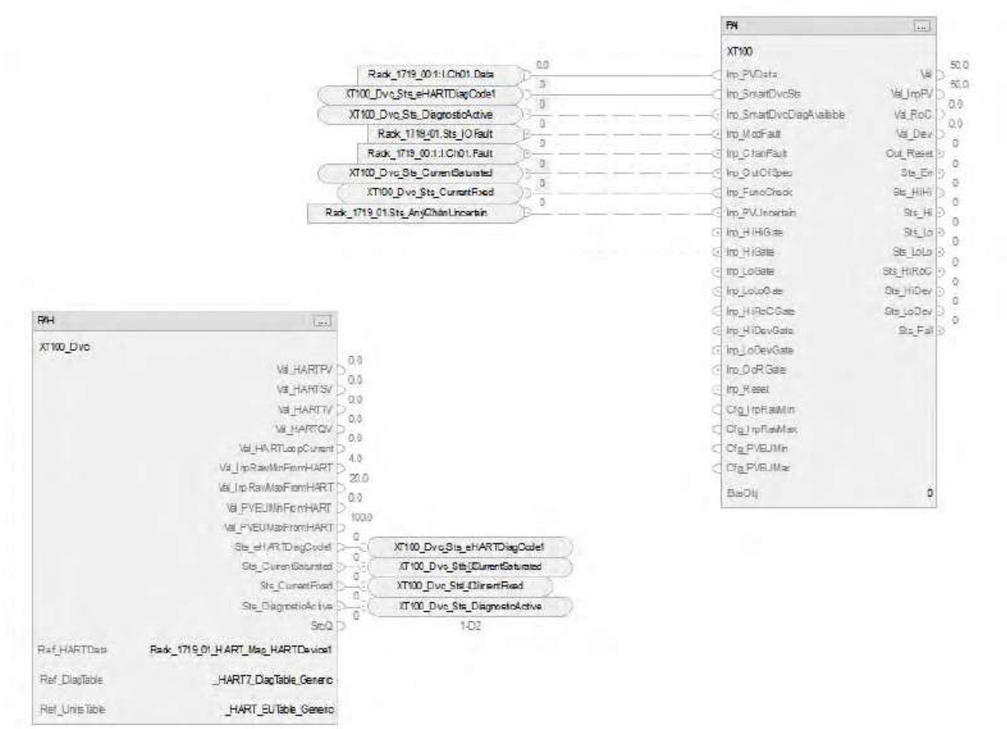
The controller code contains a routine for each HART instrument and a Hart_Modules routine.



The Hart_Modules routine:

- An Add-On Instruction is inserted which gets the data from the HART Module (in this case I_1718_1719_AI4H).
- The HART data is mapped into an array for each device (Rack_1719_01_HART_Map_HARTCH1 and Rack_1719_01_HART_Map_HARTCH1 CH4).
- Each element of the array is mapped into a tag of type PAX_HART_DEVICE:I:0 via the Add-On Instruction rap_Tec_HARTChanData_to_PAH.

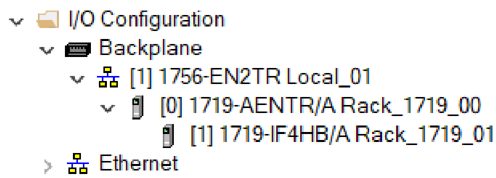
The instrument logic has no connection between the PAH module and the PAI module as the ACM parameter Cfg_UseHARTScaling is set to False.



Map HART Device Digital Data to PAI from Non-FLEX 5000 I/O

In this example, the ACM project contains:

- ControlLogix Process controller
- 1756-EN2TR communication module
- 1719-AENTR communication module connected to a 1719-IF4HB HART module
- Endress+Hauser
- ProsonicFlow 200 instrument connected to channel 2 of the 1719-IF4HB module



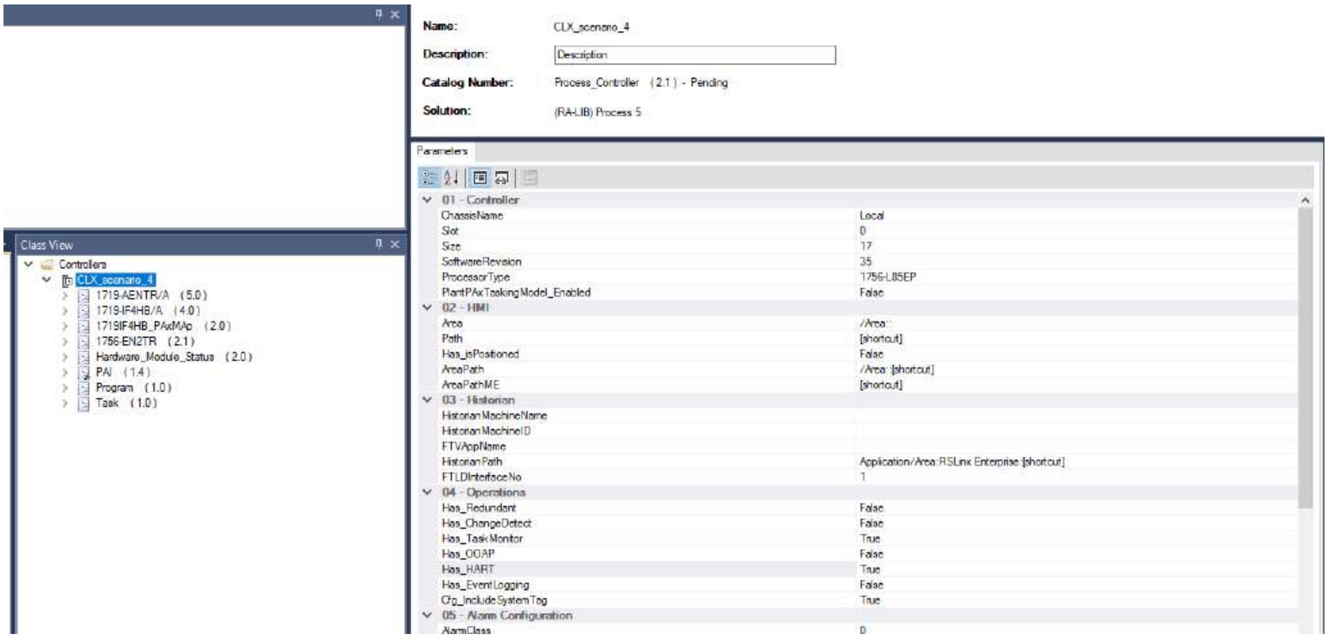
Map one of the HART digital signals PV, SV, TV, QV to a PAI Module.

The TV of the HART device that is connected to Channel 2 of the 1719-IF4HB is connected to the PAI module

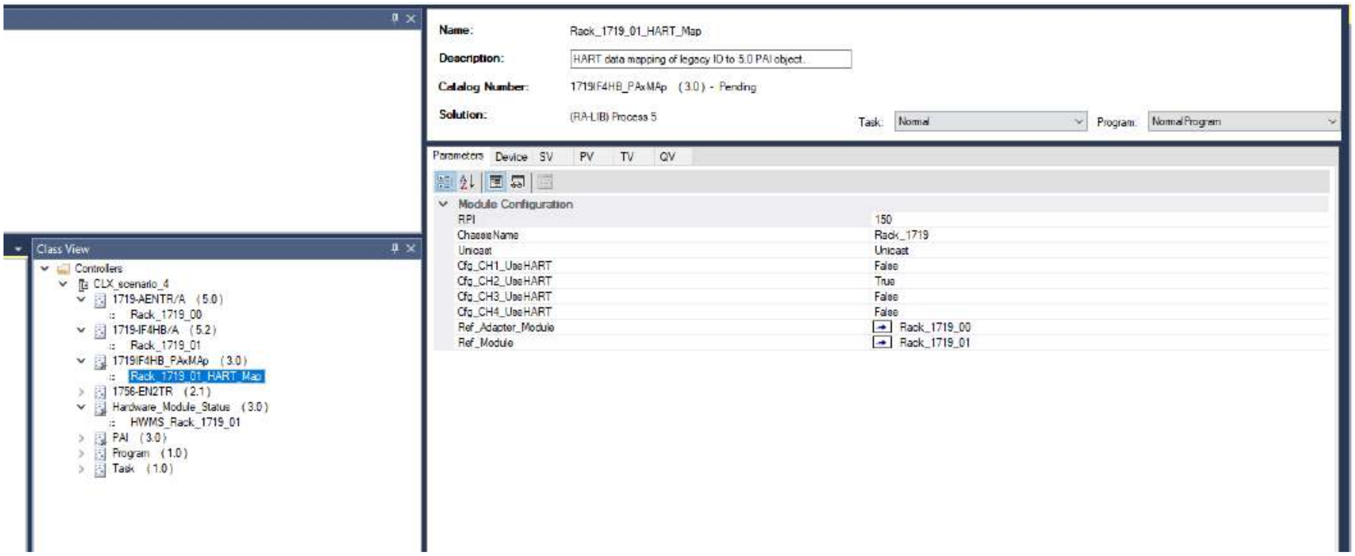
IMPORTANT	When you add the 1719-AENTR module to the ACM project, specify a unique rack name and IP address for the module.

Add the devices to the ACM project and configure parameters as needed.

1. Configure the process controller for parameters you need for your application, and make sure to set Has_HART.



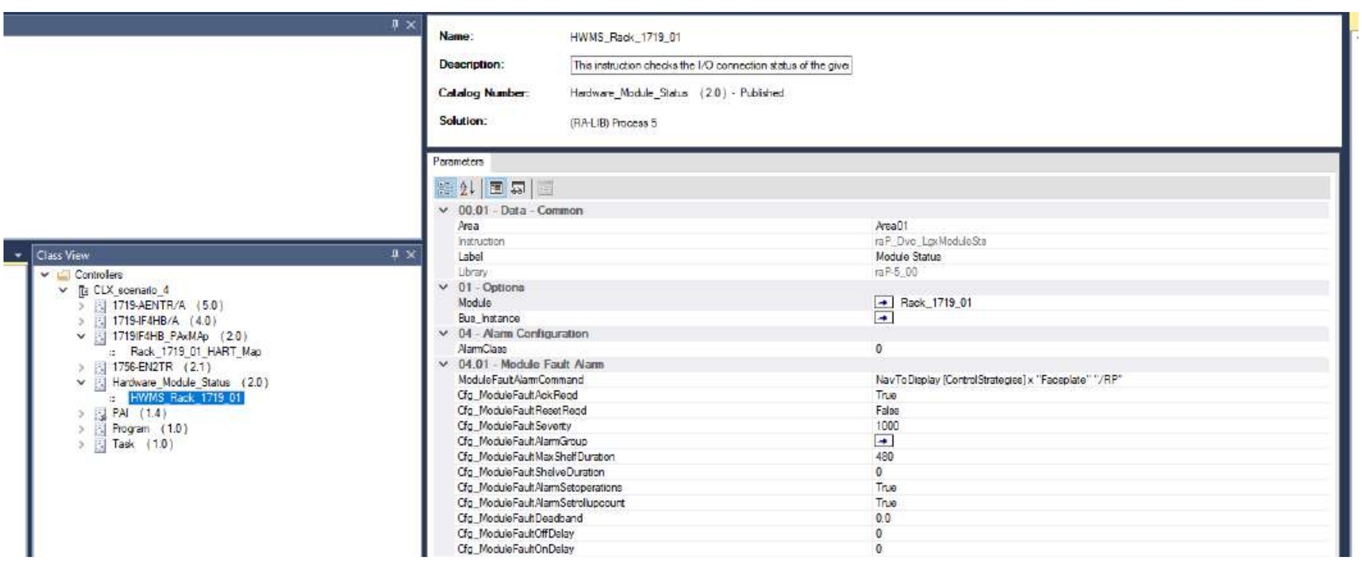
2. From the Process library > HART_Mapping > HART IO Card Mapping, create an instance of the 1719-IF4HB_PAxMap and connect to the 1719-IF4HB module.



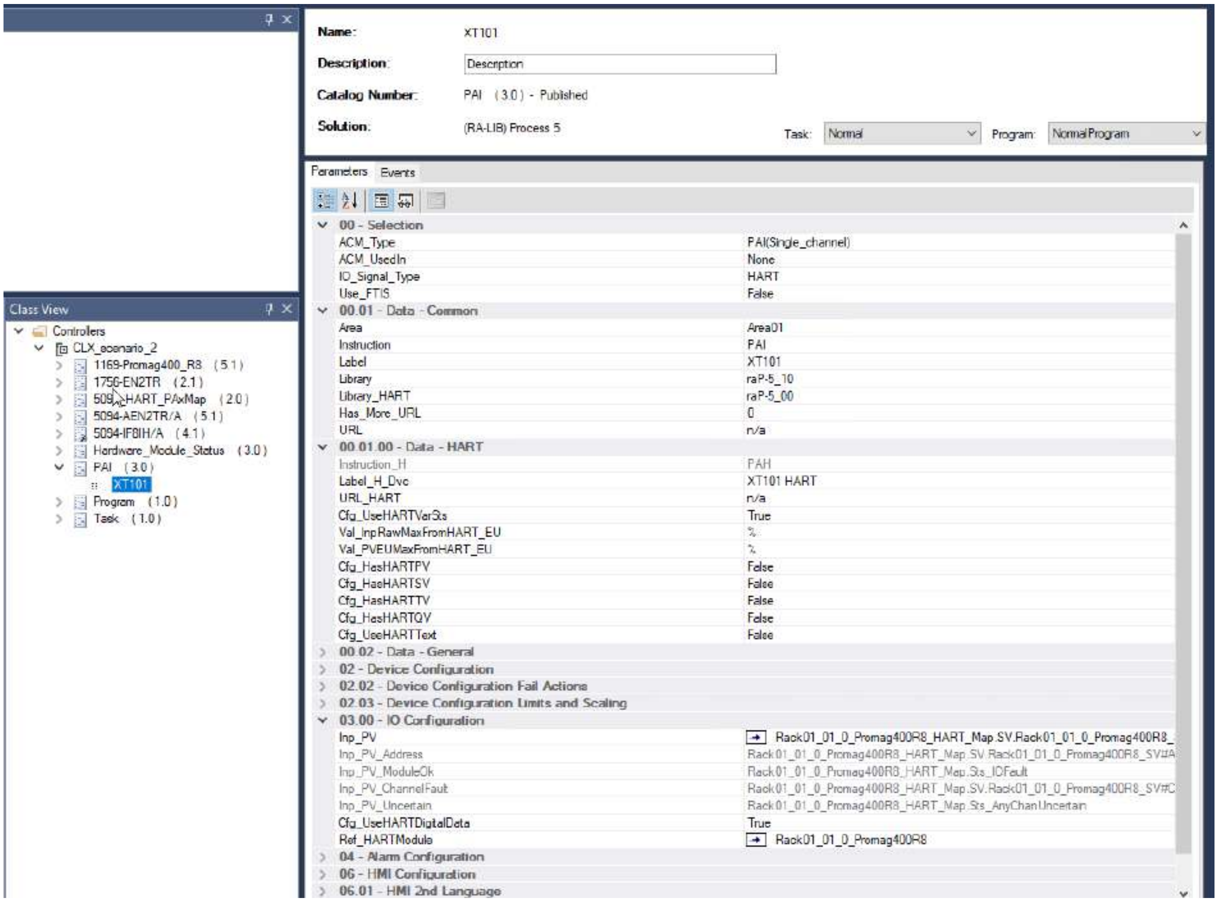
You need an instance of the library for each extra signal you want (PV, SV, TV, or QV).

ACM Parameter	Usage
Cfg_CH1_UseHART	Set to True if a HART device is connected to this channel
Cfg_CH2_UseHART	Set to True if a HART device is connected to this channel
Cfg_CH3_UseHART	Set to True if a HART device is connected to this channel
Cfg_CH4_UseHART	Set to True if a HART device is connected to this channel
Ref_Adapter_Module	Select the EtherNet/IP adapter module connected the HART I/O module
Ref_Module	Select the HART I/O module

3. From the Process library > Hardware Monitoring > Specialty folder, add a Hardware_Module_Status and configure the Module parameter for the 1719-IF4HB module.



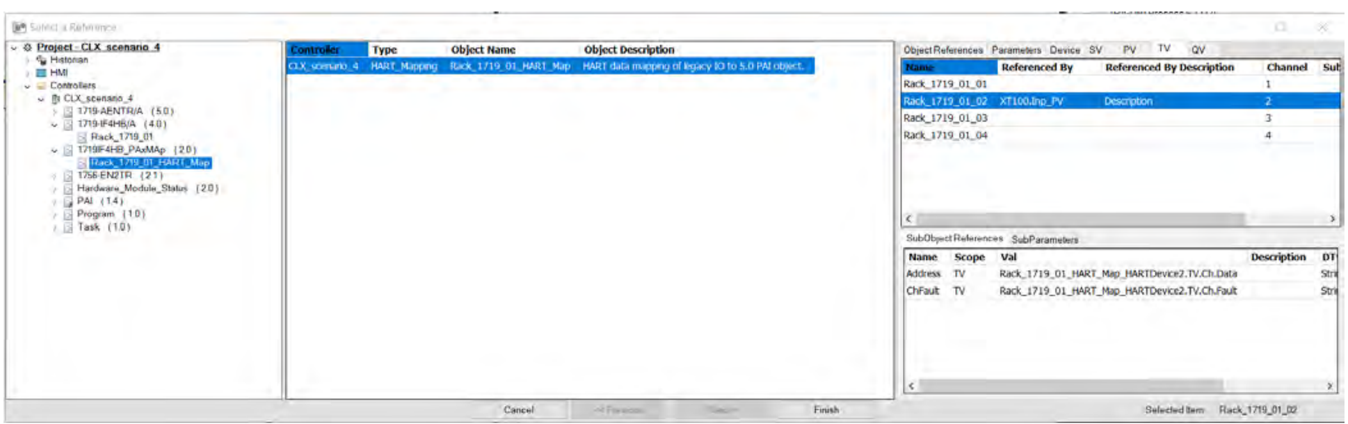
4. From the Process library > Control Strategies > Input Processing folder, add a PAI instance for each instrument and configure these parameters in the I/O Configuration section.



ACM Parameters for PAI module

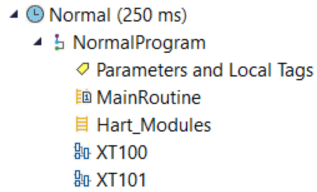
ACM Parameter	Usage
Task Program	Assign a Task and Program for the PAI control strategy.
IO_Signal	HART
Inp_PV	Set this reference to the PV, SV, TV, or QV of the HART_MapIO object that was created for the HART device
Cfg_ UseHARTDigitalData	Set to True
Ref_HARTModule	Set this reference to the HART I/O module that the instrument is connected to

- This example shows the TV value as the selection for the Inp_PV connection.



5. Generate the controller ACD file.

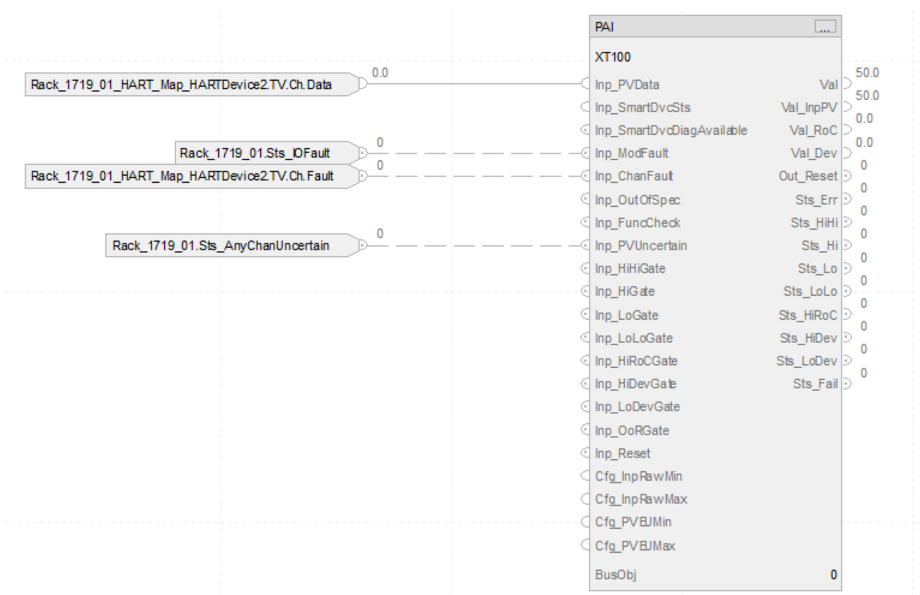
The controller code contains a routine for each HART instrument and a Hart_Modules routine.



The Hart_Modules routine:

- An Add-On Instruction is inserted which gets the data from the HART Module (in this case I_1718_1719_AI4H).
- The HART data is mapped into an array for the device (Rack_1719_01_HART_Map_HARTCH2).
- Each element of the array is mapped into a tag of type PAX_HART_DEVICE:I:0 via the Add-On Instruction rap_Tec_HARTChanData_to_PAH.

The instrument logic has no PAH module.



Notes:

Controller Fault Handler Control Strategy

If a fault condition occurs that prevents an instruction from running, the instruction aborts, and the controller reports a major fault. A major fault halts logic execution and the controller switches to faulted mode (the OK status indicator flashes red). Depending on the application, you may not want all major faults to shut down the system. If you do not want all major faults to shut down the system, create a fault routine to clear the fault and let the application continue to run. The process of resuming execution after the fault clears is known as fault recovery.

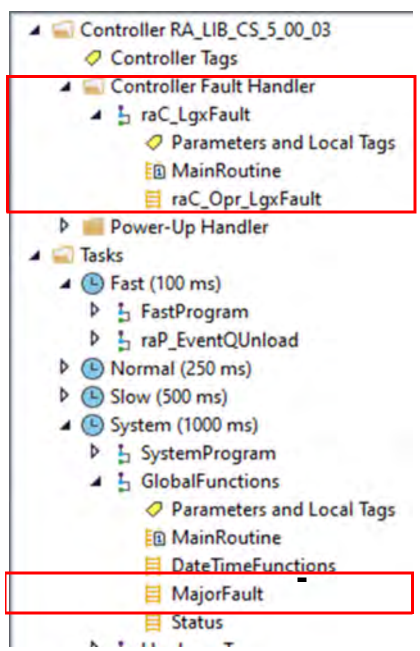
The Controller Fault Handler control strategy has the following functionality:

- Get fault information for the active fault
- Generate an alarm that a major fault occurred
- Record the last 10 major faults that occurred containing a time stamp
- Configuration to automatically clear major fault to help prevent controller from faulting (use selectively)

The Controller Fault Handler control strategy requires:

- a raC_LgxFault **program** (with raC_Opr_LgxFlt and MainRoutine **routines**) in the Controller Fault Handler folder,
- and a MajorFault **routine** in the GlobalFunctions Program.

Import the **routines** into the correct program.



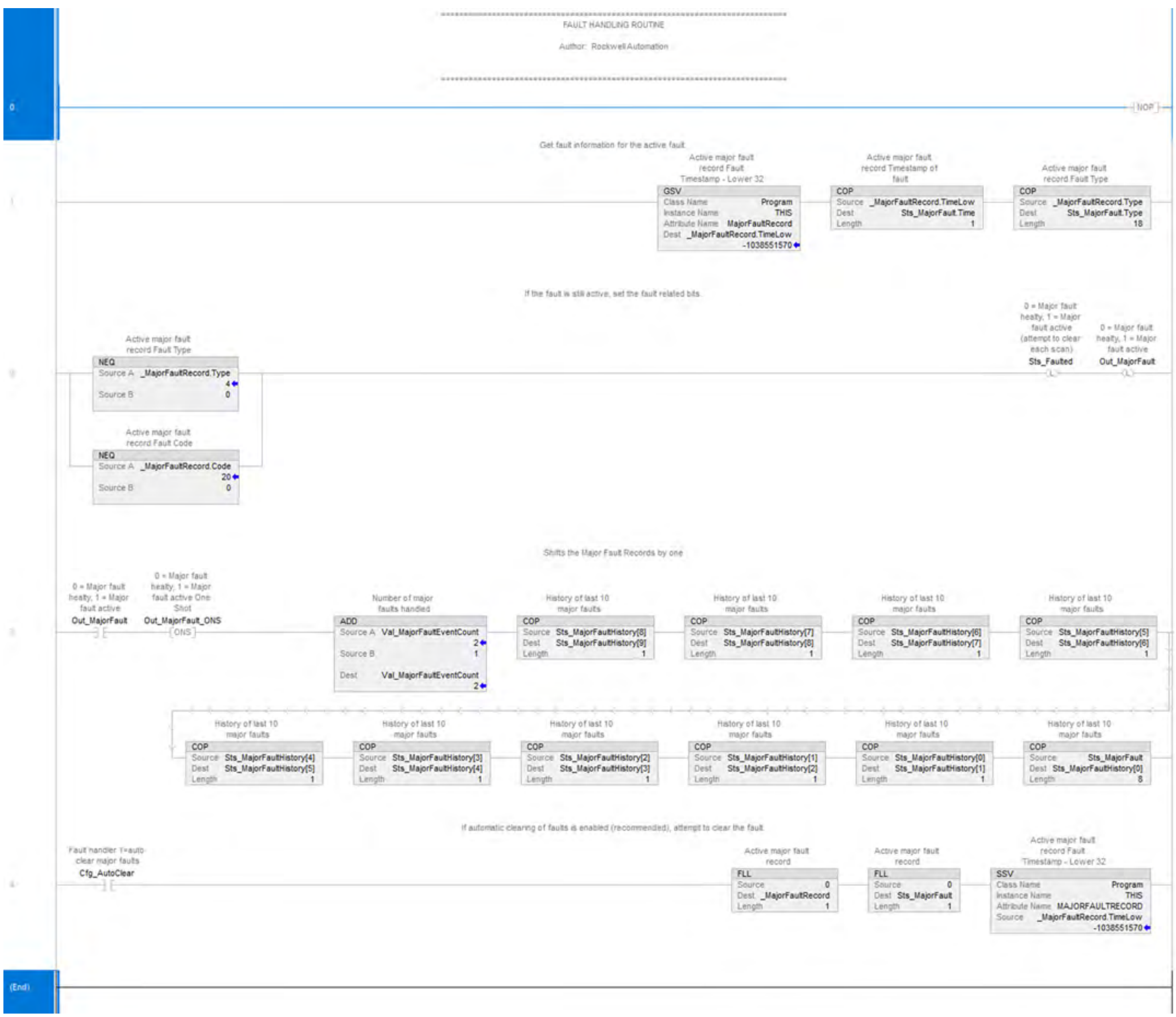
Example:

In a system that uses recipe numbers as indirect addresses, a mistyped number could produce a major fault, such as type 4, code 20.

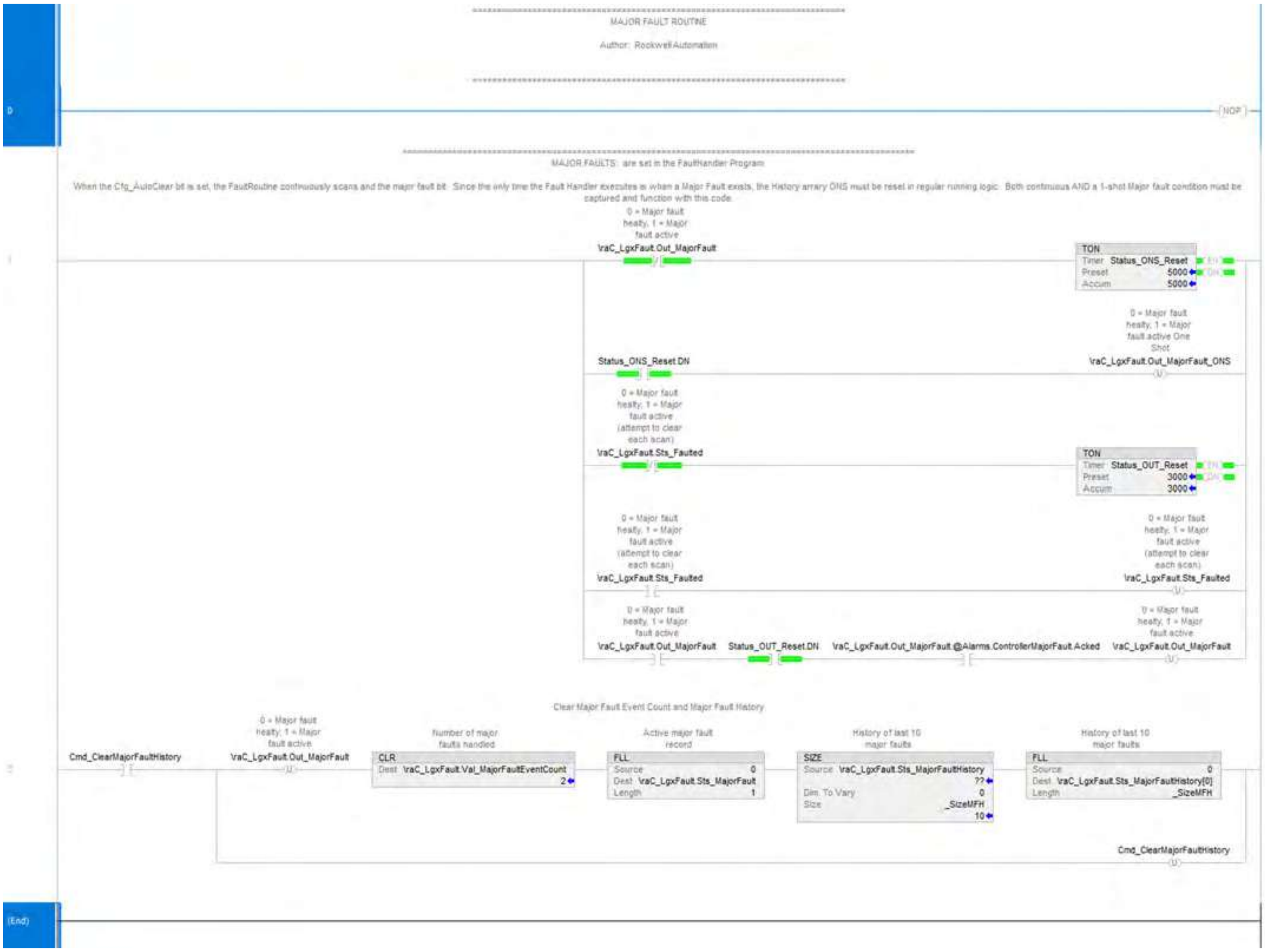
To keep the entire system from shutting down, a fault routine clears any type 4, code 20, major faults.

Controller Fault Handler Program: MainRoutine Routine

Controller Fault Handler Program: raC_Opr_LgxFault Routine



GlobalFunctions Program: MajorFault Routine



For more information on handling controller faults, see the Logix 5000® Controllers Major, Minor, and I/O Faults Programming Manual, publication 1756-PM014.

Notes:

Chapter 4

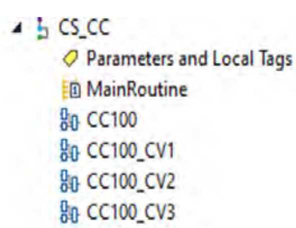
Coordinated Control (CC) Control Strategy

Use the CC control strategy to control one process variable by manipulating up to three different control variables. Any of the three outputs can be used as an input to create feed forward action in the controller. The CC instruction calculates the control variables (CV1, CV2, and CV3) in the auto mode based on the PV - SP deviation, internal models, and tuning.

The CC control strategy is a model-based instruction, where as many as three models can be configured to relate the output of each CV to the single PV. Each model is a first order plus del ay (FOPD) response, which is more effective than PID controllers for controlling processes with long deadtimes. The CC control strategy coordinates the action of the CVs to limit interactions among the CVs.

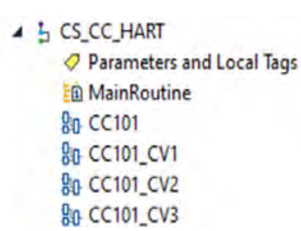
The CC control strategy is available as four routines in the process library:

Routine	Description
CC100	Coordinated Control instruction.
CC100_CV1 CC100_CV2 CC100_CV3	Control variable routines.



The CC HART control strategy is available as four routines in the process library:

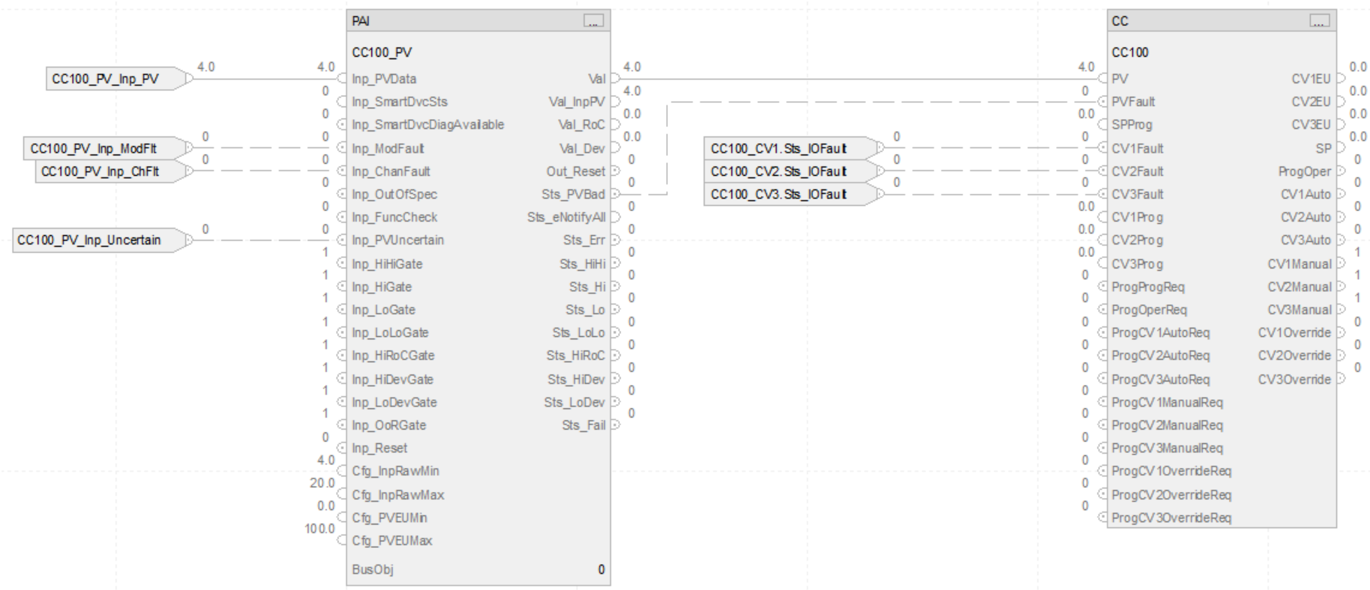
Routine	Description
CC101	Coordinated Control instruction with HART input in the CC101 routine.
CC101_CV1 CC101_CV2 CC101_CV3	Control variable routines.



Import the appropriate control strategy as a **program** in your controller project.

CS_CC Sheet

The CC100 routine has this sheet.



PAI Input References

See CS_PAI Sheet on page 71 for details.

- Substitute CC100 for XT101

PAI Outputs to CC Inputs

Parameter	Description
Val	Value for PV parameter Process Variable (PVEU) Source: Analog input channel or upstream REAL tag representing position feedback
Sts_PVBad	Quality of PV value 1 = PV quality is flagged as Bad

Input References to CC

Parameter	Description
CC_100_CV1.Sts_IOFault	Control variable 1 fault input • If CV1EU controls an analog output, then CV1Fault will normally come from the analog output's fault status. • If CV1Fault is TRUE, it indicates an error on the output module, set bit in Status.
CC_100_CV2.Sts_IOFault	Control variable 2 fault input • If CV2EU controls an analog output, then CV2Fault will normally come from the analog output's fault status. • If CV2Fault is TRUE, it indicates an error on the output module, set bit in Status.
CC_100_CV3.Sts_IOFault	Control variable 3 fault input • If CV3EU controls an analog output, then CV3Fault will normally come from the analog output's fault status. • If CV3Fault is TRUE, it indicates an error on the output module, set bit in Status.

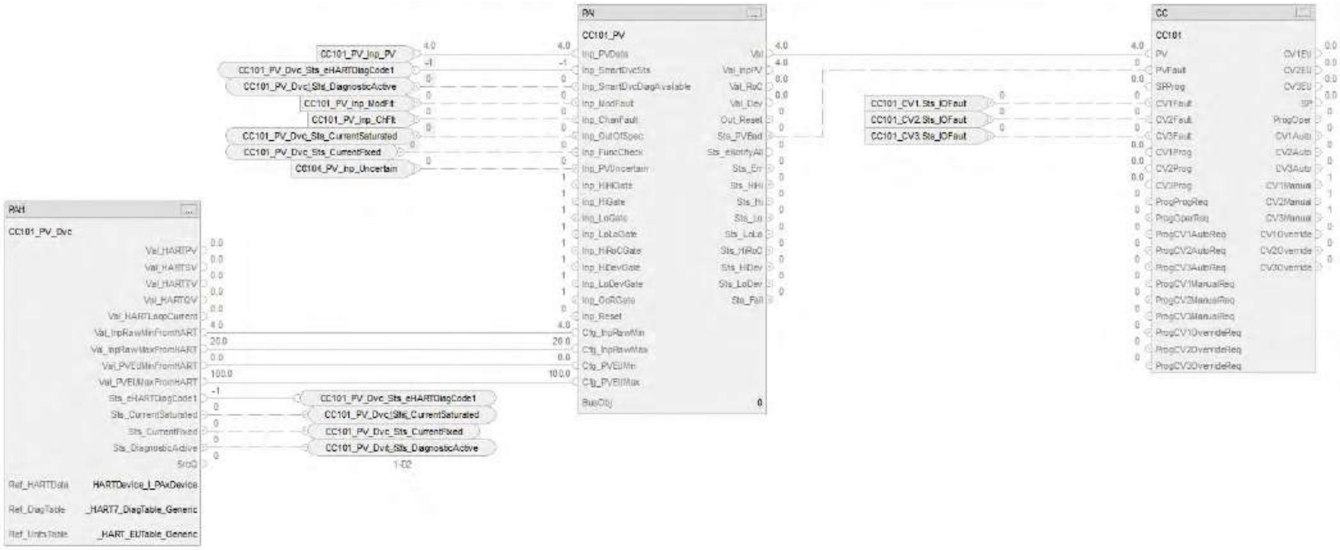
For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

CC Configuration Considerations

Operand	Type	Description
CC tag	COORDINATED_CONTROL	Instance of data structure (backing tag) required for proper operation of instruction.

CS_CC_HART Sheet

The CC101 routine has this sheet.



The CS_CC_Hart control strategy operates the same as the CS_CC control strategy but relies on HART input data.

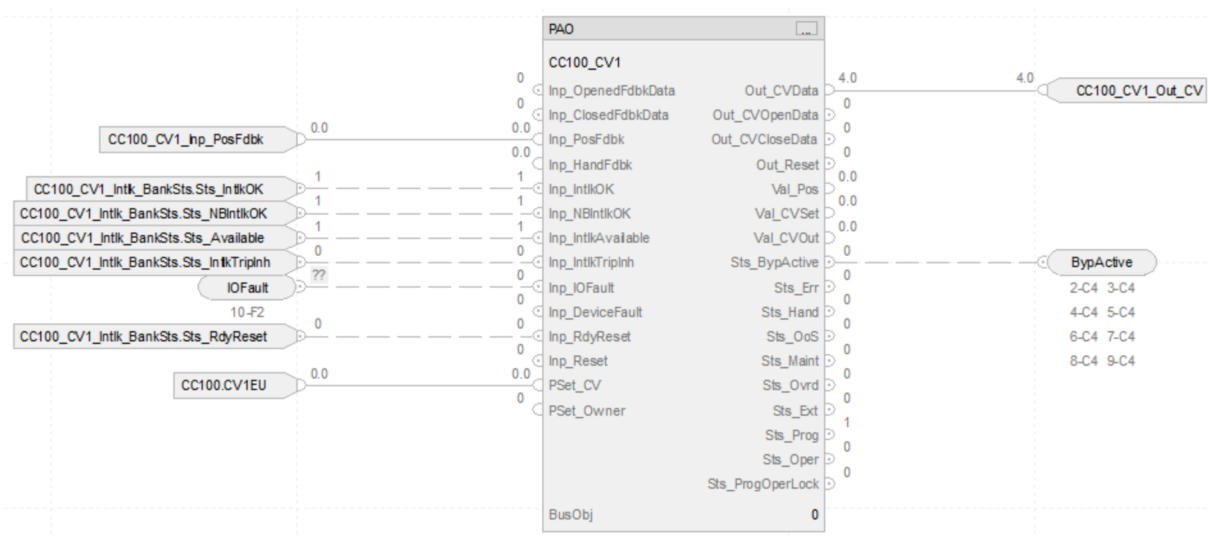
- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute CC101 for XT100
- For more information, see HART Integration on page 31.

Control Variable Routines

This control variable routines include the PAO control strategy, with an additional input reference.

- For PAO configuration considerations, and input and output references, see CS_PAO Sheet on page 84 for details.
- The routines also include these PAO input references:

Parameter	Description
CC100.CV1EU	Scaled control variable output for CV1. Scaled by using CV1EUMax and CV1EUMin, where CV1EUMax corresponds to 100% and CV1EUMin corresponds to 0%. This output is typically used to control an analog output module or a secondary loop. CV1EU = (CV1 * CV1EUSpan / 100) + CV1EUMin CV1EU span calculation: CV1EUSpan = (CV1EUMax - CV1EUMin)
CC100.CV2EU	Scaled control variable output for CV2. Scaled by using CV2EUMax and CV2EUMin, where CV2EUMax corresponds to 100% and CV2EUMin corresponds to 0%. This output is typically used to control an analog output module or a secondary loop. CV2EU = (CV2 * CV2EUSpan / 100) + CV2EUMin CV2EU span calculation: CV2EUSpan = (CV2EUMax - CV2EUMin)
CC100.CV3EU	Scaled control variable output for CV3. Scaled by using CV3EUMax and CV3EUMin, where CV3EUMax corresponds to 100% and CV3EUMin corresponds to 0%. This output is typically used to control an analog output module or a secondary loop. CV3EU = (CV3 * CV3EUSpan / 100) + CV3EUMin CV3EU span calculation: CV3EUSpan = (CV3EUMax - CV3EUMin)



Internal Model Control (IMC) Control Strategy

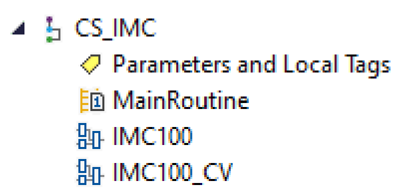
Use the IMC control strategy to control a single process variable by manipulating a single control-variable output. This control strategy performs an algorithm where the actual error signal is compared against that of an internal first-order lag plus deadtime model of the process. The IMC instruction calculates the control variable output (CV) in the Auto mode based on the PV - SP deviation, internal model, and tuning. IMC is a model-based instructions that is more effective than PID control for processes with long deadtimes.

The following IMC control strategies are available as routines in the process library:

- CS_IMC
- CS_IMC_HART

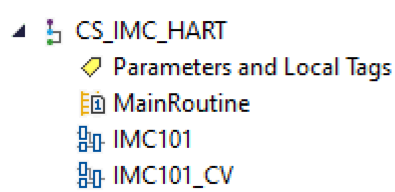
The IMC control strategy is available as two routines in the process library:

Routine	Description
IMC100	Internal Model Control instruction.
IMC100_CV1	Control variable routine.



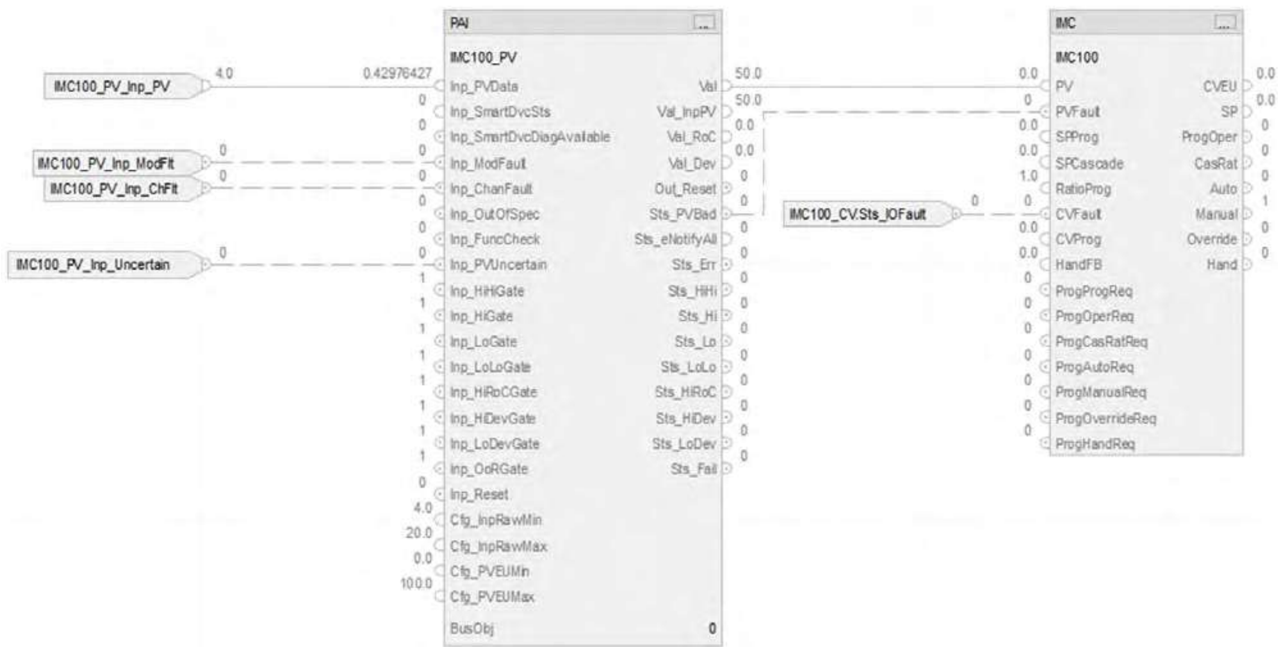
The IMC HART control strategy is available as two routines in the process library:

Routine	Description
IMC101	Coordinated Control instruction with HART input.
IMC101_CV1	Control variable routine.



Import the appropriate control strategy as a **program** in your controller project.

CS_IMC Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.

- Substitute IMC100 for XT101

PAI Outputs to IMC Inputs

Parameter	Description
Val	Value for PV parameter Process Variable (PVEU) Source: Analog input channel or upstream REAL tag that represents position feedback
Sts_PVBad	Quality of PV value 1 = PV quality is flagged as Bad

IMC Input Reference

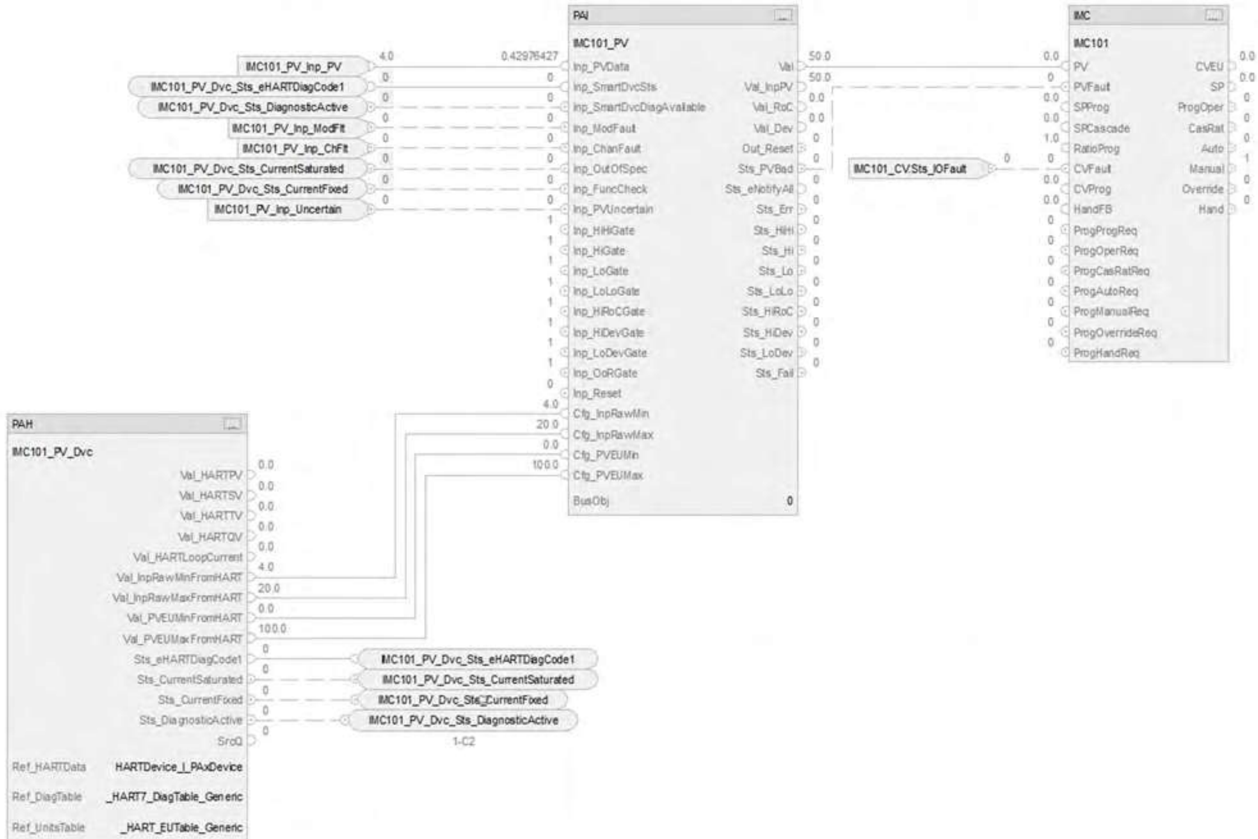
Parameter	Description
IMC_100_CV.Sts_IOFault	Control variable fault input • If CVEU controls an analog output, then CVFault normally comes from the analog output's fault status. • If CVFault is TRUE, it indicates an error on the output module, set bit in Status.

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

IMC Configuration Considerations

Operand	Type	Description
IMC tag	Internal Model Control	Instance of data structure (backing tag) required for proper operation of instruction

CS_IMC_HART Sheet



The CS_IMC_Hart control strategy operates the same as the CS_IMC control strategy but relies on HART input data.

- For information on PAH outputs to PAI, see CS_PAI_HART Sheet on page 72.
    - Substitute CC101 for XT100

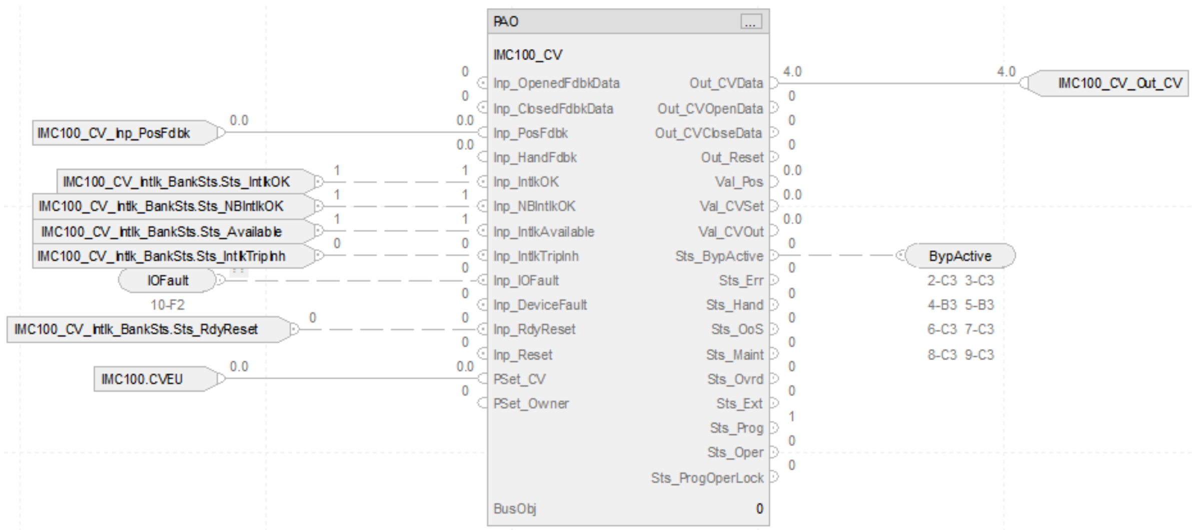
For more information, see HART Integration on page 31.

Control Variable Routines

This control variable routines include the PAO control strategy, with an additional input reference.

- For PAO configuration considerations, and input and output references, see CS_PAO Sheet on page 84 for details.
- The routine also includes this PAO input reference:

Parameter	Description
IMC100.CVEU	Scaled control variable output for CV1. Scaled by using CV1EUMax and CV1EUMin, where CV1EUMax corresponds to 100% and CV1EUMin corresponds to 0%. This output is typically used to control an analog output module or a secondary loop. CV1EU = (CV1 * CV1EUSpan / 100) + CV1EUMin CV1EU span calculation: CV1EUSpan = (CV1EUMax - CV1EUMin)



Modular Multivariable Control (MMC) Control Strategy

Use the MMC control strategy to control two process variables to their setpoints using as many as three control variables. The MMC instruction calculates the control variables (CV1, CV2, and CV3) in the auto mode based on the PV1 -SP1, PV2 - SP2 deviation, internal model, and tuning.

The MMC controller is a model-based instuction, where you can configure as many as six models to relate the output of each CV to the two PVs. Each model is a first order plus delay (FOPD) response, which is more effective than PID controllers at controlling processes with long deadtimes. The MMC control strategy coordinates the actions of the CVs to limit interactions among the CVs and control the 2 PVs to their respective setpoints.

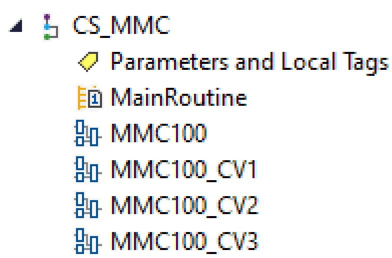
The following MMC control strategies are available as routines in the process library:

- CS_MMC
- CS_MMC_HART

Import the appropriate control strategy as a **program** in your controller project.

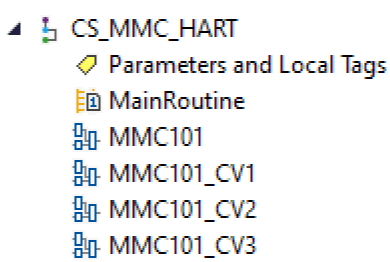
The CC control strategy is available as four routines in the process library:

Routine	Description
MMC100	Coordinated Control instruction.
MMC100_CV1 MMC100_CV2 MMC100_CV3	Control variable routines.

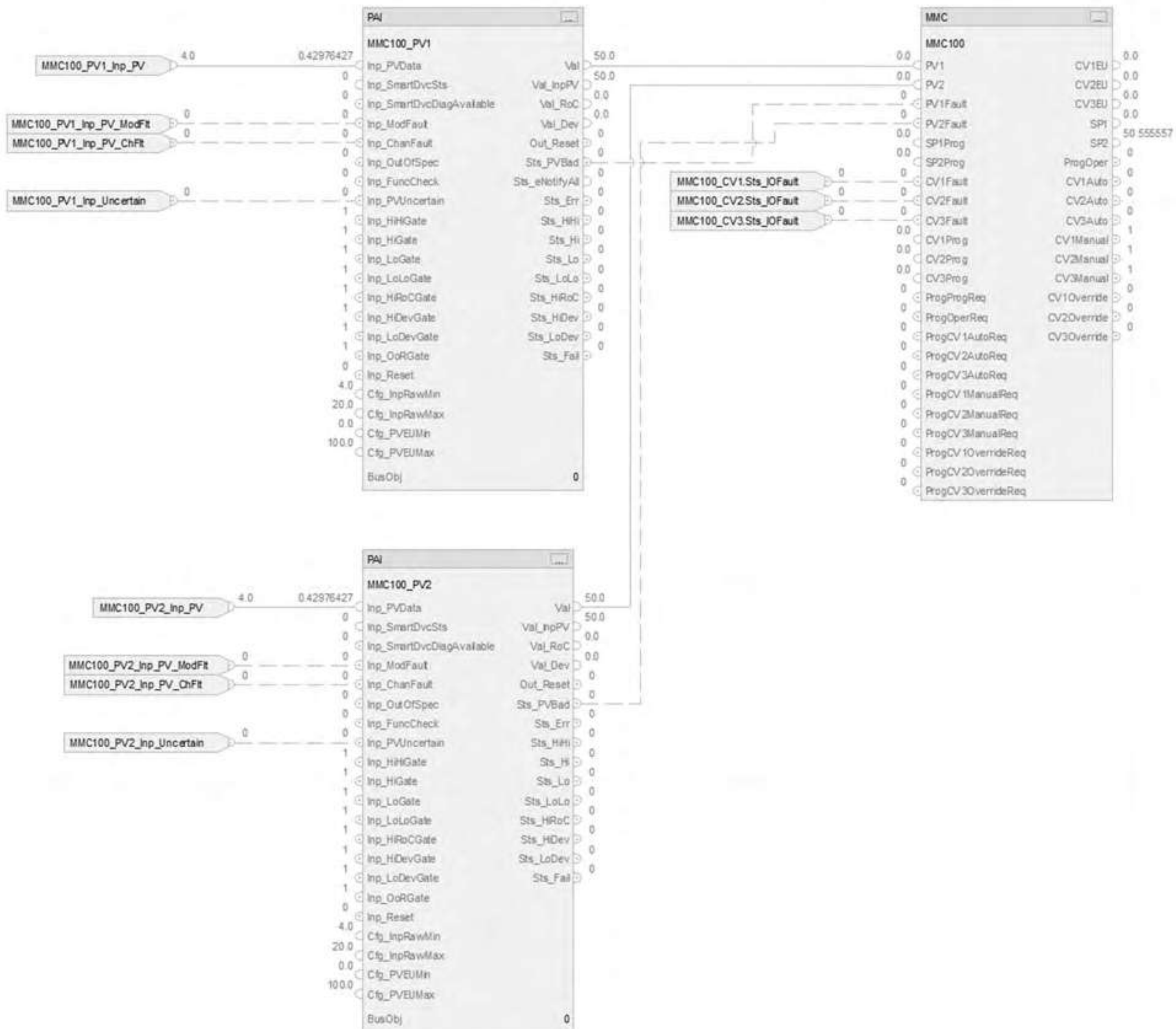


The CC HART control strategy is available as four routines in the process library:

Routine	Description
MMC101	Coordinated Control instruction with HART input in the CC101 routine.
MMC101_CV1 MMC101_CV2 MMC101_CV3	Control variable routines.



CS_MMC Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.

- Substitute MMC100_PV1 for the first instance of XT101
- Substitute MMC100_PV2 for the second instance of XT101

PAI Outputs to MMC Inputs

One instance for PV1 and a second instance for PV2.

Parameter	Description
Val	Value for PV parameter Process Variable (PVEU) Source: Analog input channel or upstream REAL tag that represents position feedback
Sts_PVBad	Quality of PV value 1 = PV quality is flagged as Bad

MMC Input References

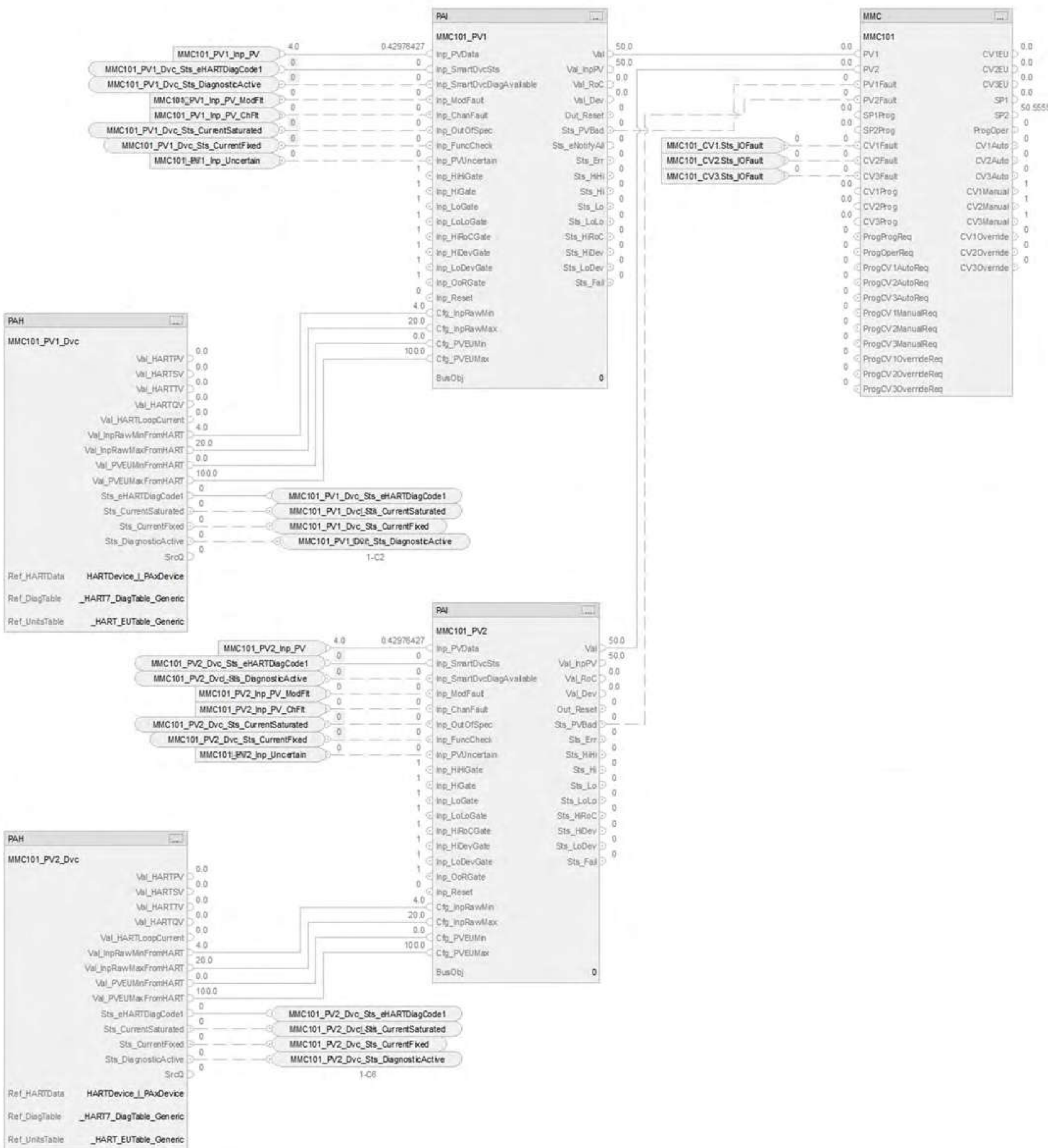
Parameter	Description
MMC_100_CV1.Sts_IOFault	Control variable 1 fault input If CV1EU controls an analog output, then CV1Fault normally comes from the analog output's fault status. If CV1Fault is TRUE, it indicates an error on the output module, set bit in Status.
MMC_100_CV2.Sts_IOFault	Control variable 2 fault input If CV2EU controls an analog output, then CV2Fault normally comes from the analog output's fault status. If CV2Fault is TRUE, it indicates an error on the output module, set bit in Status.
MMC_100_CV3.Sts_IOFault	Control variable 3 fault input If CV3EU controls an analog output, then CV3Fault normally comes from the analog output's fault status. If CV3Fault is TRUE, it indicates an error on the output module, set bit in Status.

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

MMC Configuration Considerations

Operand	Type	Description
MMC tag	MODULAR MULTIVARIABLE CONTROL	Instance of data structure (backing tag) required for proper operation of instruction

CS_MMC_HART Sheet



The CS_MMC_Hart control strategy operates the same as the CS_MMC control strategy but relies on HART input data.

- For information on PAH outputs to PAI, see CS_PAI_HART Sheet on page 72.
  - Substitute MMC101_PV1 for the first instance of XT100
  - Substitute MMC101_PV2 for the second instance of XT100

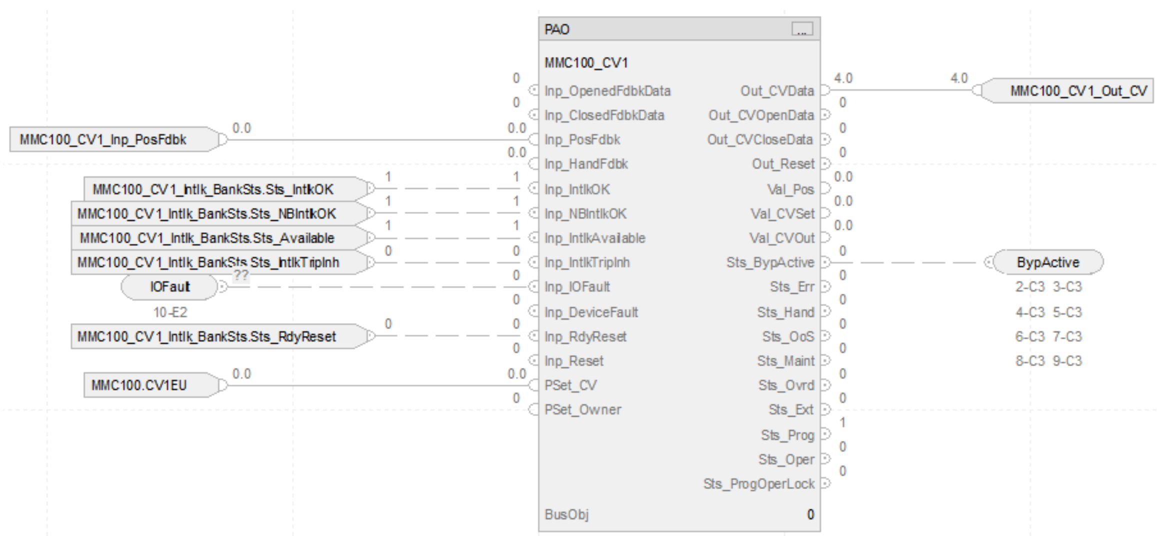
For more information, see HART Integration on page 31.

Control Variable Routines

This control variable routines include the PAO control strategy, with an additional input reference.

- For PAO configuration considerations, and input and output references, see CS_PAO Sheet on page 84 for details.
- The routines also include these PAO input references:

Parameter	Description
MMC100.CV1EU	Scaled control variable output for CV1. Scaled by using CV1EUMax and CV1EUMin, where CV1EUMax corresponds to 100% and CV1EUMin corresponds to 0%. This output is typically used to control an analog output module or a secondary loop. CV1EU = (CV1 * CV1EUSpan / 100) + CV1EUMin CV1EU span calculation: CV1EUSpan = (CV1EUMax - CV1EUMin)
MMC100.CV2EU	Scaled control variable output for CV2. Scaled by using CV2EUMax and CV2EUMin, where CV2EUMax corresponds to 100% and CV2EUMin corresponds to 0%. This output is typically used to control an analog output module or a secondary loop. CV2EU = (CV2 * CV2EUSpan / 100) + CV2EUMin CV2EU span calculation: CV2EUSpan = (CV2EUMax - CV2EUMin)
MMC100.CV3EU	Scaled control variable output for CV3. Scaled by using CV3EUMax and CV3EUMin, where CV3EUMax corresponds to 100% and CV3EUMin corresponds to 0%. This output is typically used to control an analog output module or a secondary loop. CV3EU = (CV3 * CV3EUSpan / 100) + CV3EUMin CV3EU span calculation: CV3EUSpan = (CV3EUMax - CV3EUMin)



Notes:

Process Analog Input (PAI) Control Strategies

Use a PAI control strategy to monitor an analog input and check for alarm conditions. The PAI control strategy that is included with the library download processes a signal from a channel of an analog input module, but it can be used to process any analog (REAL) signal.

The following PAI control strategies are available as routines in the process library:

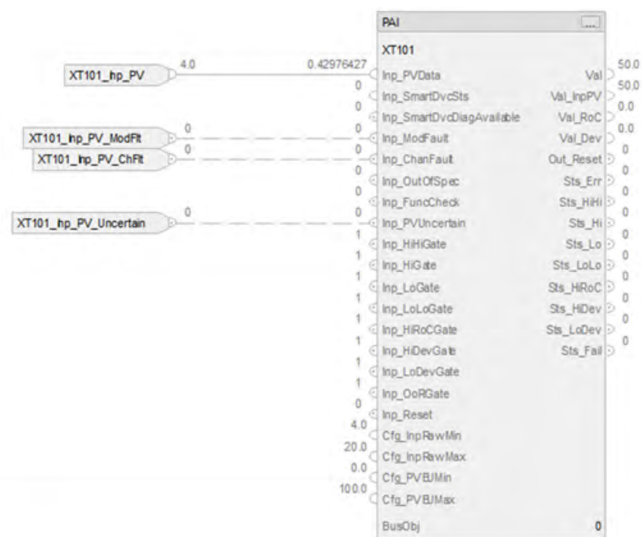
- CS_PAI
- CS_PAI_HART

Import the appropriate control strategy as a **routine** in your controller project.

Each PAI control strategy contains one Function Block sheet:

Sheet	Description
CS_PAI	Process Analog Input instruction
CS_PAI_HART	Process Analog Input instruction with HART input

CS_PAI Sheet



PAI Input References

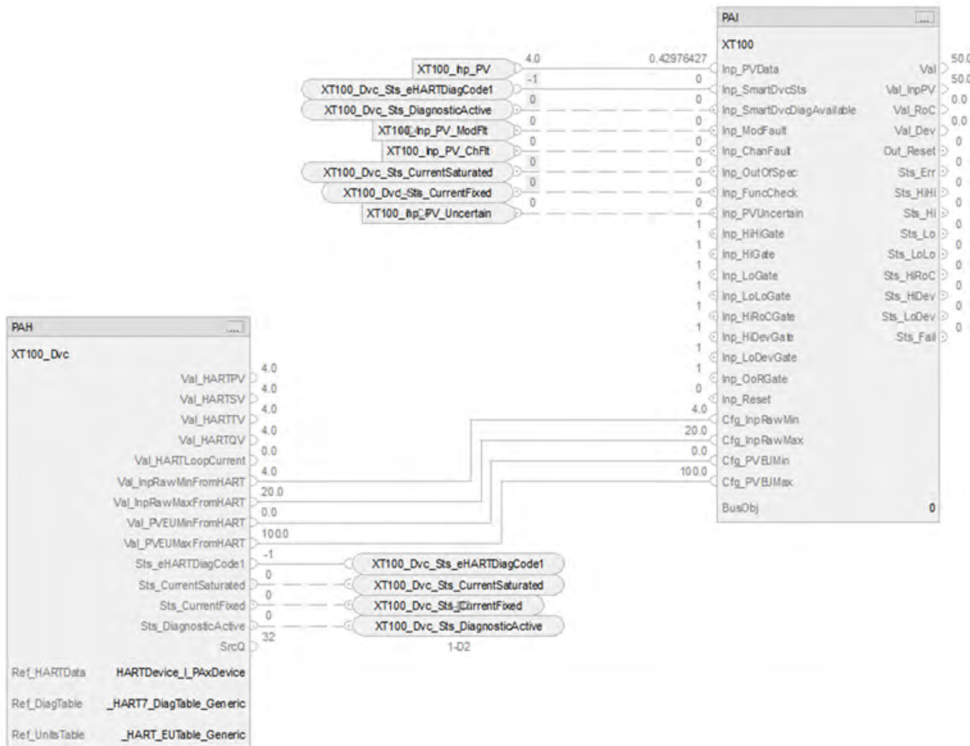
Input	Description
XT101_Inp_PVData	Process variable input Source: sensor or input
XT101_Inp_ModFault	Process variable input module fault 1 = I/O module failure or module communication status bad 0 = OK
XT101_Inp_ChanFault	Process variable input channel fault 1 = I/O channel fault or failure 0 = OK
XT101_Inp_PVUncertain	Process variable input uncertain Indicates the channel data accuracy is undetermined 1 = The channel data is uncertain This input sets Sts_PVUncertain if not in Virtual

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

PAI Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_ANALOG_INPUT	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

CS_PAI_HART Sheet



The CS_PAI_HART control strategy operates the same as the CS_PAI control strategy but relies on HART input data.

- Substitute XT100 for XT101

PAH Outputs to PAI Inputs

Output	Description
Val_InpRawMinFromHART	Analog input unscaled signal minimum from HART module (in module units).
Val_InpRawMaxFromHART	Analog input unscaled signal maximum from HART module (in module units).
Val_PVEUMinFromHART	Analog input scaled range minimum from HART device (in engineering units).
Val_PVEUMaxFromHART	Analog input scaled range maximum from HART device (in engineering units).

PAH Configuration Considerations

Operand	Type	Description
PlantPAx control	P_ANALOG_HART	Instance of data structure (backing tag) required for proper operation of instruction.
Ref_HARTData	PAX_HART_DEVICE:I:0	Required data type HART data from I/O module assembly Select the HART device in your Controller Organizer; the device must support the PAxDevice data type: IOTreeObject:I.PAxDevice
Ref_DiagTable	P_HART_CODE_DESC_STATUS[2]	Lookup table for diagnostic bit number (to message and status) Select the correct table for your HART device; see table below
Ref_UnitsTable	RAC_CODE_DESCRIPTION[2]	Lookup table for units of measure code (to units text) Select _HART_EUTable_Generic

Available Diagnostic Tables

Diagnostic tables are available for these HART devices. The HART number indicates the version of the table.

Option	Description
_HART5_DiagTable_FMG_60	HART Cmd48 Diagnostic Lookup Table: E+H Gammapilot FMG 60
_HART5_DiagTable_Generic	HART Cmd48 Diagnostic Lookup Table: Generic HART5 device
_HART5_DiagTable_LevelflexM	HART Cmd48 Diagnostic Lookup Table: E+H Levelflex M
_HART5_DiagTable_MicropilotM	HART Cmd48 Diagnostic Lookup Table: E+H Micropilot M
_HART5_DiagTable_ProsonicM	HART Cmd48 Diagnostic Lookup Table: E+H Prosonic M
_HART5_DiagTable_ProsonicS	HART Cmd48 Diagnostic Lookup Table: E+H Prosonic S
_HART5_DiagTable_Prowirl73	HART Cmd48 Diagnostic Lookup Table: E+H Prowirl 73
_HART5_DiagTable_TMass65I	HART Cmd48 Diagnostic Lookup Table: E+H TMass 65I
_HART5_DiagTable_TMT162	HART Cmd48 Diagnostic Lookup Table: E+H TMT 162
_HART5_DiagTable_TMT182	HART Cmd48 Diagnostic Lookup Table: E+H TMT182 Temperature
_HART6_DiagTable_Pressure_M	HART Cmd48 Diagnostic Lookup Table: E+H Cerabar, Deltabar, Deltapilot M
_HART7_DiagTable_GammapilotFMG5x_rev1	HART Cmd48 Diagnostic Lookup Table: E+H Gammapilot FMG5x rev 1.x
_HART7_DiagTable_Generic	HART Cmd48 Diagnostic Lookup Table: Generic HART7 device
_HART7_DiagTable_LevelflexFMP5x	HART Cmd48 Diagnostic Lookup Table: E+H Levelflex FMP5x
_HART7_DiagTable_LiquilineCM44x	HART Cmd48 Diagnostic Lookup Table: E+H Liquiline CM442 / 444 / 448
_HART7_DiagTable_LiquilineCM82_rev1	HART Cmd48 Diagnostic Lookup Table: E+H Liquiline CM82 FW Rev. 1
_HART7_DiagTable_LiquilineM_Cond	HART Cmd48 Diagnostic Lookup Table: E+H Liquiline M Conductivity
_HART7_DiagTable_LiquilineM_Cond_rev4	HART Cmd48 Diagnostic Lookup Table: E+H Liquiline M Conductivity rev 4.x
_HART7_DiagTable_LiquilineM_Oxy_rev4	HART Cmd48 Diagnostic Lookup Table: E+H Liquiline M Oxygen rev 4.x
_HART7_DiagTable_LiquilineM_Oxygen	HART Cmd48 Diagnostic Lookup Table: E+H Liquiline M Oxygen
_HART7_DiagTable_LiquilineM_pH_rev4	HART Cmd48 Diagnostic Lookup Table: E+H Liquiline M pH/ORP rev 4.x
_HART7_DiagTable_LiquilineM_pHORP	HART Cmd48 Diagnostic Lookup Table: E+H Liquiline M pH / ORP
_HART7_DiagTable_LiquistationCSFxx	HART Cmd48 Diagnostic Lookup Table: E+H Liquistation CSFxx
_HART7_DiagTable_Metso_ND7x_ND9x	HART Cmd48 Diagnostic Lookup Table: Metso ND7xxx and ND9xxx Positioners
_HART7_DiagTable_MicropilotFMR5x	HART Cmd48 Diagnostic Lookup Table: E+H Micropilot FMR5x
_HART7_DiagTable_MicropilotFMR6x	HART Cmd48 Diagnostic Lookup Table: E+H Micropilot FMR6x
_HART7_DiagTable_MicropilotFMR20	HART Cmd48 Diagnostic Lookup Table: E+H Micropilot FMR 20
_HART7_DiagTable_Pressure_S	HART Cmd48 Diagnostic Lookup Table: E+H Cerabar, Deltabar, Deltapilot S
_HART7_DiagTable_Promag53	HART Cmd48 Diagnostic Lookup Table: E+H Promag 53
_HART7_DiagTable_Promag100	HART Cmd48 Diagnostic Lookup Table: E+H Promag 100
_HART7_DiagTable_Promag200	HART Cmd48 Diagnostic Lookup Table: E+H Promag 200
_HART7_DiagTable_Promag300_500	HART Cmd48 Diagnostic Lookup Table: E+H Promag 300 and Promg 500
_HART7_DiagTable_Promag400	HART Cmd48 Diagnostic Lookup Table: E+H Promag 400
_HART7_DiagTable_Promag400_rev6	HART Cmd48 Diagnostic Lookup Table: E+H Promag 400 rev 6
_HART7_DiagTable_Promag400_rev9	HART Cmd48 Diagnostic Lookup Table: E+H Promag 400 rev 9
_HART7_DiagTable_Promass83	HART Cmd48 Diagnostic Lookup Table: E+H Promass 83

Option	Description
_HART7_DiagTable_Promass100	HART Cmd48 Diagnostic Lookup Table: E+H Promass 100
_HART7_DiagTable_Promass200	HART Cmd48 Diagnostic Lookup Table: E+H Promass 200
_HART7_DiagTable_Promass300_500	HART Cmd48 Diagnostic Lookup Table: E+H Promass 300 and Promass 500
_HART7_DiagTable_ProsonicFlow_100_rev1	HART Cmd48 Diagnostic Lookup Table: E+H Prosonic Flow 100 rev 1.x
_HART7_DiagTable_ProsonicFlow300_500rev1	HART Cmd48 Diagnostic Lookup Table: E+H Prosonic Flow 300 or 500 rev 1.x
_HART7_DiagTable_ProsonicFlowB200	HART Cmd48 Diagnostic Lookup Table: E+H Prosonic Flow B200
_HART7_DiagTable_Prowirl200	HART Cmd48 Diagnostic Lookup Table: E+H Prowirl
_HART7_DiagTable_TMT72_rev1	HART Cmd48 Diagnostic Lookup Table: E+H TMT72 rev 1
_HART7_DiagTable_TMT82	HART Cmd48 Diagnostic Lookup Table: E+H TMT82
_HART7_DiagTable_TMT162_rev4	HART Cmd48 Diagnostic Lookup Table: E+H TMT162 rev 4.x
_HART7_DiagTable_TrustSensTM37x_rev1	HART Cmd48 Diagnostic Lookup Table: E+H TrustSens TM37x rev 1.x

For more information, see HART Integration on page 31.

Process Analog Dual Sensor Input (PAID) Control Strategies

Use a PAID control strategy to provide one analog Process Variable (PV) by using two analog input signals, from sources such as dual sensors, dual transmitters, and dual-input channels. The PAID instruction monitors the conditions of the channels and reports configured PV quality. The PAID instruction has functions for input selection, averaging, and failure detection. Should one of the two upstream PAI signals have bad quality, the PAID continues to provide an output using the remaining good quality signal. If both upstream signals are flagged as bad, the PAID PV is also flagged as bad. Additional functions, such as filtering and alarming, are done by a downstream PAI control strategy.

The following PAID control strategies are available as routines in the process library:

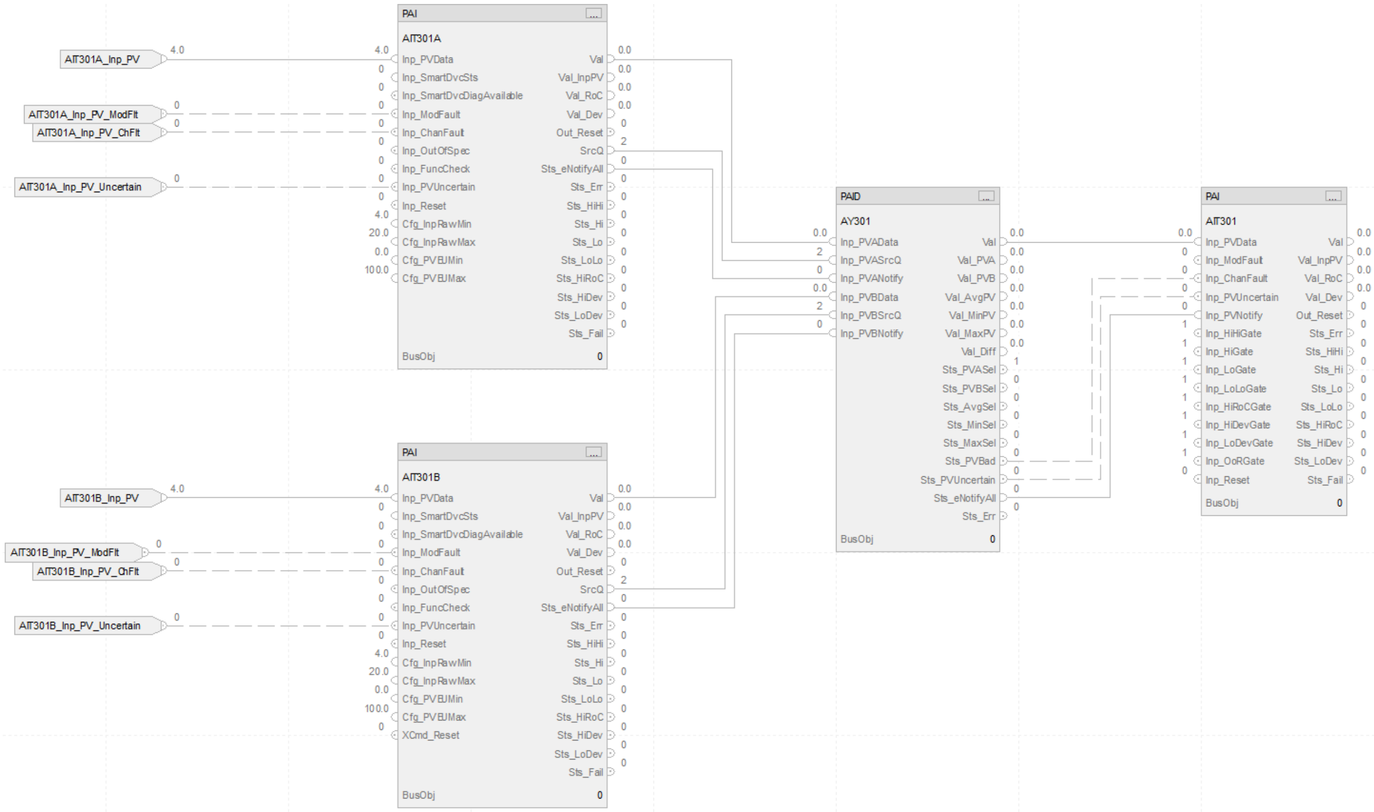
- CS_PAID
- CS_PAID_HART

Import the appropriate control strategy as a **routine** in your controller project.

Each PAID control strategy contains one Function Block sheet:

Sheet	Description
CS_PAID	Process Dual Sensor Analog Input instruction
CS_PAID_HART	Process Dual Sensor Analog Input instruction with HART input

CS_PAID Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.

- Substitute AIT301A for the first instance of XT101
- Substitute AIT301B for the second instance of XT101

PAI Outputs to PAID Inputs

Parameter	Description
Val	Value for PV parameter Process Variable (PVEU)
SrcQ	Value for Inp_PVSrcQ parameter Inp_PV source status and quality: 0 = Good, live, confirmed good 1 = Good, live, assumed good 2 = Good, no feedback, assumed good 8 = Test, simulated 9 = Test, loopback 10 = Test, manually entered 16 = Uncertain, live, off-spec 17 = Uncertain, substituted at device 18 = Uncertain, substituted at instruction 19 = Uncertain, using last known good 20 = Uncertain, using replacement value 32 = Bad, signal failure 33 = Bad, channel fault 34 = Bad, module/communications fault 35 = Bad, invalid configuration
Sts_eNotifyAll	Value for PPID Inp_PVNotify parameter Related PV object alarm priority and acknowledgment status: 0 = Not in alarm, acknowledged 1 = Not in alarm, unacknowledged or reset required 2 = Low severity alarm, acknowledged 3 = Low severity alarm, unacknowledged 4 = Medium severity alarm, acknowledged 5 = Medium severity alarm, unacknowledged 6 = High severity alarm, acknowledged 7 = High severity alarm, unacknowledged 8 = Urgent severity alarm, acknowledged 9 = Urgent severity alarm, unacknowledged

PAID Outputs to PAI Inputs

Parameter	Description
Val	Analog PV, including substitute PV, if used (PV units)
Sts_PVBad	1 = PV bad quality or out of range
Sts_PVUncertain	1 = PV value is uncertain (quality)
Sts_eNotifyAll	Alarm status

PAID Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_ANALOG_INPUT_DUAL	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

CS_PAID_HART Sheet

The CS_PAID_HART control strategy operates the same as the CS_PAID control strategy but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute AIT300A for the first instance of XT100
- Substitute AIT300B for the second instance of XT100
- For more information, see HART Integration on page 31.

Process Analog Input Multi Sensor (PAIM) Control Strategies

Use a PAIM control strategy to provide one analog Process Variable (PV) by using as many as eight analog input signals from sources such as sensors, transmitters, and input channels. The PAIM instruction monitors the conditions of the channels and reports configured PV quality. The PAIM instruction has functions for input selection, averaging, and failure detection. In addition, there is configuration selection for the minimum number of good, unrejected input signals required to have a good PV value, and an alarm if the required number of good inputs is not met. Configure which PV to use if there are only two unrejected signals remaining: the lesser, the greater, or the average of the two. Additional functions, such as for filtering and alarming, are done by a downstream PAI block.

The following PAIM control strategies are available as routines in the process library:

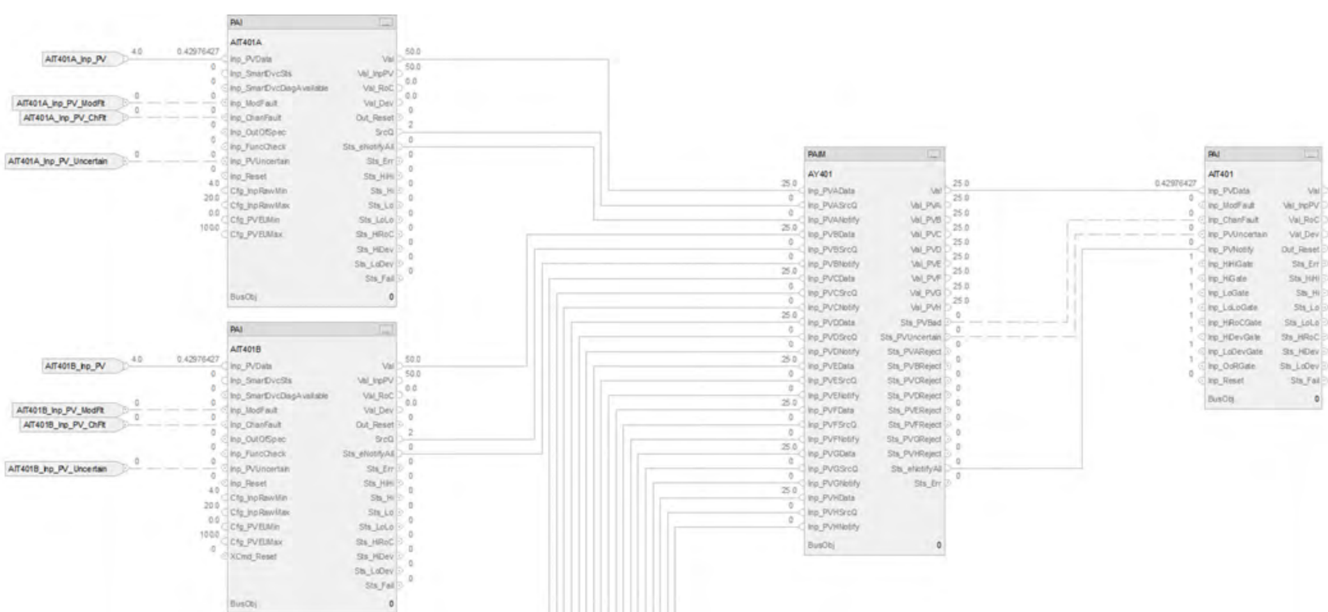
- CS_PAIM
- CS_PAIM_HART

Import the appropriate control strategy as a **routine** in your controller project.

Each PAIM control strategy contains one Function Block sheet:

Sheet	Description
CS_PAIM	Process Multi Sensor Dual Sensor Analog Input instruction
CS_PAIM_HART	Process Multi Sensor Dual Sensor Analog Input instruction with HART input

CS_PAIM Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.

- Substitute AIT401x for each instance of XT101

PAI Outputs to PAIM Inputs

Parameter	Description
Val	Value for PV parameter Process Variable (PVEU)
SrcQ	Value for Inp_PVSrcQ parameter Inp_PV source status and quality: 0 = Good, live, confirmed good 1 = Good, live, assumed good 2 = Good, no feedback, assumed good 8 = Test, simulated 9 = Test, loopback 10 = Test, manually entered 16 = Uncertain, live, off-spec 17 = Uncertain, substituted at device 18 = Uncertain, substituted at instruction 19 = Uncertain, using last known good 20 = Uncertain, using replacement value 32 = Bad, signal failure 33 = Bad, channel fault 34 = Bad, module/communications fault 35 = Bad, invalid configuration
Sts_eNotifyAll	Value for PPID Inp_PVNotify parameter Related PV object alarm priority and acknowledgment status: 0 = Not in alarm, acknowledged 1 = Not in alarm, unacknowledged or reset required 2 = Low severity alarm, acknowledged 3 = Low severity alarm, unacknowledged 4 = Medium severity alarm, acknowledged 5 = Medium severity alarm, unacknowledged 6 = High severity alarm, acknowledged 7 = High severity alarm, unacknowledged 8 = Urgent severity alarm, acknowledged 9 = Urgent severity alarm, unacknowledged

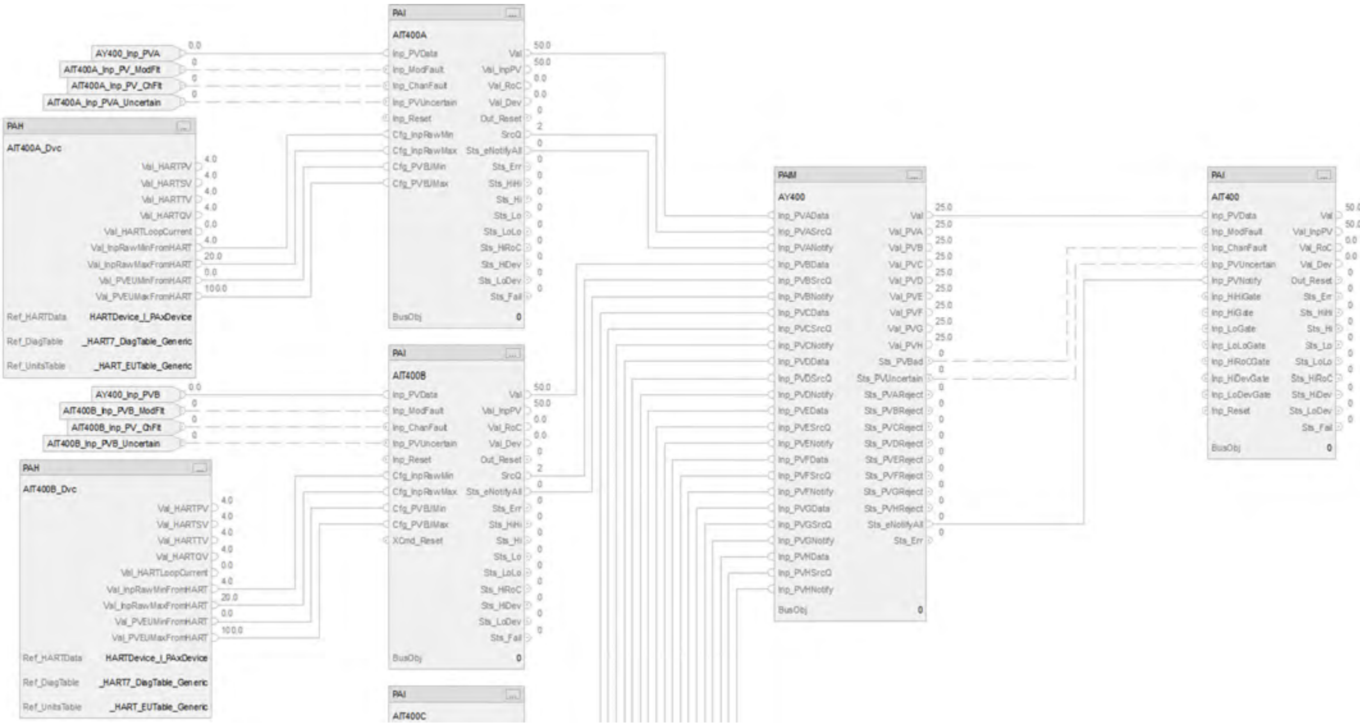
PAIM Outputs to PAI Inputs

Parameter	Description
Val	Analog PV, including substitute PV, if used (PV units)
Sts_PVBad	1 = PV bad quality or out of range
Sts_PVUncertain	1 = PV value is uncertain (quality)
Sts_eNotifyAll	Alarm status

PAIM Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_ANALOG_INPUT_MULTI	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

CS_PAIM_HART Sheet



The CS_PAIM_HART control strategy operates the same as the CS_PAIM control strategy but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute AIT400x for each instance of XT100
- For more information, see HART Integration on page 31.

Notes:

Process Analog Output (PAO) Control Strategies

Use a PAO control strategy to drive an analog field device to a reference value. The reference value can be entered by operator entry at the HMI or by a program input. The control strategies use a single output to an analog output channel to drive the field device. Optional opened and closed limit indications can be configured requiring additional digital input(s).

Alternatively, use the PAO control strategies to position the field device by using two digital output pulses (one to pulse open and another to pulse close). Pulsed outputs to position the field device require two digital output channels to position the device, as well as an additional analog channel to represent the current field device position. Digital positioning also requires additional configuration in the PAO instruction for the pulse timing.

In addition to positioning a field device based on program or HMI entries, these control strategies provide the ability to position (shed) the device based on IO fault status and interlock conditions.

The following PAO control strategies are available as routines in the process library:

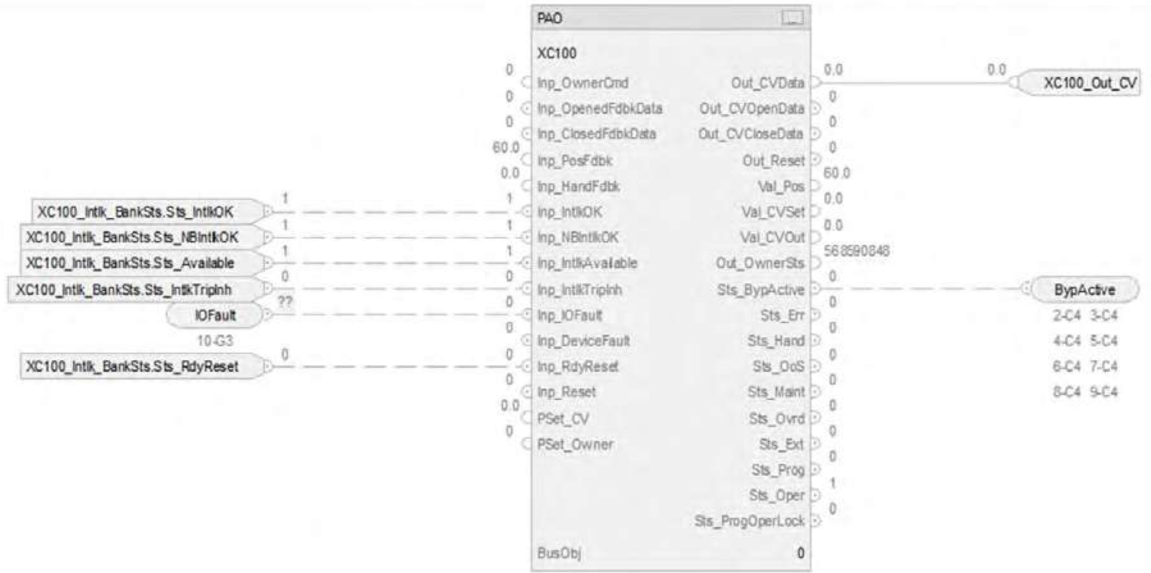
- CS_PAO (with interlocks)
- CS_PAO_noIntlk (without interlocks)

Import the appropriate control strategy as a **routine** in your controller project.

The PAO control strategies contain these Function Block sheets:

Sheet	Description
CS_PAO	Process Analog Output instruction
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	Only in CS_PAO. The PAO instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder
IO Faults	The logic monitors one analog output channel for I/O fault input and raises alarm on an I/O fault.

CS_PAO Sheet



PAO Input References

Parameter	Description
XC100_Inp_PosFdbk	Tieback input. Feedback from actual device position PV (CV engineering units). Valid any float. Default is 0.0.
XC100_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
XC100_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
XC100_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
XC100_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
IOFault	Input connection from IO Faults sheet
XC100_Intlk_BankSts.Sts_RdyReset	Interlock bank status 1 = A latched interlock (returned to OK) is ready to be reset

PAO Output References

Parameter	Description
XC100_Out_CVData	Control Variable output CV output in raw (I/O Card) units. Extended properties of this member: Engineering Unit - Raw units (text) used for the analog output Destination: Analog output channel or downstream REAL tag
BypActive	Output connection to interlock bank sheet

For a HART analog output, include these outputs:

HART Parameter	Description
XC101_Val_CVOut	Value of CV Output after optional rate limiting, in engineering units. Extended Properties of this member: Engineering Unit - Engineering units (text) used for the analog output Destination: Analog output channel or downstream REAL tag
XC101_Sts_Available	1 = Analog output available for control by automation (Program)

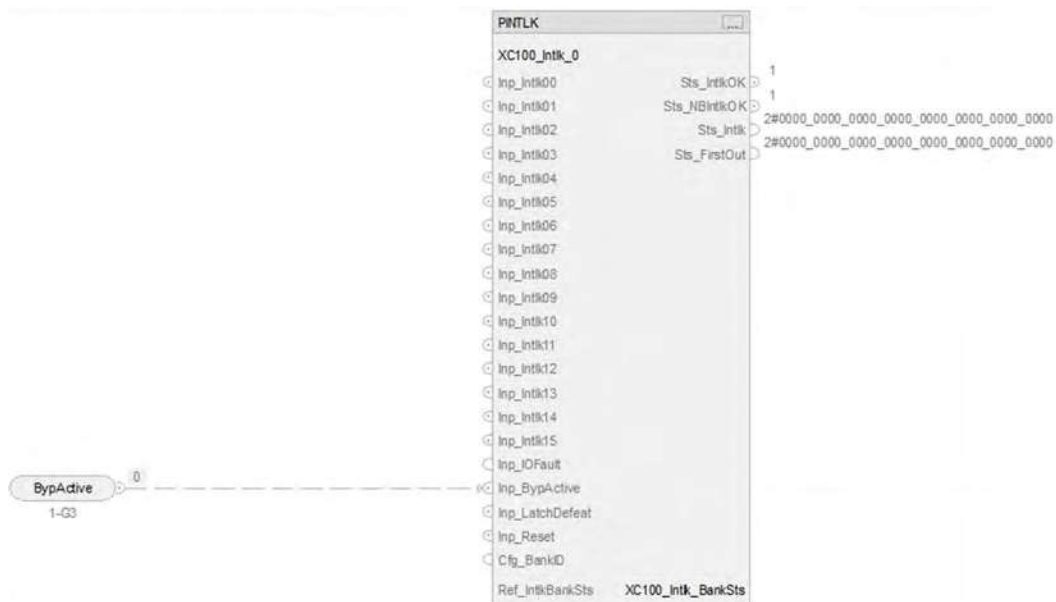
PAO Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_ANALOG_OUTPUT	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

If you use digital output pulses, configure these PAO **Parameter**:

Parameter	Description
Cfg_HasPulseOut	Enables the pulse1 = Device provides pulse output (open, close).
Cfg_HasOpenedFdbk	1 = Use device opened feedback for failure checking.
Cfg_HasCloseFdbk	1 = Device provides closed feedback signal.

Interlock Bank Sheet



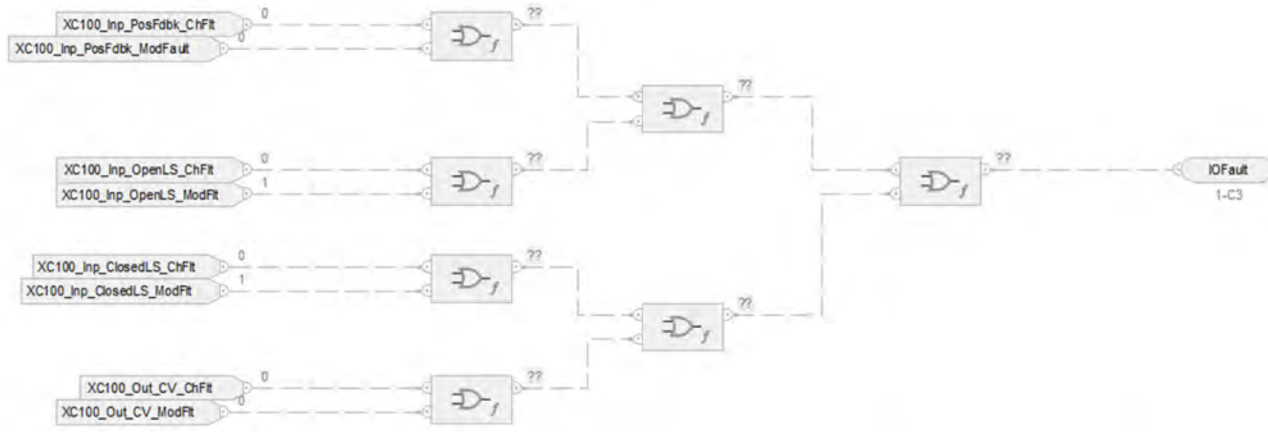
Input Reference to PINTLK

Parameter	Description
BypActive	Input connection from CS_PAO sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

IO Faults Sheet



Fault Input References

Parameter	Description
XC100_Inp_PosFdbk_ChanFault	Tieback input channel fault
XC100_Inp_PosFdbk_ModFault	Tieback input module fault
XC100_Inp_OpenLS_ChFlt	Open limit switch channel fault
XC100_Inp_OpenLS_ModFlt	Open limit switch module fault
XC100_Inp_ClosedLS_ChFlt	Closed limit switch channel fault
XC100_Inp_ClosedLS_ModFlt	Closed limit switch module fault
XC100_Out_CVData_ChanFault	Control Variable output channel fault
XC100_Out_CVData_ModFault	Control Variable output data module fault

Fault Output References

Parameter	Description
IOFault	Output connection to CS_PAO sheet

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

Process Boolean Logic (PBL) Control Strategy

Use the PBL control strategy to process as many as four digital inputs by applying as many as eight gates of configurable logic. Gate types available include AND, OR, XOR (Exclusive-OR), Set/Reset, Select, and Majority. A benefit of the PBL control strategy is that assembly of the logical gates is done from the HMI, which helps to make sure that the HMI representation is accurate with respect to the underlying logic.

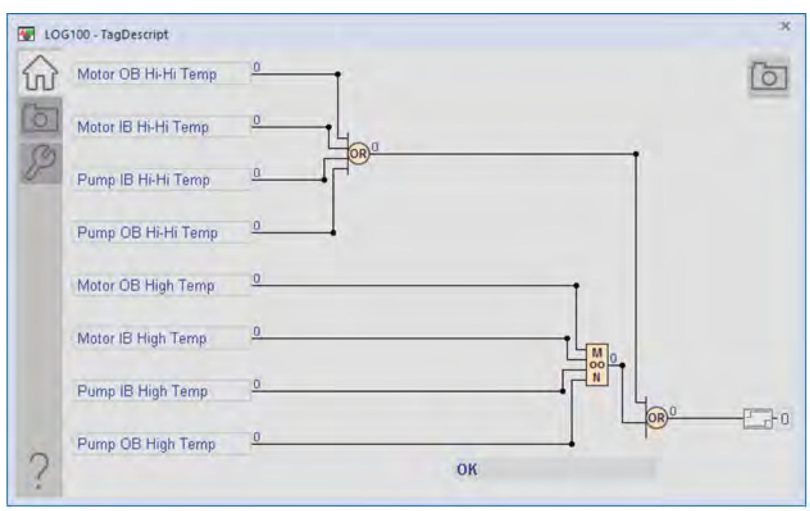
The PBL instruction can record its current state (via snapshot of current graphical representation):

- After a change in output state
- On Operator or Program command
- Based on a logic loopback input

Use the PBL instruction in these situations:

- A project requires an Interlock or Permissive condition that is more complicated than the simple OR-ing or AND-ing provided by the PINTLK (Interlocks) or PPERM (Permissives) Add-On Instructions.
- A project requires some Boolean (combination) logic that can be reconfigured from the HMI online, or which requires the snapshot capability for saving a copy of the logic state with a time stamp.
- A project contains more than the 16 interlock conditions or permissive conditions provided by the PINTLK and PPERM Add-On Instructions, but some of the conditions can be grouped together under one identification. For example, all bearing over-temperature signals for a pump and motor (Pump Inboard Bearing, Pump Outboard Bearing, Motor Inboard Bearing, and Motor Outboard Bearing) can be ORed together in a PBL instruction and the result presented to a PINTLK instruction as a single Bearing Overtemp condition.

The PBL logic is typically configured from an HMI display.



The CS_PBL control strategy is available as a routine in the process library. Import the control strategy as a **routine** in your controller project. The PBL control strategy contains one CS_PBL Function Block sheet.

CS_PBL Sheet

PBL Input References

Parameter	Description
Inp_0	Logic inputs
Inp_1	
Inp_2	
Inp_3	
Inp_4	
Inp_5	
Inp_6	
Inp_7	

PBL Output Reference

Parameter	Description
LOG100_Out_Live	Condition logic output (result) after delay.

PBL Configuration Considerations

Operand	Type	Description
PBL tag	P_BOOLEAN_LOGIC	Instance of data structure (backing tag) required for proper operation of instruction

Process Discrete 2-, 3-, or 4-State Device (PD4SD) Control Strategy

The Process Discrete 2-, 3-, 4-state Device Add-On Instruction controls and monitors feedback (using up to four discrete outputs and up to four discrete feedbacks) from a discrete 2-state, 3-state, or 4-state device in various modes while monitoring for fault conditions. These devices include multiple-speed motors or multiple-position valves.

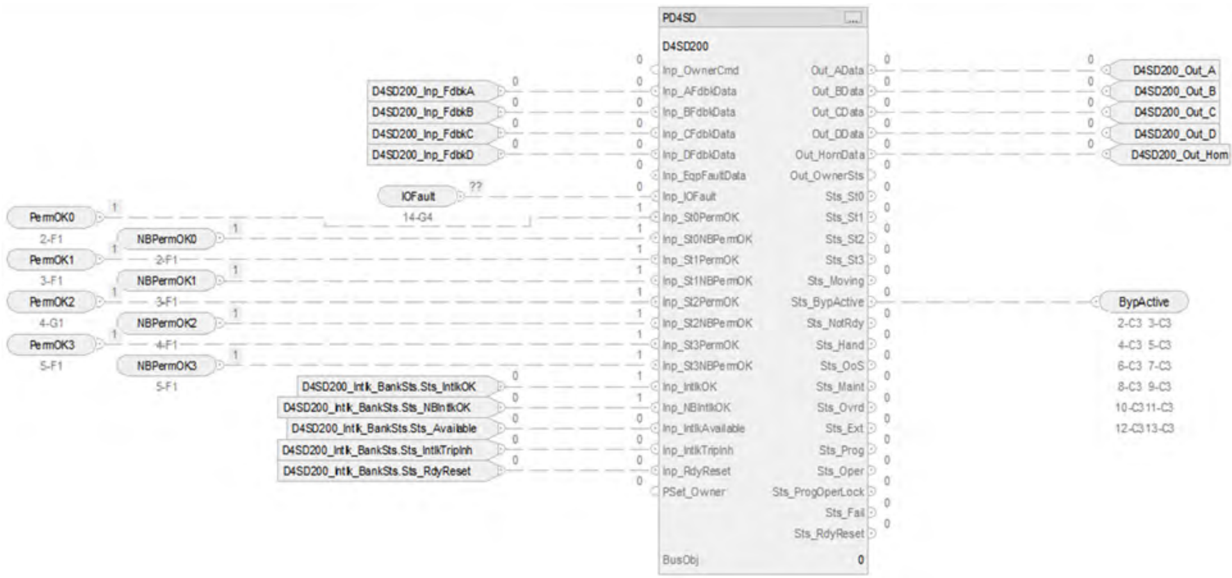
The CS_PD4SD control strategy is available as a routine in the process library.

Import the control strategy as a **routine** in your controller project.

The PD4SD control strategy contains these Function Block sheets:

Sheet	Description
CS_PD4SD	Discrete State Device Add-On Instruction
Permissives 0 Permissives 1 Permissives 2 Permissives 3	Process Permissives instruction The Process Permissives (PPERM) instruction collects, or sums up, the permissive conditions that let a piece of equipment energize. In most cases, permissive conditions must be true to energize equipment. Once the equipment is energized, permissives are ignored.
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder.
IO Fault	The logic monitors as many as four discrete input channels and as many as five discrete output channels for I/O fault input and raises an alarm on an I/O fault.

CS_PD4SD Sheet



PD4SD Input References

Parameter	Description
D4SD200_Inp_FdbkA	Input Signal: Feedback A from device.
D4SD200_Inp_FdbkB	Input Signal: Feedback B from device.
D4SD200_Inp_FdbkC	Input Signal: Feedback C from device.
D4SD200_Inp_FdbkD	Input Signal: Feedback D from device.
IOFault	Input connection from IO Faults sheet
PermOK0	Input connection from Permissives sheet 0 (State 1) 1 = On permissives OK, device can turn On
NBPermOK0	Input connection from Permissives sheet 0 (State 1) 1 = Non-bypassable On permissives OK, device can turn On
PermOK1	Input connection from Permissives sheet 1 (State 2) 1 = On permissives OK, device can turn On
NBPermOK1	Input connection from Permissives sheet 1 (State 2) 1 = Non-bypassable On permissives OK, device can turn On
PermOK2	Input connection from Permissives sheet 2 (State 3) 1 = On permissives OK, device can turn On
NBPermOK2	Input connection from Permissives sheet 2 (State 3) 1 = Non-bypassable On permissives OK, device can turn On
PermOK3	Input connection from Permissives sheet 3 (State 4) 1 = On permissives OK, device can turn On
NBPermOK3	Input connection from Permissives sheet 3 (State 4) 1 = Non-bypassable On permissives OK, device can turn On
D4SD200_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
D4SD200_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
D4SD200_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
D4SD200_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
D4SD200_Intlk_BankSts.Sts_RdyReset	Interlock bank status 1 = A latched interlock (returned to OK) is ready to be reset

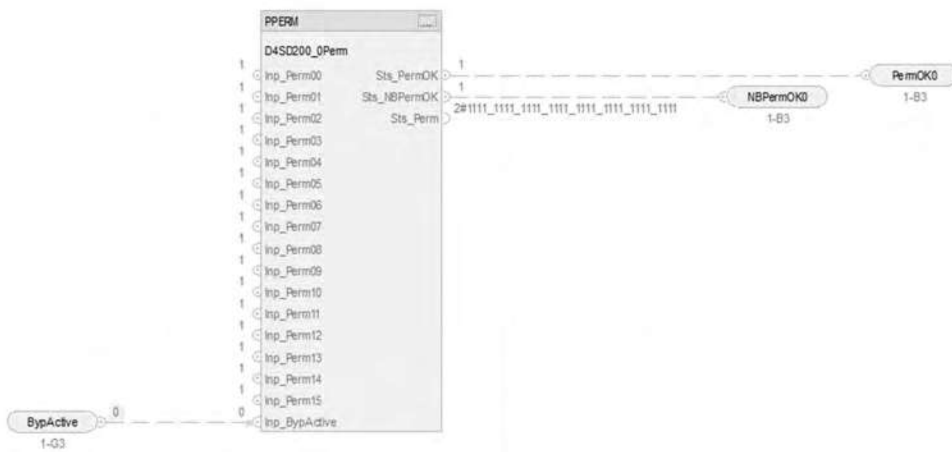
PD4SD Output References

Parameter	Description
D4SD200_Out_A	Output A to device
D4SD200_Out_B	Output B to device
D4SD200_Out_C	Output C to device
D4SD200_Out_D	Output D to device
D4SD200_Out_Horn	1 = Sound audible before commanded state change
D4SD200_Out_Reset	1 = Reset command has been received and accepted
BypActive	Output connection to permissives and interlock bank sheets

PD4SD Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_DISCRETE_4STATE	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

Permissive Sheet



PPERM Input References

Parameter	Description
BypActive	Input connection from the CS_PD4SD sheet

PPERM Output References

Parameter	Description
PermOK0 PermOK1 PermOK2 PermOK3	Overall permissive status (1 = OK to energize)
NBPermOK0 NBPermOK1 NBPermOK2 NBPermOK3	Non-bypassable permissive status (1 = all non-bypassable permissives OK to energize)

Interlock Bank Sheet

PINTLK Input Reference

Parameter	Description
BypActive	Input connection from the CS_PD4SD sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction DS4D100 in this example corresponds to a state device
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

IO Fault Sheet

Fault Input References

Parameter	Description
D4SD200_Inp_FdkA_ChFlt D4SD200_Inp_FdkB_ChFlt D4SD200_Inp_FdkC_ChFlt D4SD200_Inp_FdkD_ChFlt	Tieback input channel faults
D4SD200_Inp_FdkA _ModFault D4SD200_Inp_FdkB _ModFault D4SD200_Inp_FdkC _ModFault D4SD200_Inp_FdkD _ModFault	Tieback input module faults
D4SD200_Out_A_ChFlt D4SD200_Out_B_ChFlt D4SD200_Out_C_ChFlt D4SD200_Out_D_ChFlt	Output channel faults
D4SD200_Out_A_ModFlt D4SD200_Out_B_ModFlt D4SD200_Out_B_ModFlt D4SD200_Out_B_ModFlt	Output module faults
D4SD200_Out_Horn_ChFlt	Sound audible for output channel fault
D4SD200_Out_Horn_ModFlt	Sound audible for output module fault

Fault Output References

Parameter	Description
IOFault	Output connection to CS_PD4SD sheet

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

Notes:

Process Deadband Controller (PDBC) Control Strategy

Use the PDBC control strategy to maintain a PV within a deadband of the SP by triggering one or two digital outputs (a raise output and a lower output).

The following PDBC control strategies are available as routines in the process library:

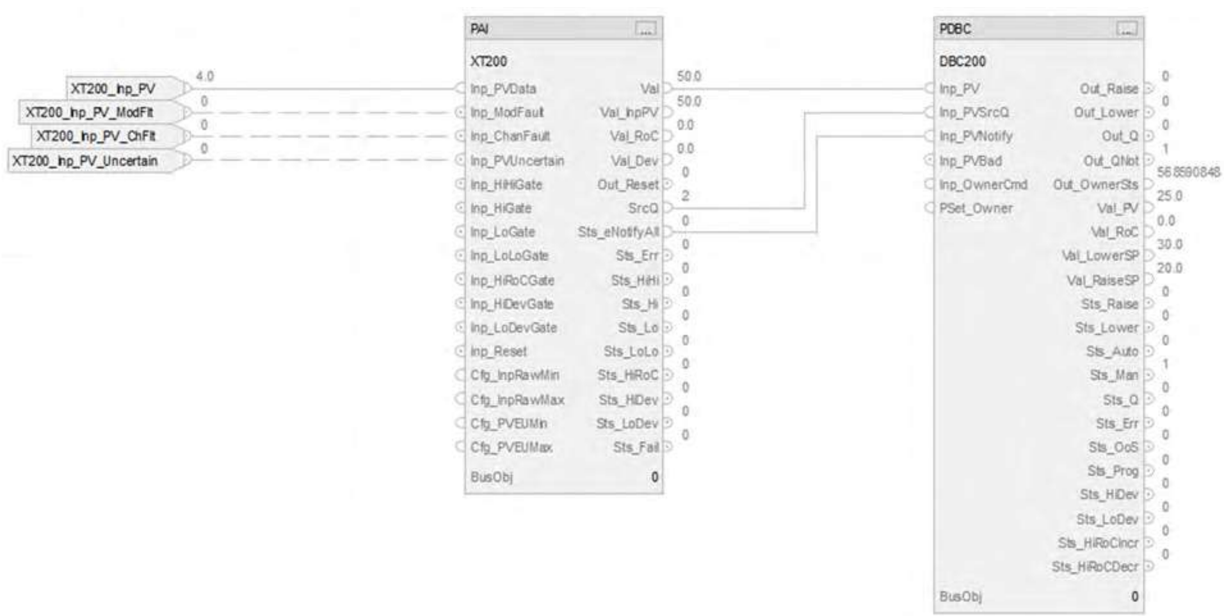
- CS_PDBC
- CS_PDBC_HART

Import the appropriate control strategy as a **routine** in your controller project.

The PDBC control strategy contains one Function Block sheet:

Sheet	Description
CS_PDBC	Process Deadband Controller instruction
CS_PDBC_HART	Process Deadband Controller with HART input

CS_PDBC Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.

- Substitute XT200 the instances of XT101

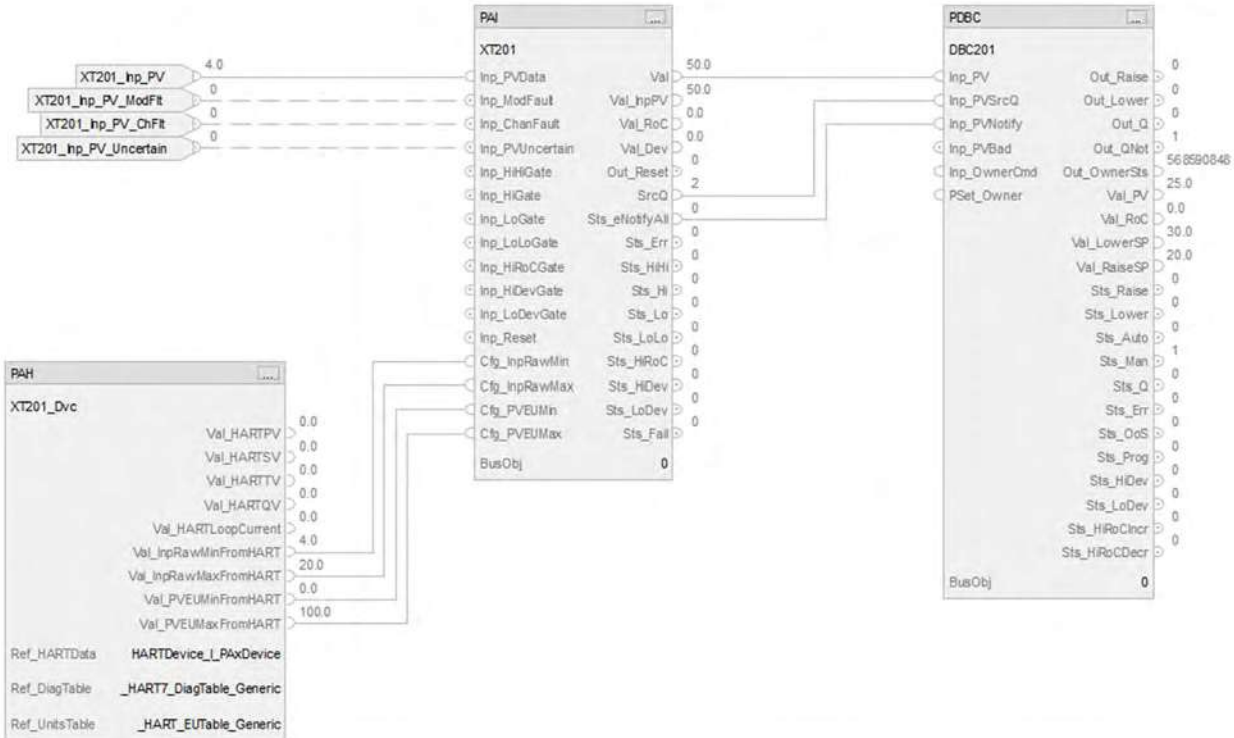
PAI Outputs to PDBC Inputs

Parameter	Description
Val	Value for PV parameter Process Variable (PVEU)
SrcQ	Value for PDBC Inp_PVSrcQ parameter Inp_PV source status and quality: 0 = Good, live, confirmed good 1 = Good, live, assumed good 2 = Good, no feedback, assumed good 8 = Test, simulated 9 = Test, loopback 10 = Test, manually entered 16 = Uncertain, live, off-spec 17 = Uncertain, substituted at device 18 = Uncertain, substituted at instruction 19 = Uncertain, using last known good 20 = Uncertain, using replacement value 32 = Bad, signal failure 33 = Bad, channel fault 34 = Bad, module/communications fault 35 = Bad, invalid configuration
Sts_eNotifyAll	Value for PDBC Inp_PVNotify parameter Related PV object alarm priority and acknowledgment status: 0 = Not in alarm, acknowledged 1 = Not in alarm, unacknowledged or reset required 2 = Low severity alarm, acknowledged 3 = Low severity alarm, unacknowledged 4 = Medium severity alarm, acknowledged 5 = Medium severity alarm, unacknowledged 6 = High severity alarm, acknowledged 7 = High severity alarm, unacknowledged 8 = Urgent severity alarm, acknowledged 9 = Urgent severity alarm, unacknowledged

PDBC Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_DEADBAND	Instance of data structure (backing tag) required for proper operation of instruction.
BusObj	BUS_OBJ	Bus component for organization control 0 if not using organization Bus[x].Obj when using organization. See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

CS_PDBC_HART Sheet



The CS_PDBC_HART control strategy operates the same as the CS_PDBC control strategy but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute DBC201 for the PV data instance of XT100
- Substitute XT201 for the remaining instances of XT100

For more information, see HART Integration on page 31.

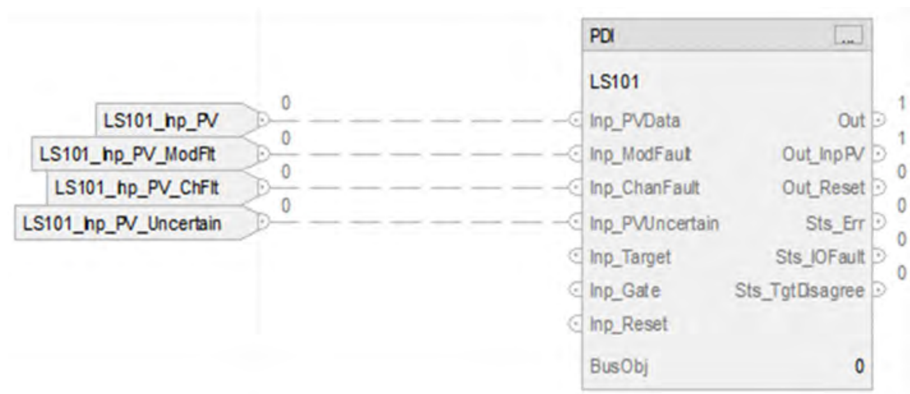
Notes:

Process Discrete Input (PDI) Control Strategy

Use the PDI control strategy to monitor a discrete (true or false) input and check for alarm conditions. The PDI instruction processes a signal from a channel of a discrete input module. Use the PDI instruction with any discrete (BOOL) signal.

The CS_PDI control strategy is available as a routine in the process library. Import the control strategy as a **routine** in your controller project. The PDI control strategy contains one CS_PDI Function Block sheet.

CS_PDI Sheet



PDI Input References

Parameter	Description
LS101_Inp_PVData	Process variable input Source: sensor or input
LS101_Inp_PV_ModFlt	Process variable input module fault 1 = I/O module failure or module communication status bad 0 = OK
LS101_Inp_PV_ChFlt	Process variable input channel fault 1 = I/O channel fault or failure 0 = OK
LS101_Inp_PV_Uncertain	Process variable input uncertain Indicates the channel data accuracy is undetermined 1 = The channel data is uncertain This input sets Sts_PVUncertain if not in Virtual

For examples on how to map device input tags to the Inp_PVData, Inp_ModFault, Inp_ChanFault, and Inp_PVUnceratin references, see: PlantPAx Control Strategies on page 17.

PDI Configuration Considerations

Operand	Type	Description
PDI tag	P_DISCRETE_INPUT	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

Notes:

Process Discrete Output (PDO) Control Strategies

Use a PDO control strategy to drive a discrete (true/false) output, monitor discrete inputs serving as feedbacks from a device driven by the discrete output, and check for alarm conditions. Use the PDO instruction for a channel of a discrete output module. Use the PDO instruction with any discrete (BOOL) signal.

The PDO instruction operates in a variety of modes, and can provide steady, single pulsed, or continually pulsed output.

The following PDO control strategies are available as routines in the process library:

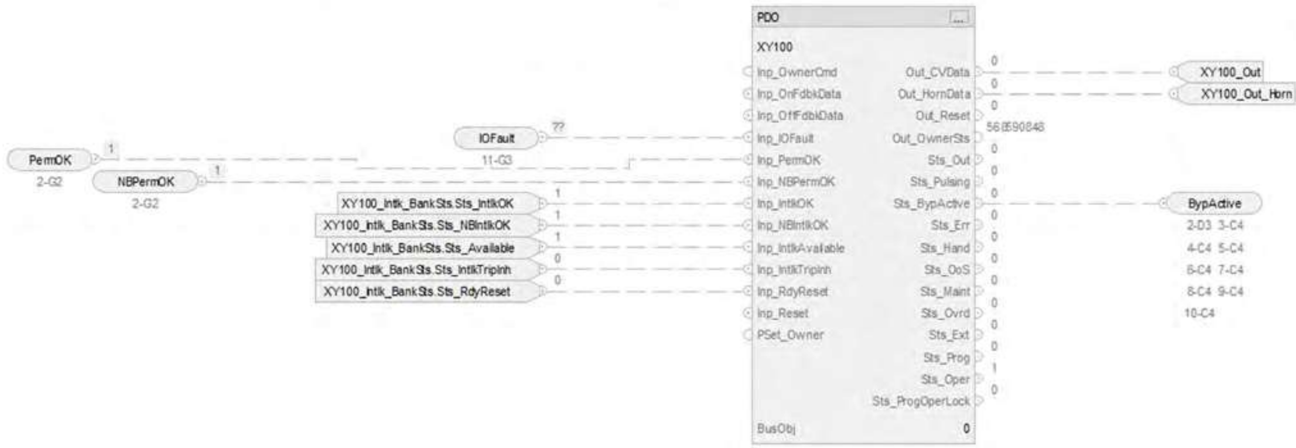
- CS_PDO (with interlocks)
- CS_PDO_noIntlk (without interlocks)

Import the appropriate control strategy as a **routine** in your controller project.

The PDO control strategies contain these Function Block sheets:

Sheet	Description
CS_PDO	Process Discrete Output instruction
Permissive	Process Permissives instruction The Process Permissives (PPERM) instruction collects, or sums up, the permissive conditions that let a piece of equipment energize. In most cases, permissive conditions must be true to energize equipment. Once the equipment is energized, permissives are ignored.
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	Only in CS_PDO The PDO instruction monitors bypassable and non-bypassable Interlocks that force the output to the configured safe state. There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder
IO Faults	The logic monitors the input and output modules and channels used to interface with the device for fault conditions and raises an alarm on an I/O fault.

CS_PDO Sheet



PDO Input References

Parameter	Description
IOFault	Input connection from the IO Faults sheet
PermOK	Input connection from Permissive sheet 1 = On permissives OK, device can turn On
NBPermOK	Input connection from Permissive sheet 1 = Non-bypassable On permissives OK, device can turn On
XY100_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
XY100_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
XY100_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
XY100_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
XY100_Intlk_BankSts.Sts_RdyReset	Interlock bank status 1 = A latched interlock (returned to OK) is ready to be reset

PDO Output References

Parameter	Description
XY100_Out_CVData	Control Variable output CV output in raw (I/O Card) units. Extended properties of this member: Engineering Unit - Raw units (text) used for the analog output
XY100_Out_Horn	1 = Sound audible prior to commanded state change
BypActive	Output connection to permissives and interlock bank sheets

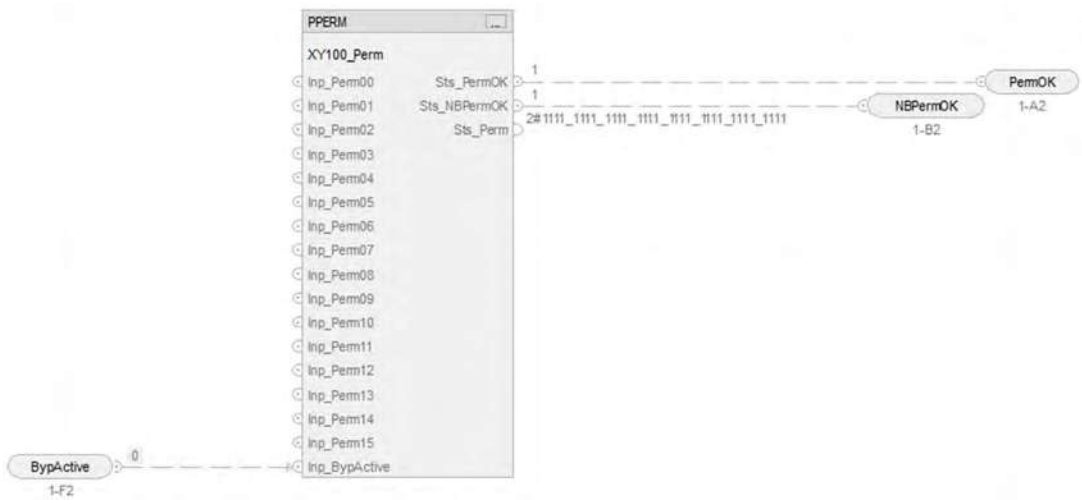
PDO Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_DISCRETE_OUTPUT	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

If you use digital input pulses, configure these PDO operands:

Parameter	Description
Cfg_HasPulseOut	1 = Enable pulsing functions
Cfg_HasOnFdbk	1 = Device provides an On feedback signal
Cfg_HasOffFdbk	1 = Device provides an Off feedback signal

Permissive Sheet



PPERM Input References

Parameter	Description
BypActive	Input connection from the interlock bank sheet

PPERM Output References

Parameter	Description
PermOK	Overall permissive status (1 = OK to energize)
NBPermOK	Non-bypassable permissive status (1 = all non-bypassable permissives OK to energize)

Interlock Bank Sheet

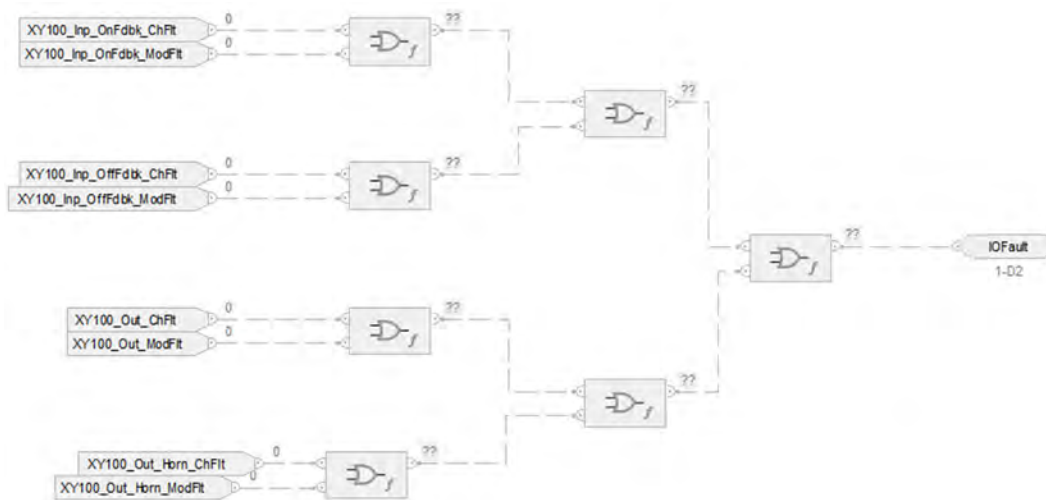
PINTLK Input Reference

Parameter	Description
BypActive	Input connection from CS_PDO sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

IO Faults Sheet



Fault Input References

Parameter	Description
XY100_Inp_OnFdbk_ChFlt	On feedback channel fault
XY100_Inp_OnFdbk_ModFlt	On feedback module fault
XY100_Inp_OffFdbk_ChFlt	Off feedback channel fault
XY100_Inp_OffFdbk_ModFlt	Off feedback module fault
XY100_Out_ChFlt	Output channel fault
XY100_Out_ModFlt	Output module fault
XY100_Out_Horn_ChFlt	Output horn channel fault
XY100_Out_Horn_ModFlt	Output horn module fault

Parameter	Description
IOFault	Output connection to CS_PDO sheet

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17

Process Dosing Flow Meter (PDOSEFM) Control Strategy

Use the PDOSEFM control strategy to control an ingredient addition that uses a flow meter to measure the quantity of ingredient added. The flow meter can be an analog flow meter (signal proportional to flow), a pulse generating flow meter (pulse count proportional to quantity delivered), or a digital flow meter providing flow rate or quantity (totalized flow) information.

The following PDOSEFM control strategies are available as routines in the process library:

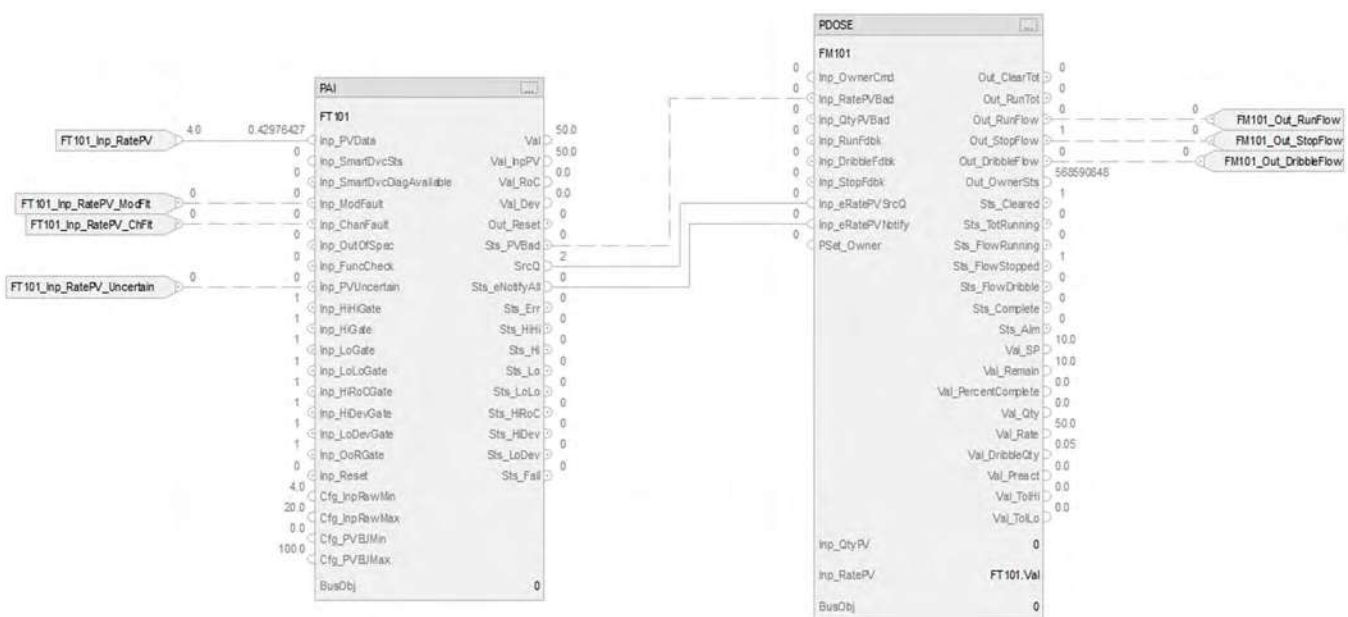
- CS_PDOSEFM
- CS_PDOSEFM_HART

Import the appropriate control strategy as a **routine** in your controller project.

The PDOSEFM control strategy contains one Function Block sheet:

Sheet	Description
CS_PDOSEFM	Process Dosing Flow Meter instruction
CS_PDOSEFM_HART	Process Dosing Flow Meter instruction with HART input

CS_PDOSEFM Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.

- Substitute FM101 for the PV data instance of XT101
- Substitute FT101 for the remaining instances XT101

PAI Outputs to PDOSE Inputs

Parameter	Description
Sts_PVBad	Quality of PV value 1 = PV quality is flagged as Bad
SrcQ	Source and quality of primary value or status: 0 = Good, live, confirmed good 1 = Good, live, assumed good 2 = Good, no feedback, assumed good 8 = Test, simulated 9 = Test, loopback 10 = Test, manually entered 16 = Uncertain, live, off-spec 17 = Uncertain, substituted at device 18 = Uncertain, substituted at instruction 19 = Uncertain, using last known good 20 = Uncertain, using replacement value 32 = Bad, signal failure 33 = Bad, channel fault 34 = Bad, module/communications fault 35 = Bad, invalid configuration
Sts_eNotifyAll	All alarm status enumerated values including related objects: 0 = Not in alarm, acknowledged 1 = Not in alarm, unacknowledged, or unacknowledged or reset required 2 = Low severity alarm, acknowledged 3 = Low severity alarm, unacknowledged 4 = Medium severity alarm, acknowledged 5 = Medium severity alarm, unacknowledged 6 = High severity alarm, acknowledged 7 = High severity alarm, unacknowledged 8 = Urgent severity alarm, acknowledged 9 = Urgent severity alarm, unacknowledged

PDOSE Output References

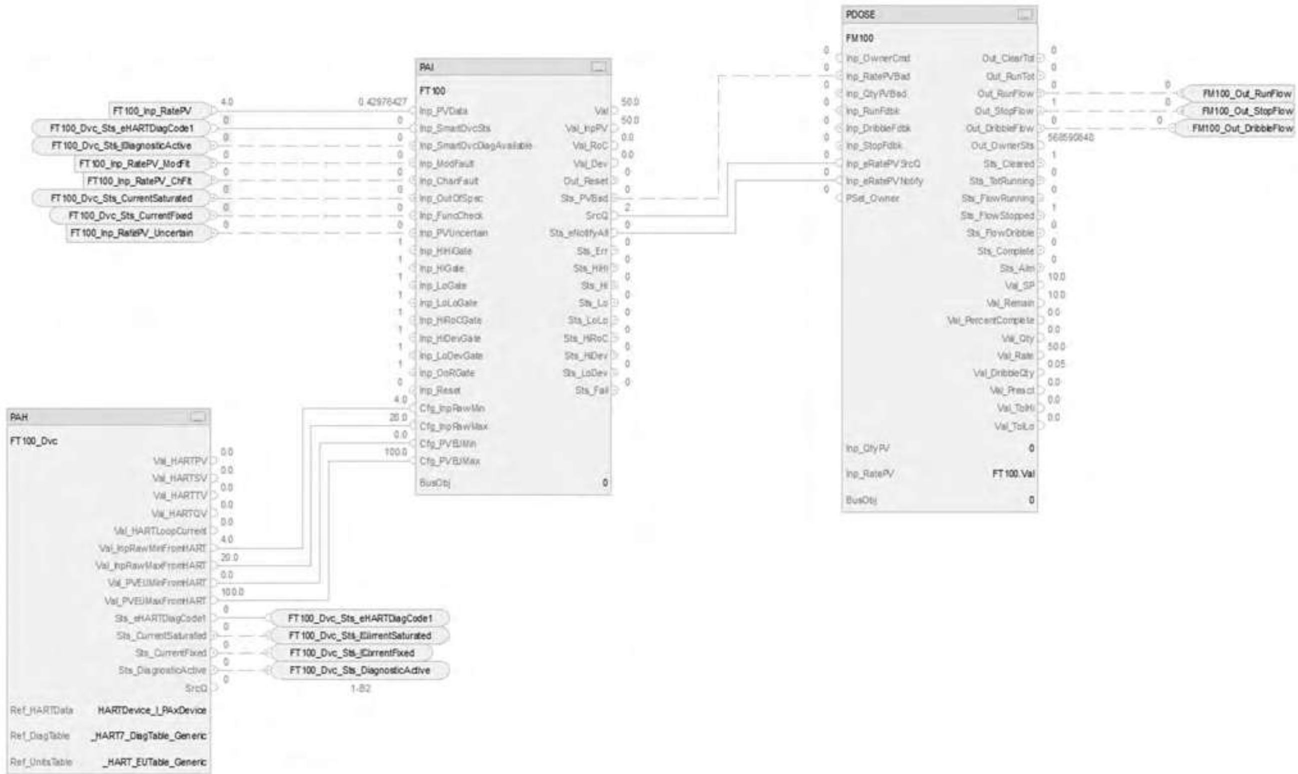
Parameter	Description
FM101_Out_RunFlow	1 = Deliver at full (fast) flow
FM101_Out_StopFlow	1 = Stop delivery equipment
FM101_Out_DribbleFlow	1 = Deliver at dribble (slow) flow

PDOSE Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_DOSING	Instance of data structure (backing tag) required for proper operation of instruction
Inp_QtyPV	REAL	Quantity from flowmeter (EU or pulse count). Input is disabled if Sts_CalcQty is either of the following: • 1=integrate Inp_RatePV to get quantity • 0=use Inp_QtyPV
Inp_RatePV	REAL	Flow rate from flowmeter (EU/Time, see Cfg_RateTime). Input is disabled if Sts_CalcRate is either of the following: • 1=differentiate Inp_QtyPV to get rate • 0=use Inp_RatePV
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

For a flowmeter, you usually use the rate input. If the flowmeter provides a rate and a totalized quantity, use both the rate and quantity parameters. When both parameters are connected, the instruction uses the meter's quantity and does not need to calculate a quantity from the rate. Connect the clear totalizer output back to the meter to reset the totalizer as needed.

CS_PDOSEFM_HART Sheet



The CS_PDOSEFM_HART control strategy operates the same as the CS_PDOESFM control strategy but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute FM100 for the PV data instance of XT100
- Substitute FT100 for the remaining instances of XT100

For more information, see HART Integration on page 31.

Notes:

Process Dosing Weigh Scale (PDOSEWS) Control Strategy

Use the PDOSEWS control strategy to control an ingredient addition that uses a weigh scale to measure the quantity of ingredient added. The weigh scale can be on the receiving vessel, indicating a gain in weight, or on the sourcing vessel, indicating a loss in weight. The weigh scale can be connected using an analog input, device network, or other connection.

The following PDOSEWS control strategies are available as routines in the process library:

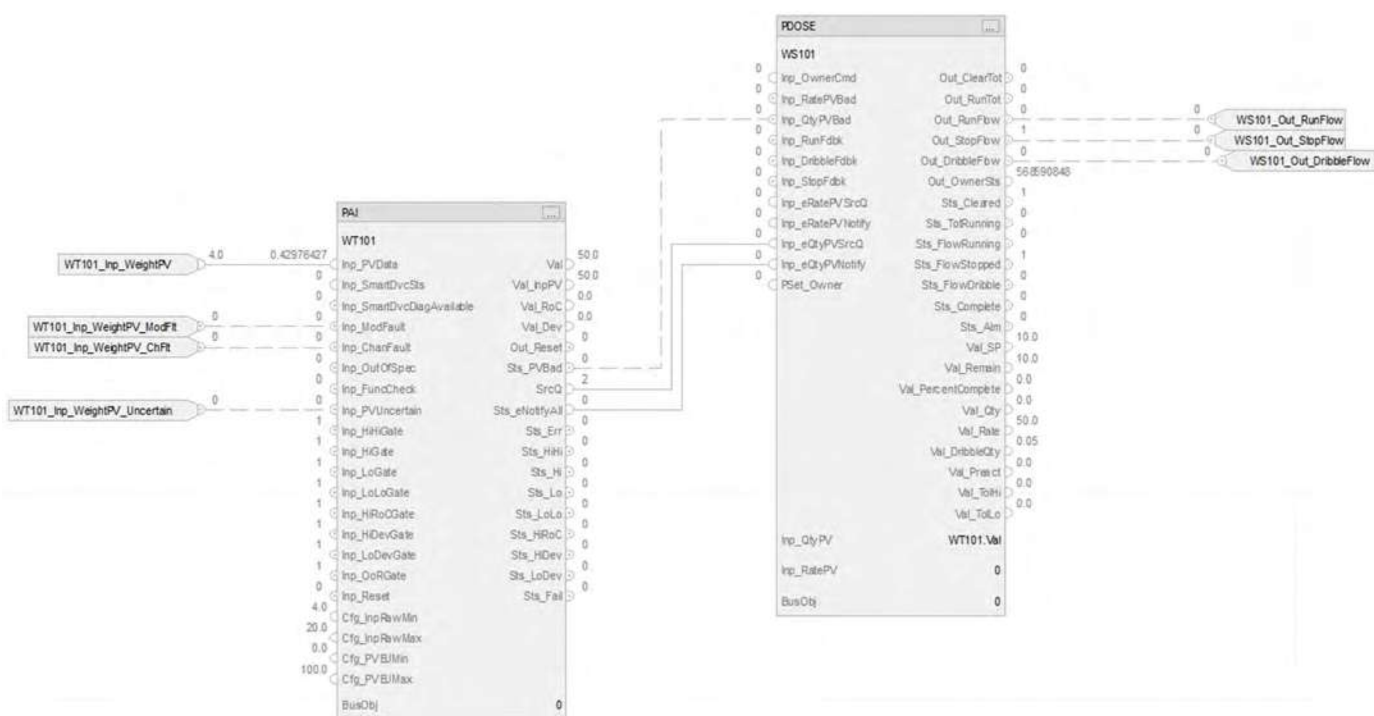
- CS_PDOSEWS
- CS_PDOSEWS_HART

Import the appropriate control strategy as a **routine** in your controller project.

The PDOSEWS control strategy contains one Function Block sheet:

Sheet	Description
CS_PDOSEWS	Process Dosing Weigh Scale instruction
CS_PDOSEWS_HART	Process Dosing Weigh Scale instruction with HART input

CS_PDOSEWS Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.

- Substitute WS101 for the PV data instance of XT101
- Substitute WT101 for the remaining instances of XT101

PAI Outputs to PDOSE Inputs

Parameter	Description
Sts_PVBad	Quality of PV value 1 = PV quality is flagged as Bad
SrcQ	Source and quality of primary value or status: 0 = Good, live, confirmed good 1 = Good, live, assumed good 2 = Good, no feedback, assumed good 8 = Test, simulated 9 = Test, loopback 10 = Test, manually entered 16 = Uncertain, live, off-spec 17 = Uncertain, substituted at device 18 = Uncertain, substituted at instruction 19 = Uncertain, using last known good 20 = Uncertain, using replacement value 32 = Bad, signal failure 33 = Bad, channel fault 34 = Bad, module/communications fault 35 = Bad, invalid configuration
Sts_eNotifyAll	All alarm status enumerated values including related objects: 0 = Not in alarm, acknowledged 1 = Not in alarm, unacknowledged, or reset required 2 = Low severity alarm, acknowledged 3 = Low severity alarm, unacknowledged 4 = Medium severity alarm, acknowledged 5 = Medium severity alarm, unacknowledged 6 = High severity alarm, acknowledged 7 = High severity alarm, unacknowledged 8 = Urgent severity alarm, acknowledged 9 = Urgent severity alarm, unacknowledged

PDOSE Output References

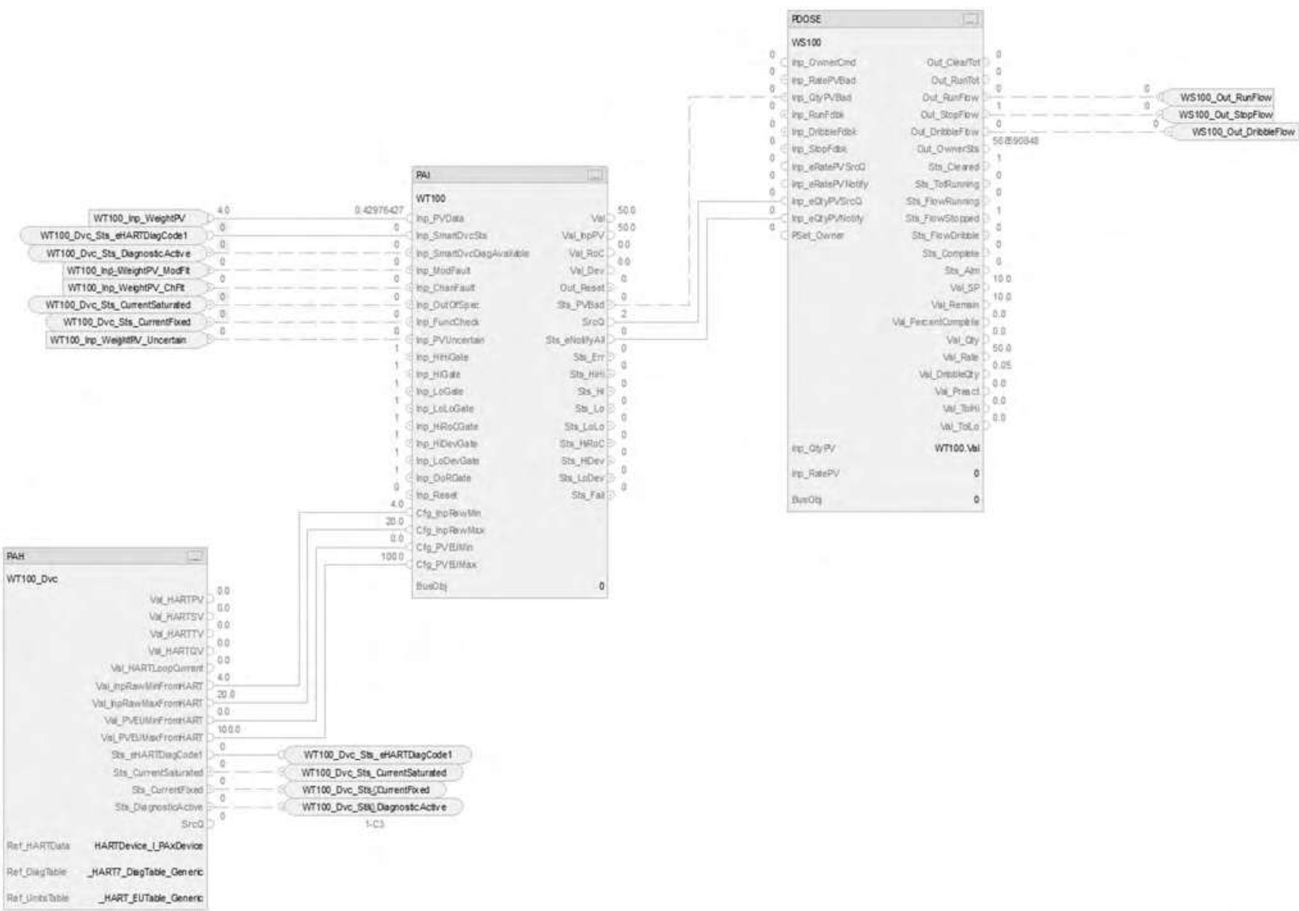
Parameter	Description
WS101_Out_RunFlow	1 = Deliver at full (fast) flow
WS101_Out_StopFlow	1 = Stop delivery equipment
WS101_Out_DribbleFlow	1 = Deliver at dribble (slow) flow

PDOSE Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_DOSING	Instance of data structure (backing tag) required for proper operation of instruction
Inp_QtyPV	REAL	Quantity per time from weigh scale (EU or pulse count). Input is disabled if Sts_CalcQty is either of the following: • 1=integrate Inp_RatePV to get quantity • 0=use Inp_QtyPV
Inp_RatePV	REAL	Rate of material that is added or removed from weigh scale (EU/Time, see Cfg_RateTime). Input is disabled if Sts_CalcRate is either of the following: • 1=differentiate Inp_QtyPV to get rate • 0=use Inp_RatePV
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

A weigh scale usually only provides a quantity signal (how much material is in the device). Connect the PDOSE instruction to the quantity parameter; the instruction calculates the rate by measuring how much the quantity changes (differentiate with respect to time).

CS_PDOSEWS_HART Sheet



The CS_PDOSEWS_HART control strategy operates the same as the CS_PDOSEWS control strategy but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute for WT100 for XT100.
- For more information, see HART Integration on page 31.

Notes:

Process Analog Fanout (PFO) Control Strategies

Use a PFO control strategy to send (fanout) one primary analog output signal to up to 8 secondary users or devices. Each secondary output has configurable gain, offset, and clamping limits.

The PFO instruction receives an input CV (controlled variable) from a primary PID loop or analog output and applies rate-of-change limiting to the input signal. This control strategy is a base component of a PPID Split Range control strategy.

The following CS_PFO control strategy is available as a routine in the process library.

Import the control strategy as a **routine** in your controller project.

The PFO control strategy contains the CS_PFO Function Block sheet.

CS_PFO Sheet

PFO Input References

Parameter	Description
FOT100_Inp_CV	Input CV from upstream block's output (engineering units)
FOT100_CV1_Sts_Available	Initialize request from downstream block #1 1 = set Out_CV1 to Inp_CV1InitializationVal.
FOT100_CV2_Sts_Available	Initialize request from downstream block #2 1 = set Out_CV2 to Inp_CV2InitializationVal.
FOT100_CV3_Sts_Available	Initialize request from downstream block #3 1 = set Out_CV3 to Inp_CV3InitializationVal.
FOT100_CV4_Sts_Available	Initialize request from downstream block #4 1 = set Out_CV4 to Inp_CV4InitializationVal.
FOT100_CV5_Sts_Available	Initialize request from downstream block #5 1 = set Out_CV5 to Inp_CV5InitializationVal.
FOT100_CV6_Sts_Available	Initialize request from downstream block #6 1 = set Out_CV6 to Inp_CV6InitializationVal.
FOT100_CV7_Sts_Available	Initialize request from downstream block #7 1 = set Out_CV7 to Inp_CV7InitializationVal.
FOT100_CV8_Sts_Available	Initialize request from downstream block #8 1 = set Out_CV8 to Inp_CV8InitializationVal.

PFO Output References

Parameter	Description
FOT100_Out_CV1	Output to downstream block #1 (out 1 engineering unit).
FOT100_Out_CV2	Output to downstream block #2 (out 2 engineering units).
FOT100_Out_CV3	Output to downstream block #3 (out 3 engineering units).
FOT100_Out_CV4	Output to downstream block #4 (out 4 engineering units).
FOT100_Out_CV5	Output to downstream block #5 (out 5 engineering units).
FOT100_Out_CV6	Output to downstream block #6 (out 6 engineering units).
FOT100_Out_CV7	Output to downstream block #7 (out 7 engineering units).
FOT100_Out_CV8	Output to downstream block #8 (out 8 engineering units).

PFO Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_ANALOG_FANOUT	Instance of data structure (backing tag) required for proper operation of instruction

Process High or Low Selector (PHLS) Control Strategies

The PHLS control strategy is a base component of the PPID Override control strategy. Use a PHLS control strategy to select the highest or the lowest of as many as six incoming controlled variables (CVs). The instruction sends the selected CV as its output and the output(s) of the 'unselected' PPID controller(s) are kept within Kp*Error of the active PPID controller output to help ensure a quick response when another PPID's output becomes the limiting output.

For example, three PID controls feed a PHLS instruction that is configured to select the lowest of the three PID outputs as the speed reference for a drive. In normal operation, the discharge pressure PID has control, and the other PIDs track the output of the discharge pressure loop. When motor current exceeds its setpoint, or if suction pressure falls below its setpoint, the limit constrained PPID takes control to help prevent motor overcurrent or pump cavitation.

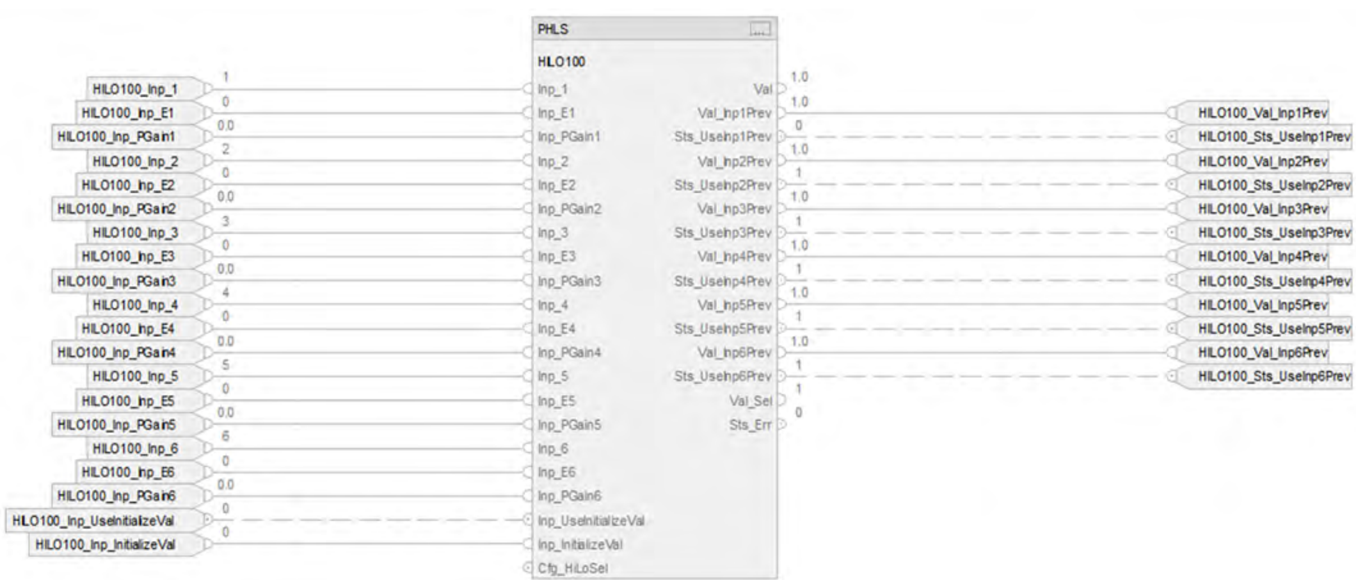
Scaling of the output of this block to CVEU can be done by a downstream PAO block. This block also supports initialization from a downstream block; the initialization is forwarded to upstream blocks.

The CS_PHLS control strategy is available as a routine in the process library.

Import the control strategy as a **routine** in your controller project.

The PHLS control strategy contains the CS_PHLS Function Block sheet.

CS_PHLS Sheet



PHLS Input References

Parameter	Description
HIL0100_Inp_1 HIL0100_Inp_2 HIL0100_Inp_3 HIL0100_Inp_4 HIL0100_Inp_5 HIL0100_Inp_6	Each input# is a CV value.
HIL0100_Inp_E1 HIL0100_Inp_E2 HIL0100_Inp_E3 HIL0100_Inp_E4 HIL0100_Inp_E5 HIL0100_Inp_E6	Loop error from primary input# (optional, used for offset calculation).
HIL0100_Inp_PGain1 HIL0100_Inp_PGain2 HIL0100_Inp_PGain3 HIL0100_Inp_PGain4 HIL0100_Inp_PGain5 HIL0100_Inp_PGain6	Proportional gain from primary input# (optional, used for offset calculation).
HIL0100_Inp_UseInitializeVal	Use an initialization value from a downstream block.
HIL0100_Inp_InitializeVal	Initialization value from a downstream block.

PHLS Output References

Parameter	Description
HIL0100_Val_Inp1Prev HIL0100_Val_Inp2Prev HIL0100_Val_Inp3Prev HIL0100_Val_Inp4Prev HIL0100_Val_Inp5Prev HIL0100_Val_Inp6Prev	Previous (Feedback) input value for primary input#
HIL0100_Sts_UseInp1Prev HIL0100_Sts_UseInp2Prev HIL0100_Sts_UseInp3Prev HIL0100_Sts_UseInp4Prev HIL0100_Sts_UseInp5Prev HIL0100_Sts_UseInp6Prev	Request for primary input# to use feedback Val_Inp#Prev

PHLS Configuration Considerations

Operand	Type	Description
PHLS tag	P_HIGH_LOW_SELECT	Instance of data structure (backing tag) required for proper operation of instruction

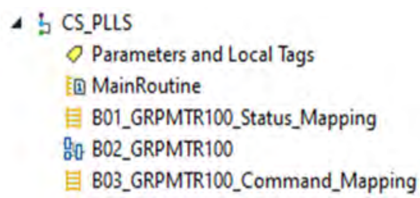
Chapter 20

Process Lead Lag Standby (PLLS) Control Strategy

Use a PLLS control strategy to control of a parallel group of motors, such as a set of pumps with a common intake source and discharge destination. The number of motors to run depends on the demand on the system. The group can be configured to consist of as few as two or as many as 30 motors. The minimum demand can be set as low as 0, so that all motors are stopped at minimum demand. The maximum demand can be set as high as the number of pumps in the group.

The PLLS control strategy is available as three routines in the process library:

Routine	Description
GRPMTR100_Status_Mapping	Map motor status into the inputs of PLLS routine.
GRPMTR100	Function Block control strategy routine
GRPMTR100_Command_ Mapping	Map the commands of the PLLS out to the commands of the motor.



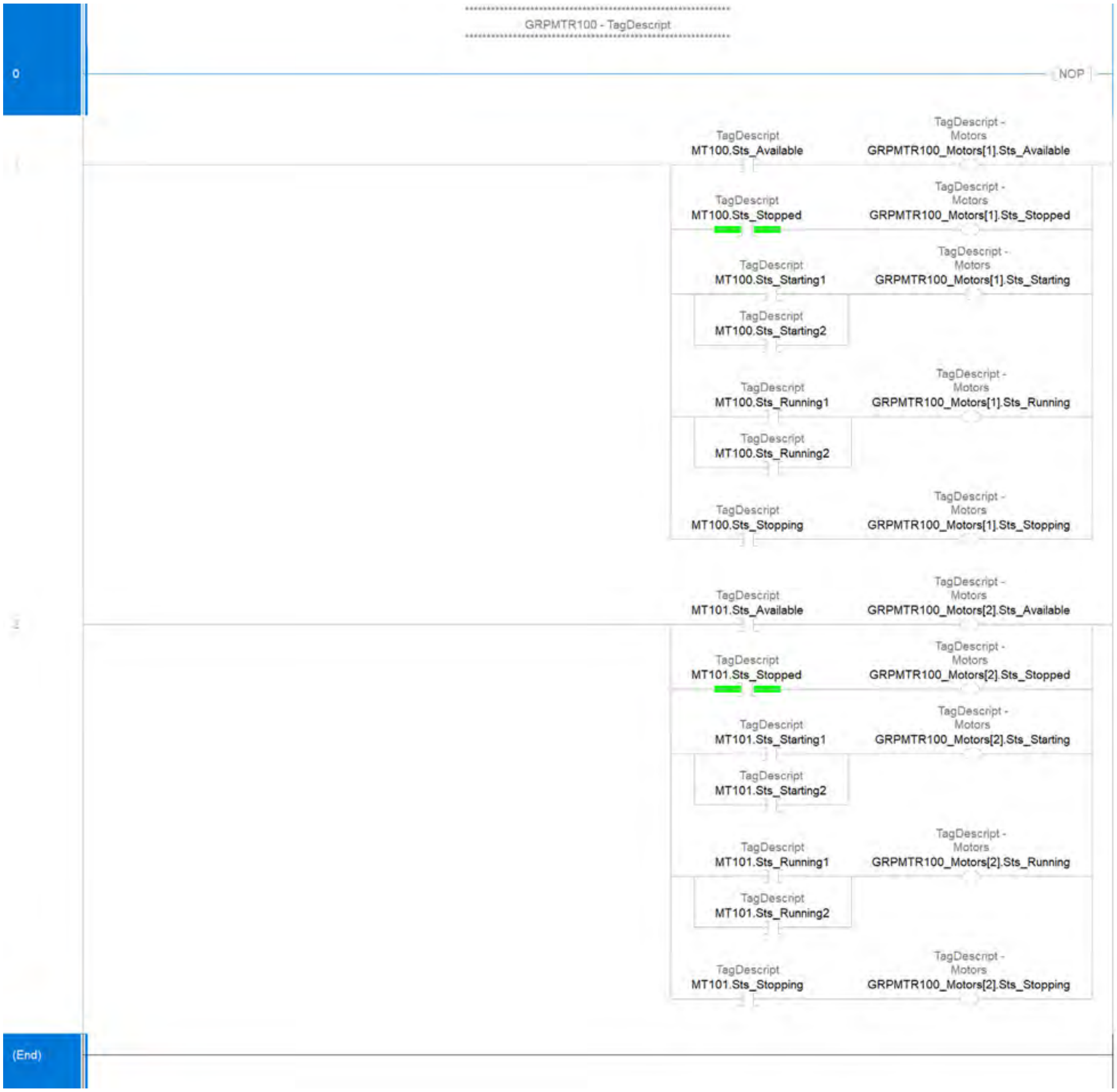
Import the PLLS as **program** your controller project. The execution order of the routines is important for the proper operation of this control strategy.

IMPORTANT For proper operation of this control strategy:

- In order for PLLS to align with the states of the motors on first scan, the motor / drive logic must be executed before this control strategy.
- The routines in this strategy must be executed in the correct order:

1. GRPMTR100_Status_Mapping
2. GRPMTR100
3. GRPMTR100_CommandMapping.

GRPMTR100 Status Mapping Routine



CS_PLLS Sheet

The GRPMTR100 routine contains these Function Block sheets:

Sheet	Description
CS_PLLS	Process Lead Lag Standby Motor Group instruction
Permissive	Process Permissives instruction The Process Permissives (PPERM) instruction collects, or sums up, the permissive conditions that let a piece of equipment energize. In most cases, permissive conditions must be true to energize equipment. Once the equipment is energized, permissives are ignored.
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The PLLS instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder.

PLLS Input References

Parameter	Description
PermOK	Input connection from permissive sheet 1 = On permissives OK, group can start
NBPermOK	Input connection from permissive sheet 1 = Non-bypassable On permissives OK, group can start
GRPMTR100_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
GRPMTR100_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
GRPMTR100_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
GRPMTR100_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops group but does not raise trip alarm
GRPMTR100_Intlk_BankSts.Sts_RdyReset	Interlock bank status 1 = A latched interlock (returned to OK) is ready to be reset

PLLS Output References

Parameter	Description
BypActive	Output connection to permissive and interlock bank sheets

PLLS Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_LEAD_LAG_STANDBY	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.
Ref_Motors	P_LEAD_LAG_STANDBY_MOTOR array	Reference to GRPMTR100_Motors array, used for motor status and motor command mapping.

Permissive Sheet

PPERM Input References

Parameter	Description
BypActive	Input connection from CS_PLLS sheet

PPERM Output References

Parameter	Description
PermOK	Overall permissive status (1 = OK to start group)
NBPermOK	Non-bypassable permissive status (1 = all non-bypassable permissives OK to start group)

Interlock Bank Sheet

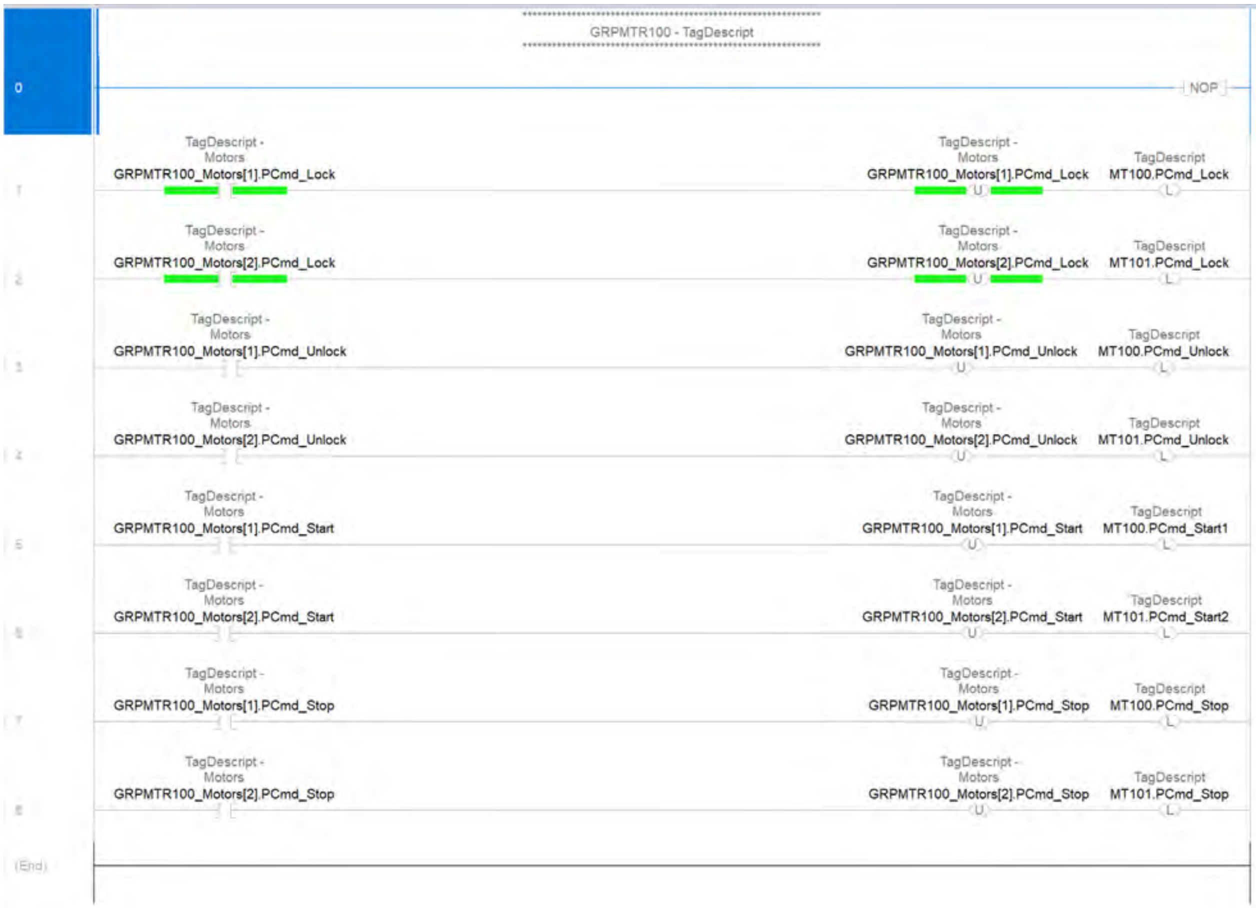
Input Reference to PINTLK

Parameter	Description
BypActive	Input connection from CS_PLLS sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

GRPMTR100 Command Mapping Routine



Chapter 21

Process Motor (PMTR) Control Strategies

Use a PMTR control strategy to monitor and control a fixed single-speed, two-speed, or reversing motor using a full-voltage contactor or intelligent motor controller (soft starter). The motor can be run or jogged, including jogging reverse or jogging fast. The control strategy uses a Device Object to interface with the hardware motor controller.

Additional features provided in this control strategy, include a Runtime and Start Counter (to record the total run time and number of drive starts) and a Restart Inhibit (to limit the number of starts within a specified time period to protect the motor windings from overheating).

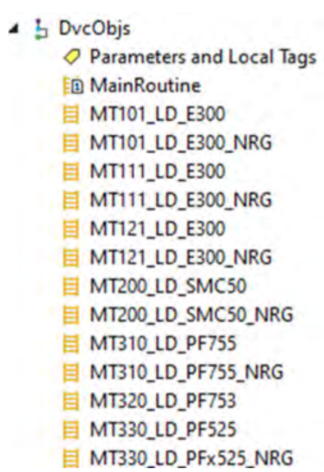
The following PMTR control strategies are available as routines in the process library:

Motor Controller Type	Control Strategy
E300™ Electronic Overload Relay	CS_PMTR_1S_E300 CS_PMTR_1S_E300_Energy CS_PMTR_2S_E300 CS_PMTR_2S_E300_Energy CS_PMTR_REV_E300 CS_PMTR_REV_E300_Energy
SMC™-50 Motor Controller	CS_PMTR_1S_SMC50 CS_PMTR_1S_SMC50_Energy
Basic	CS_PMTR_1S CS_PMTR_1S_Hand CS_PMTR_2S CS_PMTR_2S_Hand CS_PMTR_REV CS_PMTR_REV_Hand

Import the appropriate control strategy as a **routine** in your controller project.

Also, import the appropriate device object as a routine in your controller project. These objects are from the Power Device Library and must be downloaded separately from the PlantPAx® Process Library.

Each '_NRG' object uses the Energy object to group energy parameters for the device. Use this object with the corresponding, energy-related control strategy.

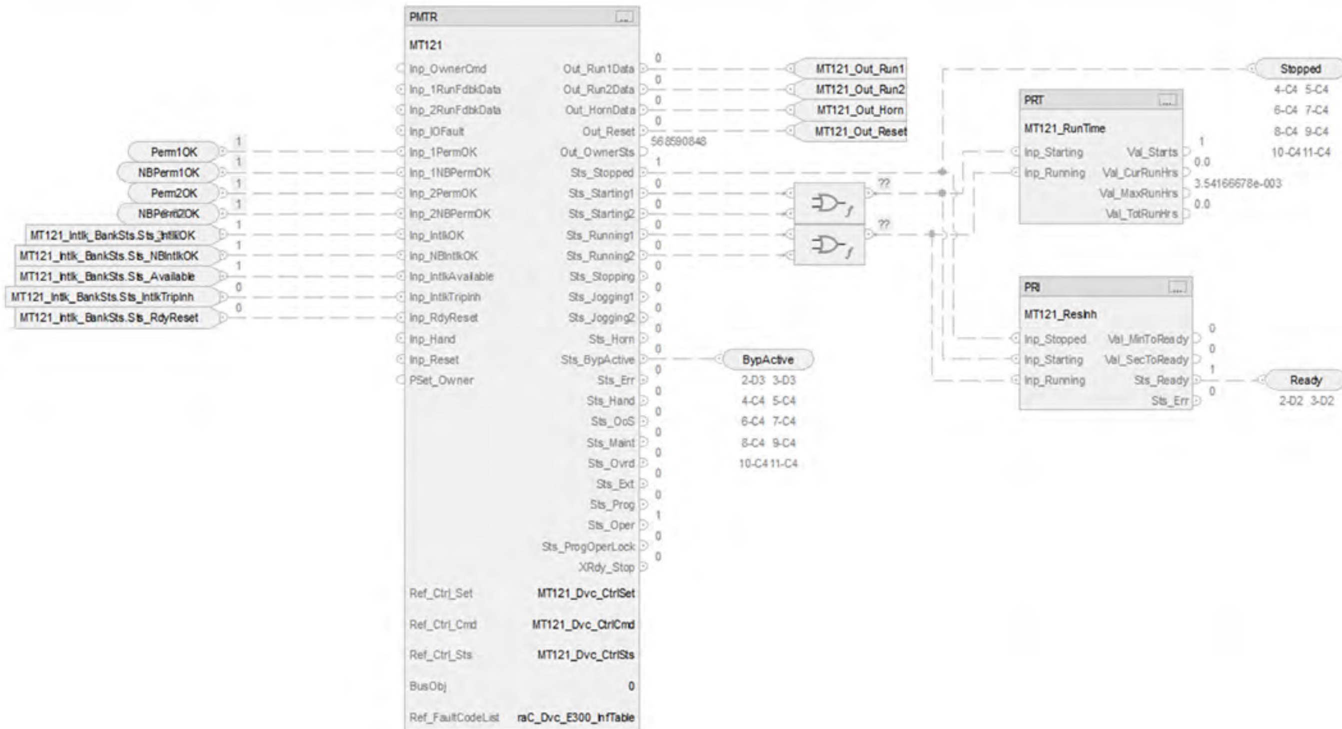


The PMTR control strategies contain these Function Block sheets:

Sheet	Description
CS_PMTR	Process Motor instruction
Permissives 1 Permissives 2	Process Permissives instruction The Process Permissives (PPERM) instruction collects, or sums up, the permissive conditions that let a piece of equipment energize. In most cases, permissive conditions must be true to energize equipment. Once the equipment is energized, permissives are ignored. The Permissives 2 sheet is only in the control strategies for two-speed and reversing motor controllers
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The PMTR instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder

In the input and output reference descriptions on each sheet, [device] = PMTR instance tag.

CS_PMTR Sheet



PMTR Input References

Parameter	Description
Perm1OK	Input connection from Permissives 1 sheet (single speed) 1 = On permissives OK, device can turn On
NBPerm1OK	Input connection from Permissives 1 sheet (single speed) 1 = Non-bypassable On permissives OK, device can turn On
Perm2OK	Input connection from Permissives 2 sheet (second speed or reverse) 1 = On permissives OK, device can turn On
NBPerm2OK	Input connection from Permissives 2 sheet (second speed or reverse) 1 = Non-bypassable On permissives OK, device can turn On
[device]_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
[device]_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
[device]_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
[device]_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
[device]_Intlk_BankSts.Sts_RdyReset	Interlock bank status 1 = A latched interlock (returned to OK) is ready to be reset

PMTR Output References

Parameter	Description
[device]_Out_Run1	Single speed 1=Start/Run Motor Reverse or Fast 0=Stop Motor (for held starter type)
[device]_Out_Run2	Second speed or reverse 1=Start/Run Motor Reverse or Fast 0=Stop Motor (for held starter type)
[device]_Out_Horn	1 = Notification before commanded state change
[device]_Out_Reset	1 = Reset command has been received and accepted
BypActive	Output connection to permissives and interlock bank sheet
Ready	Output connection to the permissive sheet
Stopped	Output connection to interlock bank sheet

The Boolean OR performs a bitwise OR based on these PMTR outputs:

- Sts_Stopped
- Sts_Starting1
- Sts_Starting2
- Sts_Running1
- Sts_Running2

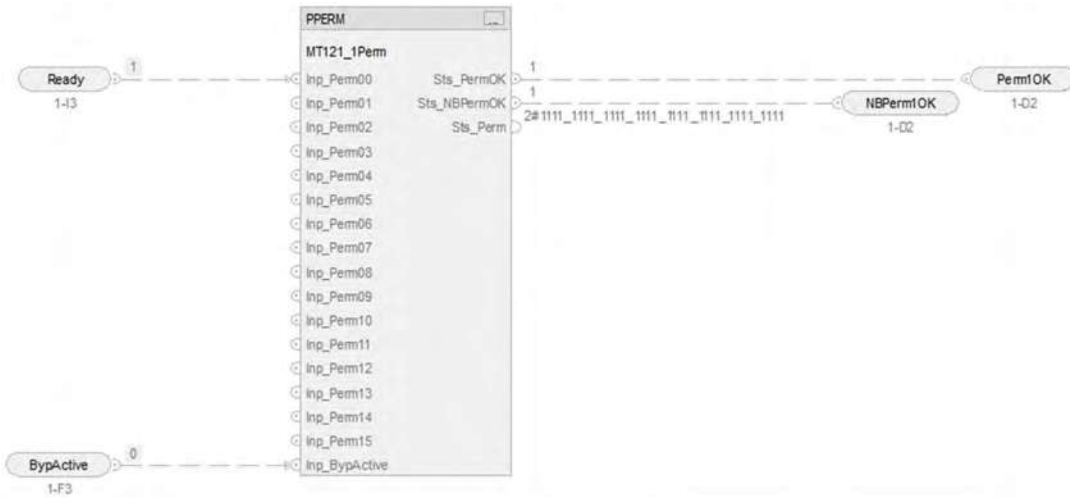
The result feeds these instructions:

Instruction	Description
Process Run Time and Start Counter (PRT)	The PRT instruction records the total run time and number of instances the drive starts.
Process Restart Inhibit (PRI)	The PRI instruction helps prevent the drive from starting repeatedly within specified time periods. Continual starts or start attempts in a short period overheat the motor windings and damage the motor.

PMTR Configuration Considerations

Operand	Type	Description
PlantPAx control	P_MOTOR_DISCRETE	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.
Ref_Ctrl_Set	RAC_ITF_DVC_PWRDISCRETE_SET	Power Discrete Automation Device Object Settings Interface Preconfigured in the device object ladder routine
Ref_Ctrl_Cmd	RAC_ITF_DVC_PWRDISCRETE_CMD	Power Discrete Automation Device Object Command Interface Preconfigured in the device object ladder routine
Ref_Ctrl_Sts	RAC_ITF_DVC_PWRDISCRETE_STS	Power Discrete Automation Device Object Status Interface Preconfigured in the device object ladder routine
Ref_FaultCodeList	RAC_CODEDESCRIPTION[400]	Fault Code to Fault Description lookup table for the motor controller Preconfigured in the device object ladder routine

Permissive Sheet



PPERM Input References

Parameter	Description
Ready	Input connection from the CS_PMTR sheet
BypActive	Input connection from the CS_PMTR sheet

PPERM Output References

Parameter	Description
Perm1OK Perm2OK	Overall permissive status (1 = OK to energize)
NBPerm1OK NBPerm2OK	Non-bypassable permissive status (1 = all non-bypassable permissives OK to energize)

Interlock Bank Sheet

PINTLK Input Reference

Parameter	Description
BypActive	Input connection from CS_PMTR sheet
Stopped	Input connection from the CS_PMTR sheet

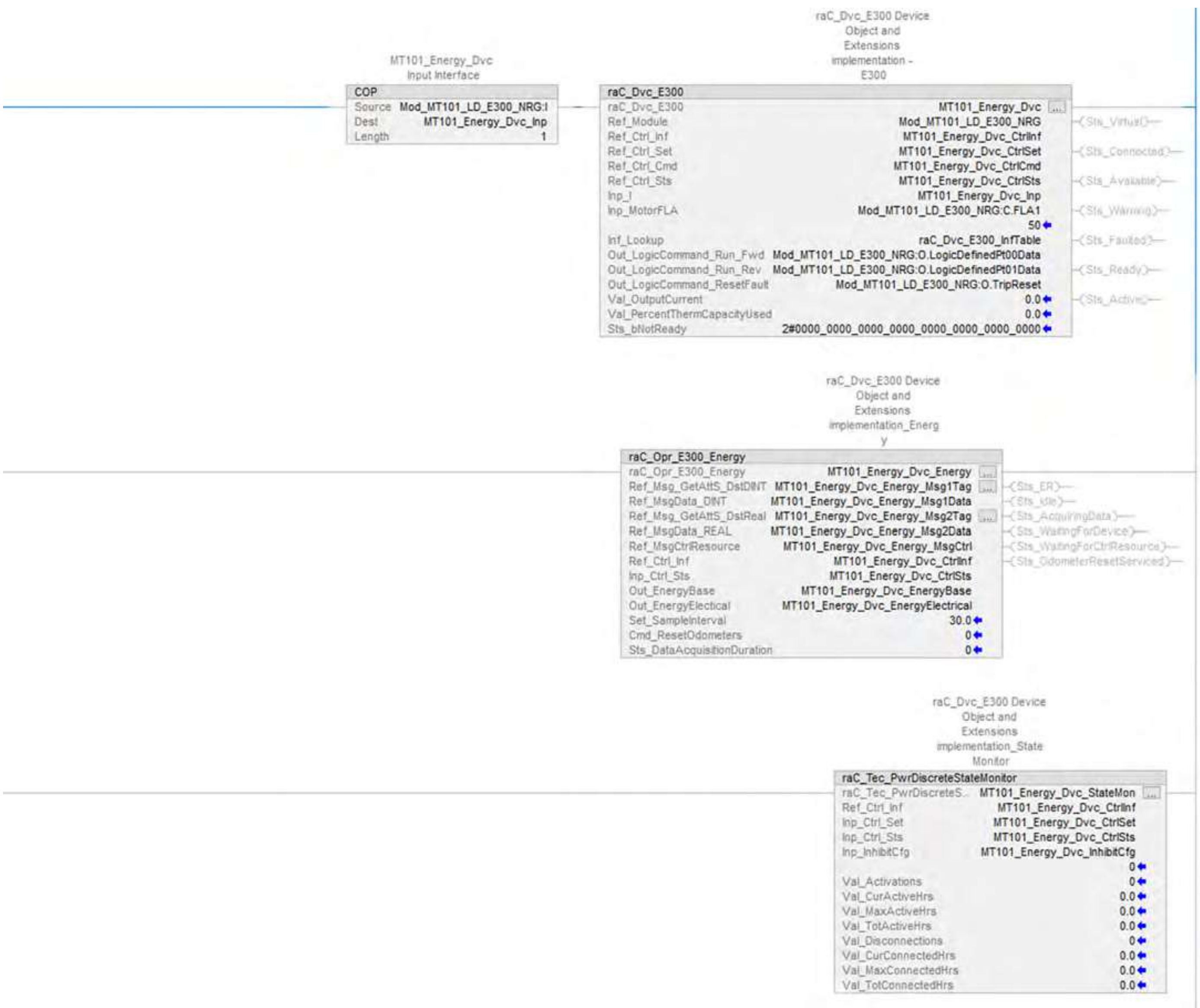
PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

Motor Controller Device Objects

Single Speed

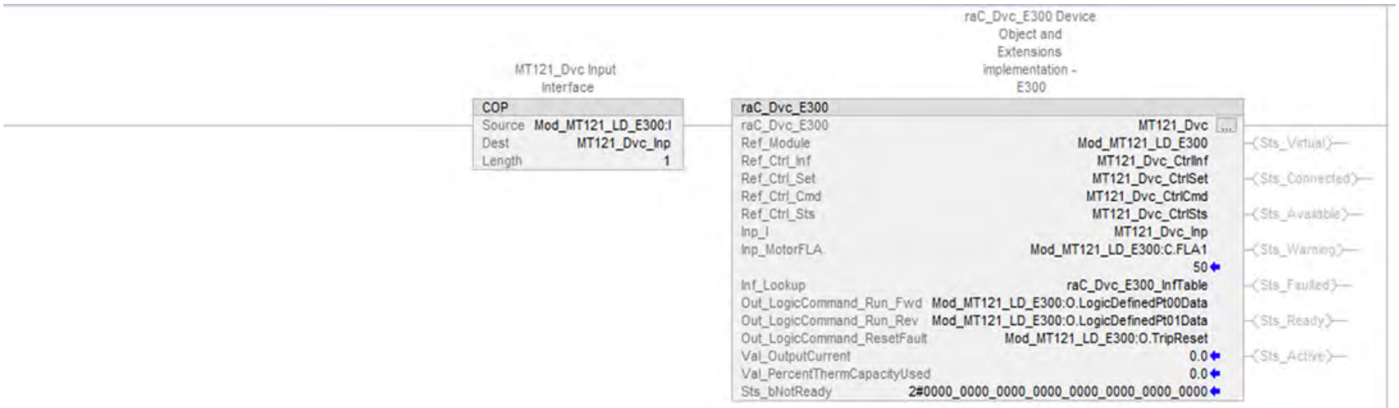
Single Speed with Energy Parameters



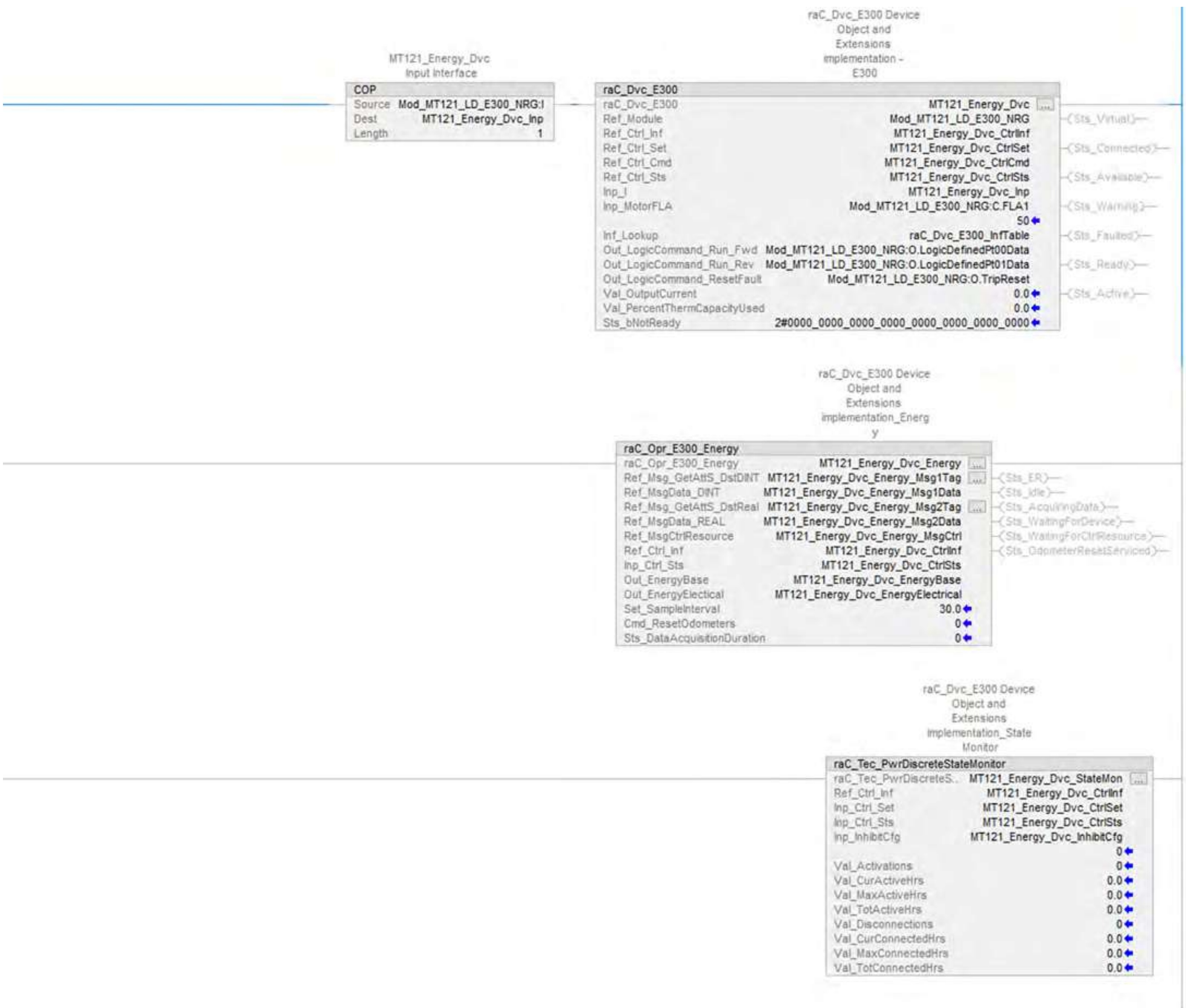
Two Speed

Two Speed with Energy Parameters

Reversing



Reversing with Energy Parameters



Notes:

Process n-Position (PNPOS) Control Strategy

The Process n-Position Device (PNPOS) instruction controls a circular or linear discrete device. The device can have between 2 and 30 positions. The instruction lets you select each position with associated outputs and feedbacks.

For linear devices, the PNPOS instruction can be configured to:

- Return to Position 1 on every move, approaching the target position from the 'same side' on each move to improve position repeatability.
- Move directly to the new position.

For circular devices, the PNPOS instruction can be configured to:

- Move only "clockwise" to increasing positions. For example, with an 8-position device, a move from position 1 to position 6 could be clockwise only (from position 1 through positions 2, 3, 4, and 5 to position 6).
- Move in whichever direction provides the shortest move. For example, with an 8-position device, it could use the shortest path (from position 1 through positions 8 and 7 to position 6).

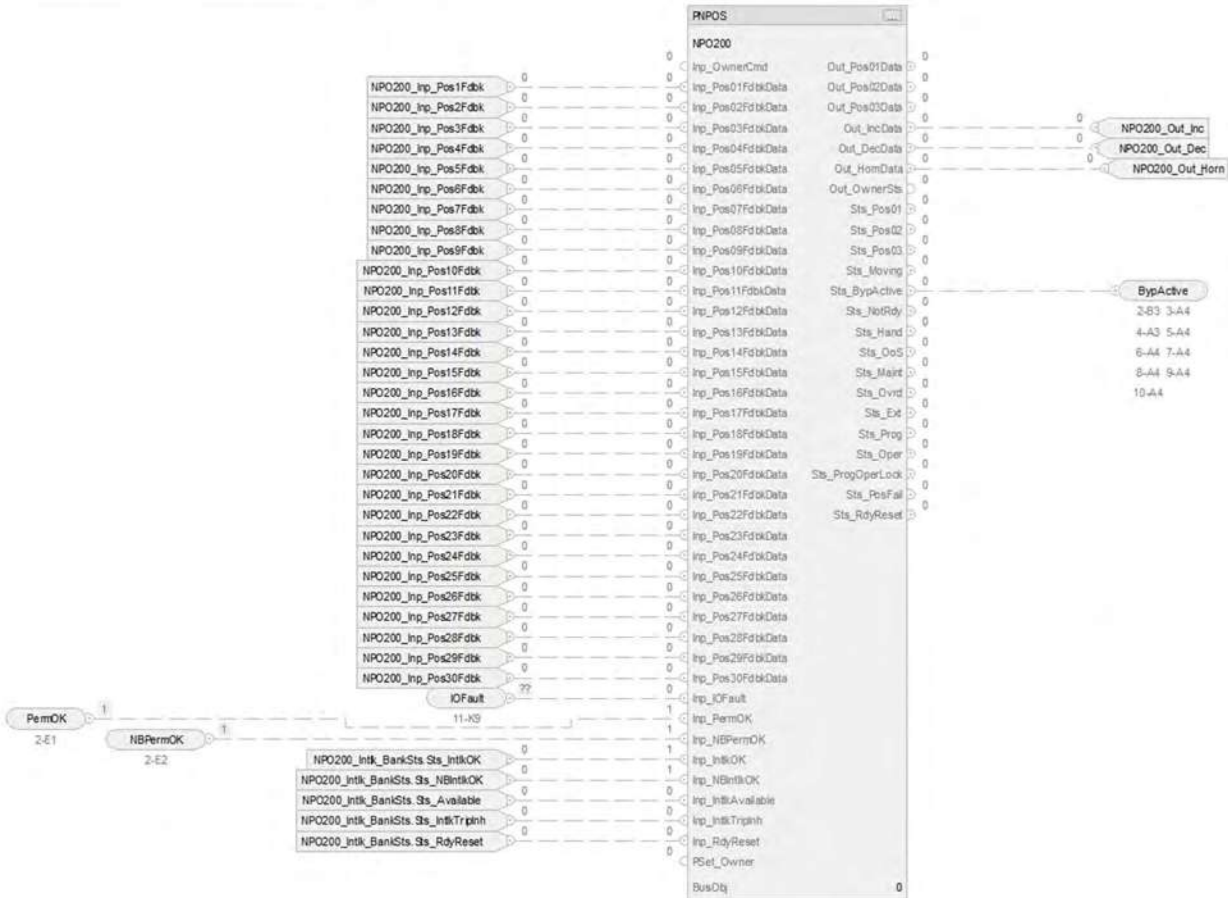
The CS_PNPOS control strategy is available as a routine in the process library.

Import the control strategy as a **routine** in your controller project.

The NPOS control strategy contains these Function Block sheets:

Sheet	Description
CS_PNPOS	n-Position Device Add-On Instruction
Permissives	Process Permissives instruction The Process Permissives (PPERM) instruction collects, or sums up, the permissive conditions that let a piece of equipment energize. In most cases, permissive conditions must be true to energize equipment. Once the equipment is energized, permissives are ignored.
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder.
IO Fault	The logic monitors as many as 30 discrete input channels and as many as three discrete output channels for I/O fault input and raises an alarm on an I/O fault.

CS_PNPOS Sheet



PNPOS Input References

Parameter	Description
NPO200_Inp_PosxFdbk Where x=1-30	Position x feedback, 1 = Device is confirmed at Position x.
IOFault	Input connection from IO Faults sheet
PermOK	Input connection from Permissives sheet 1 = On permissives OK, device can turn On
NBPermOK	Input connection from Permissives sheet 1 = Non-bypassable On permissives OK, device can turn On
NPO200_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
NPO200_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
NPO200_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
NPO200_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
NPO200_Intlk_BankSts.Sts_RdyReset	Interlock bank status 1 = A latched interlock (returned to OK) is ready to be reset

PNPOS Output References

Parameter	Description
NPO200_Out_Horn	1 = Sound audible before commanded state change
NPO200_Out_Inc	Increment output
NPO200_Out_Dec	Decrement output
BypActive	Output connection to permissives and interlock bank sheets

PNPOS Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_DISCRETE_N_POSITION	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

Permissive Sheet

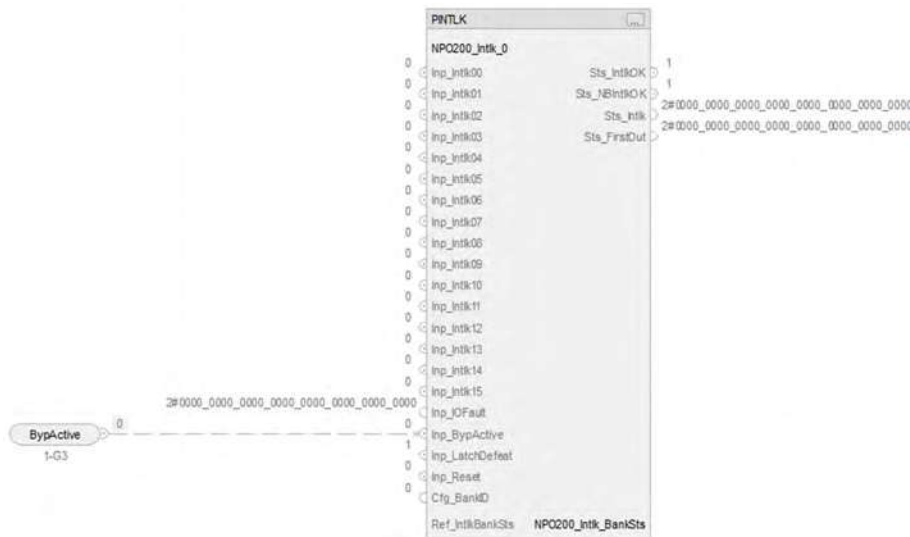
PPERM Input References

Parameter	Description
BypActive	Input connection from the CS_PNPOS sheet

PPERM Output References

Parameter	Description
PermOK	Overall permissive status (1 = OK to energize)
NBPermOK	Non-bypassable permissive status (1 = all non-bypassable permissives OK to energize)

Interlock Bank Sheet



PINTLK Input Reference

Parameter	Description
BypActive	Input connection from the CS_PNPOS sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction NPO100 in this example corresponds to a linear device
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

IO Faults Sheet

Fault Input References

Parameter	Description
NP0200_Inp_PosxFdbk_ChFlt Where x=1-30	Tieback input x channel fault
NP0200_Out_Dec_ChFlt	Decrease output channel fault
NP0200_Out_Dec_ModFlt	Decrease output module fault
NP0200_Out_Inc_ChFlt	Increase output channel fault
NP0200_Out_Inc_ModFlt	Increase output module fault
D4SD100_Out_Horn_ChFlt	Sound audible for output channel fault
D4SD100_Out_Horn_ModFlt	Sound audible for output module fault

Fault Output References

Parameter	Description
IOFault	Output connection to CS_PNPOS sheet

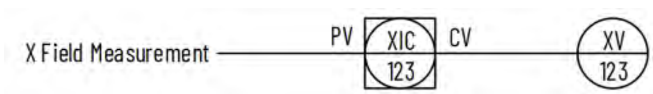
For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

Process Proportional + Integral + Derivative (PPID) Basic Control Strategies

Use this basic PPID control strategy to manipulate one Control Variable (CV) in response to an error (the difference between the Process Variable (PV) readings and the Setpoint (SP, the target PV) settings.

To scale the CV to align with the associated IO module channel range or to accommodate a fail-open (FO) valve (or air to close) use either of the following options:

- Use a basic PPID with Analog Output control strategy
- Insert a scalar Instruction between the PPID CV and the analog output channel reference



The PPID control strategies are pre-configured to enable selectable controller actions (CV Action, SP Action, and Loop Mode Action) based on various shed conditions (Interlock trip, CV fail, PV fail, and SP fail).

The following PPID control strategies are available as routines in the process library:

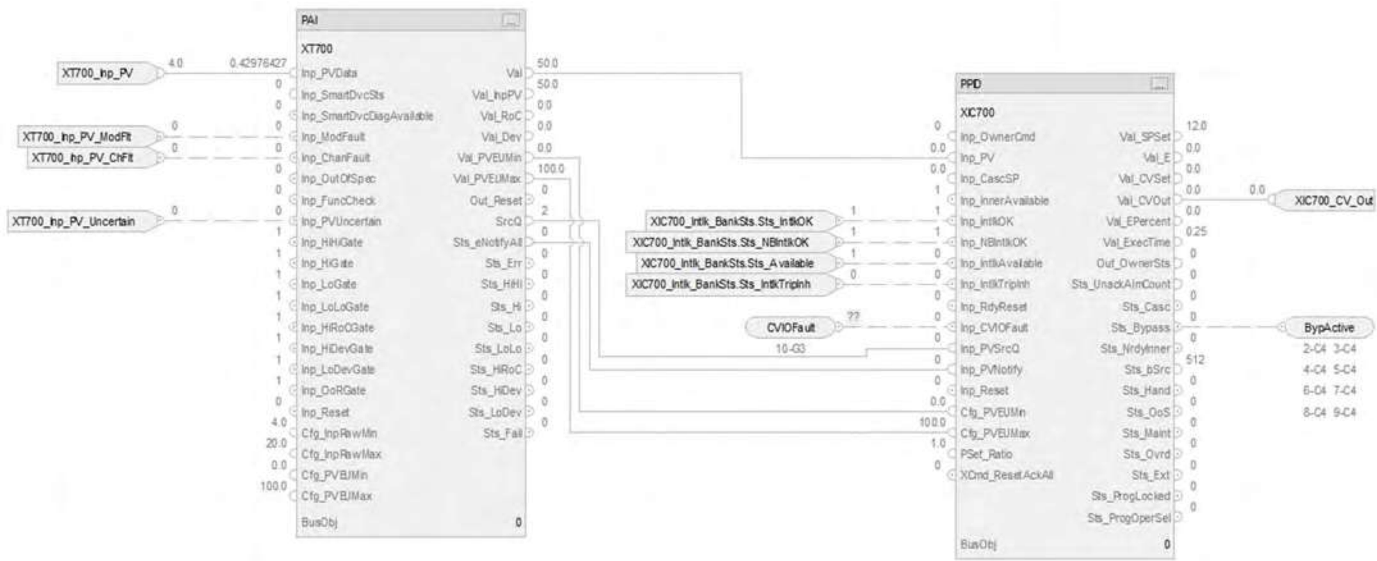
- CS_PPID
- CS_PPID_HART

Import the appropriate control strategy as a **routine** in your controller project.

The PPID control strategy contains these sheets.

Sheet	Description
CS_PPID	PPID instruction
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The PPID instruction monitors bypassablebypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder
IO Faults	The PPID instruction monitors Control Variable faults.

CS_PPID Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.

- Substitute XT700 for XT101

PAI Outputs to PPID Inputs

Parameter	Description
Val	Value for PPID Inp_PV parameter Process Variable (PVEU)
SrcQ	Value for PPID Inp_PVSrcQ parameter Inp_PV source status and quality: 0 = Good, live, confirmed good 1 = Good, live, assumed good 2 = Good, no feedback, assumed good 8 = Test, simulated 9 = Test, loopback 10 = Test, manually entered 16 = Uncertain, live, off-spec 17 = Uncertain, substituted at device 18 = Uncertain, substituted at instruction 19 = Uncertain, using last known good 20 = Uncertain, using replacement value 32 = Bad, signal failure 33 = Bad, channel fault 34 = Bad, module/communications fault 35 = Bad, invalid configuration
Sts_eNotifyAll	Value for PPID Inp_PVNotify parameter Related PV object alarm priority and acknowledgment status: 0 = Not in alarm, acknowledged 1 = Not in alarm, unacknowledged or reset required 2 = Low severity alarm, acknowledged 3 = Low severity alarm, unacknowledged 4 = Medium severity alarm, acknowledged 5 = Medium severity alarm, unacknowledged 6 = High severity alarm, acknowledged 7 = High severity alarm, unacknowledged 8 = Urgent severity alarm, acknowledged 9 = Urgent severity alarm, unacknowledged

PPID Input References

Parameter	Description
XIC700_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
XIC700_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
XIC700_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
XIC700_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
CVIOFault	Input connection from IO Faults sheet

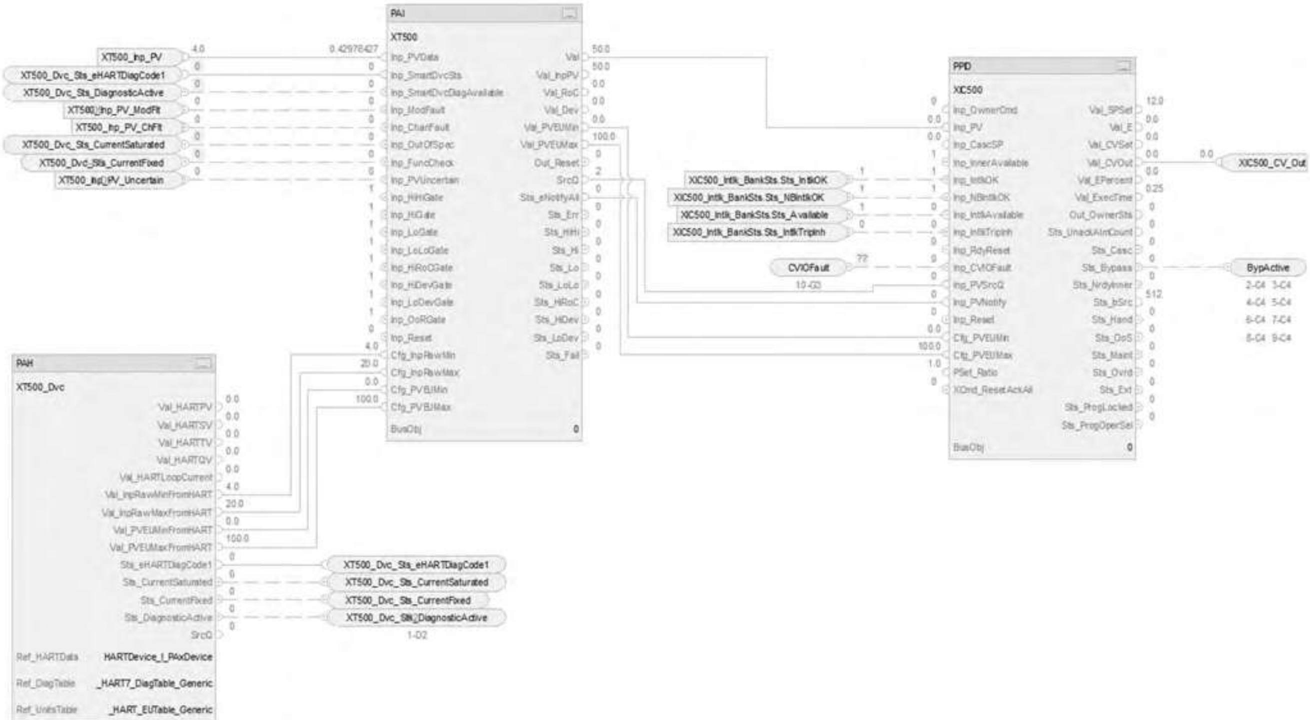
PPID Output References

Parameter	Description
XIC700_Out_CV	Control Variable output Loop CV after clamping and ramping (CVEU)
BypActive	Output connection to interlock bank sheet

PPID Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_PID	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

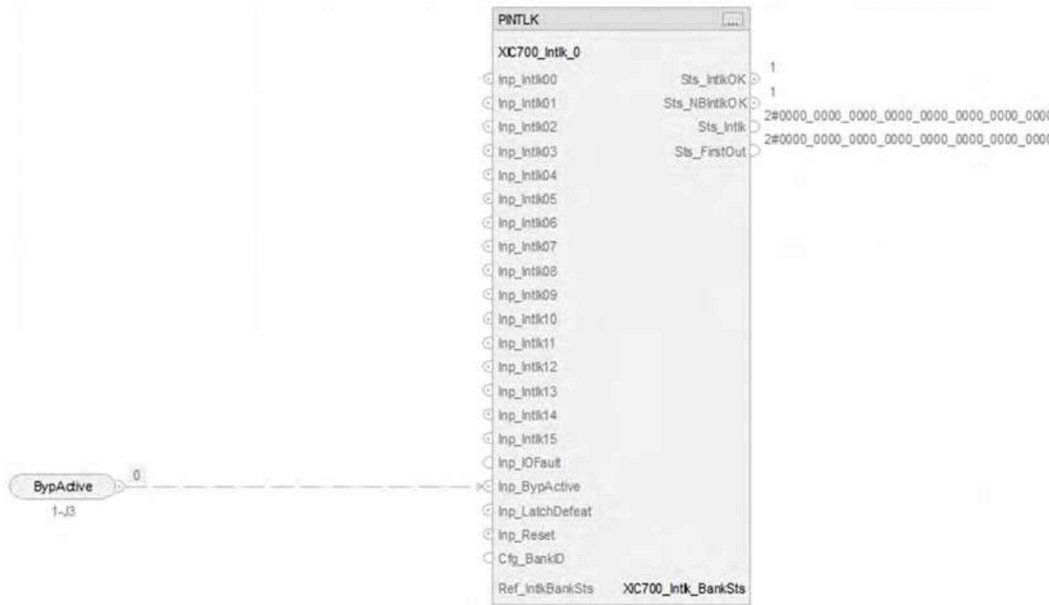
CS_PPID HART Sheet



The CS_PPID HART sheet operates the same as the CS_PPID sheet but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute for XT500 for XT100
- For more information, see HART Integration on page 31.

Interlock Bank Sheet



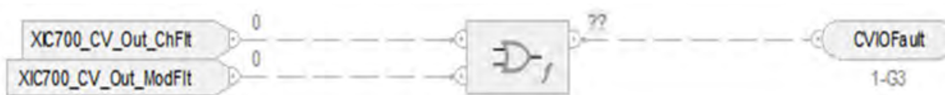
PINTLK Input Reference

Parameter	Description
BypActive	Input connection from CS_PPID sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

IO Faults Sheet



Fault Input References

Parameter	Description
XIC700_Out_CV_ChFlt XIC500_Out_CV_ChFlt	Channel fault 1 = I/O channel fault or failure 0 = OK
XIC700_Out_CV_ModFlt XIC5000_Out_CV_ModFlt	Module fault 1 = I/O module failure or module communication status bad 0 = OK

Fault Output Reference

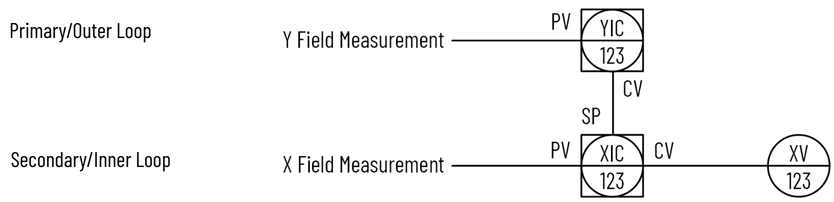
Parameter	Description
CVIOFault	Output connection to CS_PPID sheet

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

Notes:

Process Proportional + Integral + Derivative (PPID) with Cascade Control Strategies

Cascade control is defined as when an outer (primary) control loop's output (CV) is used as a setpoint (SP) to an inner (secondary) control loop. The PPID Cascade control strategy is useful when external disturbances to the inner loop process variable are frequent, which can eventually cause disturbances to the process variable of the outer control loop. Controlling the disturbance at the faster acting inner loop compensates for the resulting disturbance to the outer control loop. Also, non-linearities in the final control element can also be controlled at the faster acting Inner loop reducing potential disturbances to the outer loop.



The provided control strategies are pre-configured to provide the following features:

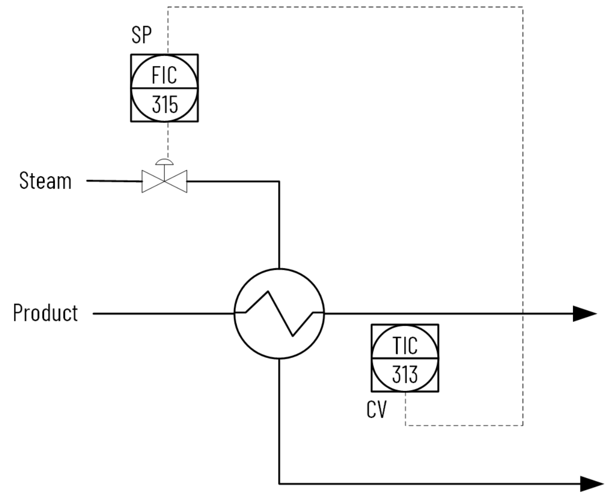
- Bumpless transfer: when the inner control loop is not available, the outer loop output (CV) tracks the inner loop setpoint.
- Anti-windup: when the inner loop hits a CV limit, the outer control loop output is prevented from winding up (increasing or decreasing) beyond that limit.

Another feature pre-configured in the control strategies is the visibility of the whole cascade control strategy status at both the outer and inner control loops. This lets you place the inner and outer PPIDs on different operator displays, while each PPID indicates the status of the whole strategy.

PPID with Cascade Control Example

In this example, the temperature of product flows through a heat exchanger where the exiting product temperature is ultimately maintained at setpoint by modulating the flow of steam to the heat exchanger.

A cascade loop provides better control by opening the steam valve when the steam flow drops before the product temperature drops. To implement a cascade loop, use a PPID instruction to control the steam valve opening based on a process variable signal from a steam flow transmitter. This is the inner loop of the cascaded pair. A second PPID instruction (outer loop) uses the product temperature as a process variable and sends its CV output into the setpoint of the inner loop. In this manner, the outer temperature loop modulates the steam flow setpoint of the inner loop. The steam flow loop is then responsible for providing the amount of steam that is requested by the temperature loop to maintain a constant product temperature.

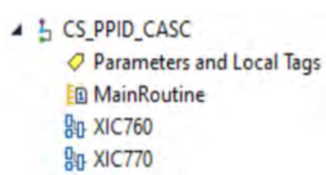


An external disturbance to the outer loop (such as an increase in product flow) results in a reduction in temperature. In this scenario, the outer loop increases its output to increase the steam flow setpoint to bring the product temperature back to setpoint.

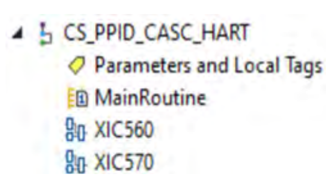
If an upstream disturbance reduces the steam pressure, the steam flow controller (inner loop) reacts by opening the steam valve to maintain steam flow that mitigates any resulting disturbance to the product temperature (outer loop).

The following PPID Cascade control strategies (consisting of multiple routines) are available in the process library:

- CS_PPID_CASC



- CS_PPID_CASC_HART



Import the **routines** for the appropriate control strategy in your controller project. Each control strategy contains multiple routines; each routine contains multiple Function Block sheets.

Each PPID Cascade control strategy contains these routines:

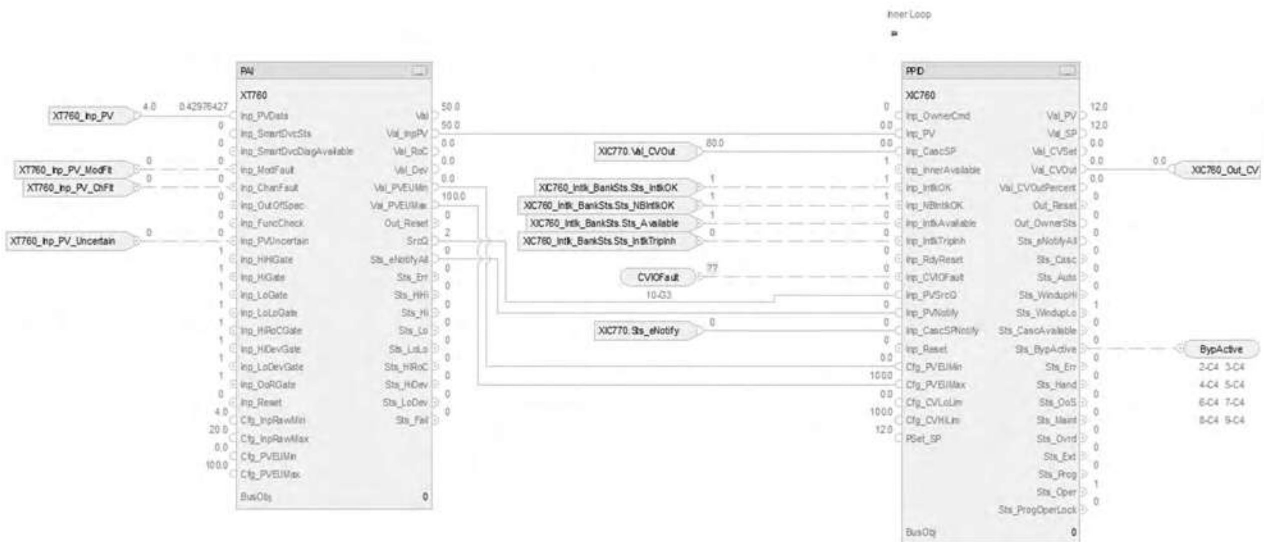
- Inner Loop (XIC760 analog/XIC560 HART)
- Outer Loop (XIC770 analog/XIC570 HART)

ROUTINE Inner Loop (XIC760 analog and XIC560 HART)

Each routine contains these sheets

Sheet	Description
CS_PPID_CASC – Inner Loop	PPID inner loop • XIC760 analog • XIC560 HART
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The PPID instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder
IO Faults	The logic monitors Control Variable faults.

CS_PPID_CASC -Inner Loop Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.

- Substitute XIC760 for the PV data instance of XT101
- Substitute XT770 for the remaining instances of XT101

PAI Outputs to PPID Inputs

Parameter	Description
Val_InpPV	Analog input value in engineering units (actual, before Substitute PV selection)
Val_PVEUmin	Value for PPID Cfg_PVEUMin parameter PV minimum value for scaling from engineering units to %, PV at 0% (PVEU). Valid any float less than Cfg_PVEUMax.
Val_PVEUMax	Value for PPID Cfg_PVEUMax parameter PV maximum value for scaling from engineering units to %, PV at 100% (PVEU). Valid any float greater than Cfg_PVEUMin.
SrcQ	Value for PPID Inp_PVSrcQ parameter Inp_PV source status and quality: 0 = Good, live, confirmed good 1 = Good, live, assumed good 2 = Good, no feedback, assumed good 8 = Test, simulated 9 = Test, loopback 10 = Test, manually entered 16 = Uncertain, live, off-spec 17 = Uncertain, substituted at device 18 = Uncertain, substituted at instruction 19 = Uncertain, using last known good 20 = Uncertain, using replacement value 32 = Bad, signal failure 33 = Bad, channel fault 34 = Bad, module/communications fault 35 = Bad, invalid configuration

PPID Input References

Parameter	Description
XIC770.Val_CVout	Outer Loop CV after clamping and ramping (CVEU).
XIC760_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
XIC760_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
XIC760_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
XIC760_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
CVIOFault	Input connection to IO Faults sheet
XIC770.Sts_eNotify	Alarm status from outer loop: 0 = Not in alarm, acknowledged, 1 = Not in alarm, unacknowledged or reset required, 2 = Low severity alarm, acknowledged, 3 = Low severity alarm, unacknowledged, 4 = Medium severity alarm, acknowledged, 5 = Medium severity alarm, unacknowledged, 6 = High severity alarm, acknowledged, 7 = High severity alarm, unacknowledged, 8 = Urgent severity alarm, acknowledged, 9 = Urgent severity alarm, unacknowledged.

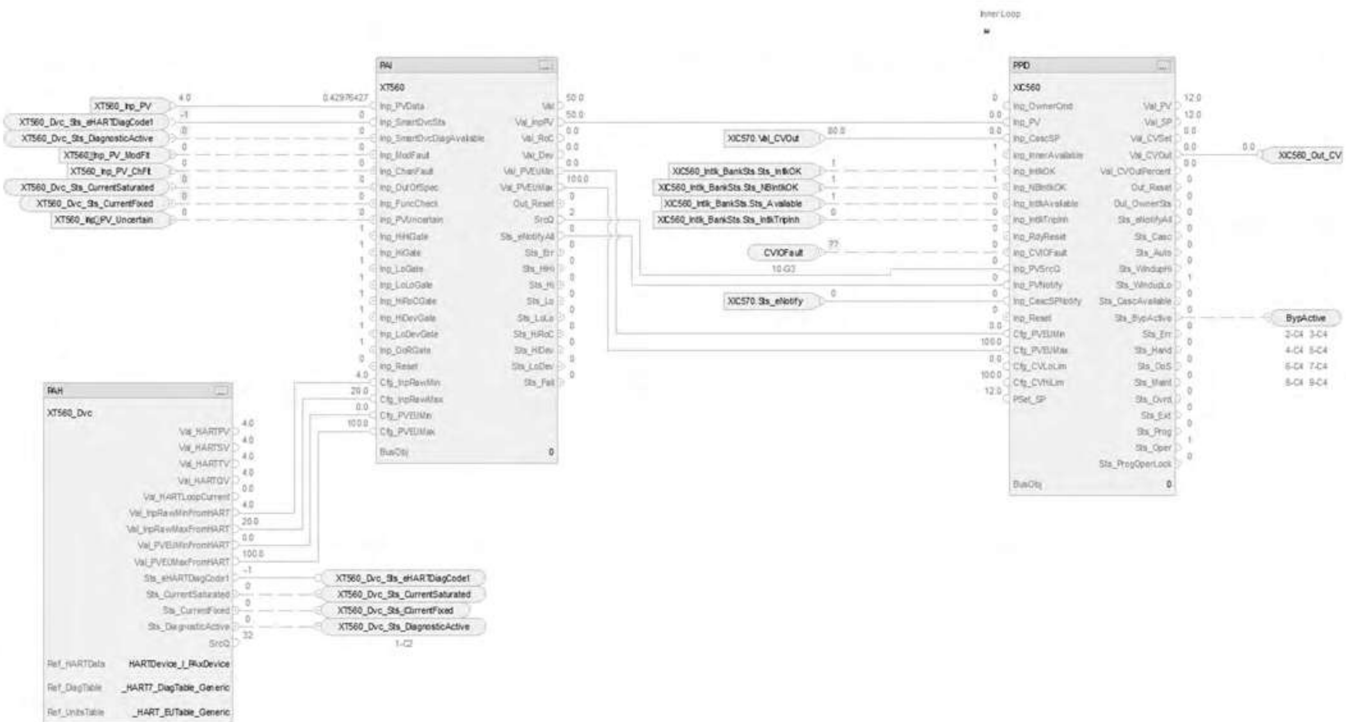
PPID Output References

Parameter	Description
XIC760_Out_CV	Control Variable output Loop CV after clamping and ramping (CVEU)
BypActive	Output connection to interlock bank sheet

PPID Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_PID	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

CS_PPID_CASC - Inner Loop HART Sheet



The CS_PPID_CASC -Inner Loop HART sheet operates the same as the CS_PPID_CASC - Inner Loop but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute for XT560 for XT100
- For more information, see HART Integration on page 31.

Interlock Bank Inner Loop Sheet

PINTLK Input Reference

Parameter	Description
BypActive	Input connection from CS_PPID_CASC - Inner Loop sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

IO Faults Inner Loop Sheet

Faults Input References

Parameter	Description
XIC760_Out_CV_ChFlt	Channel fault 1 = I/O channel fault or failure 0 = OK
XIC760_Out_CV_ModFlt	Module fault 1 = I/O module failure or module communication status bad 0 = OK

Fault Output Reference

Parameter	Description
CVIOFault	Output connection to CS_PPID_CASC Inner Loop sheet

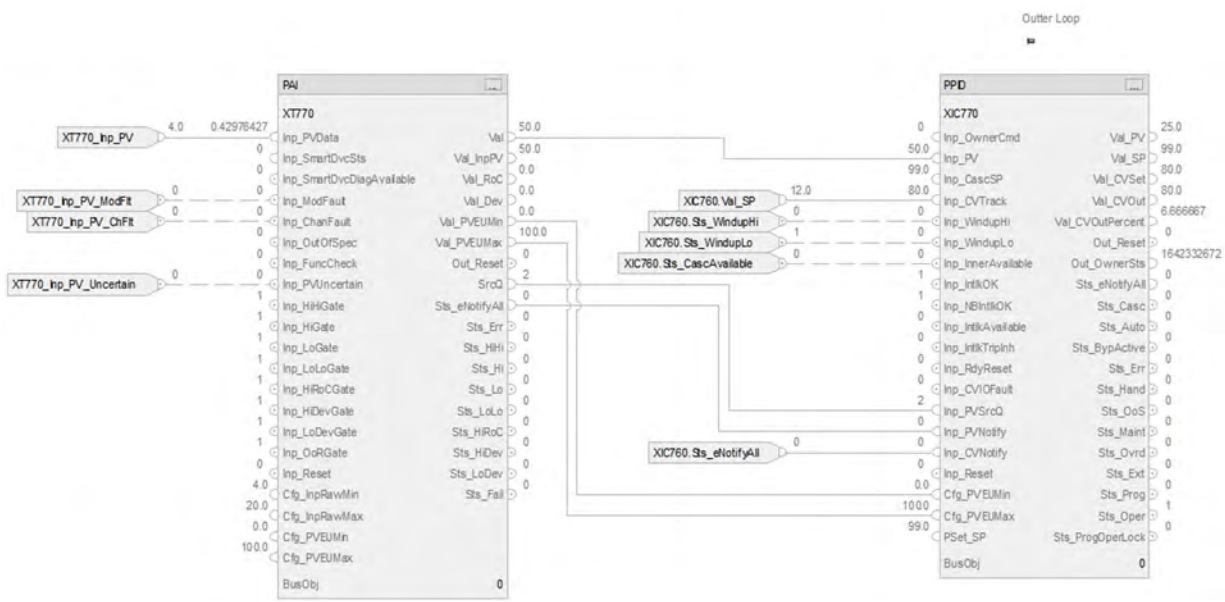
For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

ROUTINE Outer Loop (XIC770 analog and XIC570 HART)

Each routine contains one sheet

Sheet	Description
CS_PPID_CASC – Outer Loop	PPID outer loop • XIC770 analog • XIC570 HART

CS_PPID_CASC-Outer Loop Sheet



Input References to PAI

See CS_PAI Sheet on page 71 for details.

- Substitute XT770 for every instance of XT101

PAI Outputs to PPID Inputs

Parameter	Description
Val	Value for PPID Inp_PV parameter Process Variable (PVEU)
Val_PVEUmin	Value for PPID Cfg_PVEUMin parameter PV minimum value for scaling from engineering units to %, PV at 0% (PVEU). Valid any float less than Cfg_PVEUMax.
Val_PVEUMax	Value for PPID Cfg_PVEUMax parameter PV maximum value for scaling from engineering units to %, PV at 100% (PVEU). Valid any float greater than Cfg_PVEUMin.
SrcQ	Value for PPID Inp_PVSrcQ parameter Inp_PV source status and quality: 0 = Good, live, confirmed good 1 = Good, live, assumed good 2 = Good, no feedback, assumed good 8 = Test, simulated 9 = Test, loopback 10 = Test, manually entered 16 = Uncertain, live, off-spec 17 = Uncertain, substituted at device 18 = Uncertain, substituted at instruction 19 = Uncertain, using last known good 20 = Uncertain, using replacement value 32 = Bad, signal failure 33 = Bad, channel fault 34 = Bad, module/communications fault 35 = Bad, invalid configuration
Sts_eNotifyAll	Value for PPID Inp_PVNotify parameter Related PV object alarm priority and acknowledgment status: 0 = Not in alarm, acknowledged 1 = Not in alarm, unacknowledged or reset required 2 = Low severity alarm, acknowledged 3 = Low severity alarm, unacknowledged 4 = Medium severity alarm, acknowledged 5 = Medium severity alarm, unacknowledged 6 = High severity alarm, acknowledged 7 = High severity alarm, unacknowledged 8 = Urgent severity alarm, acknowledged 9 = Urgent severity alarm, unacknowledged

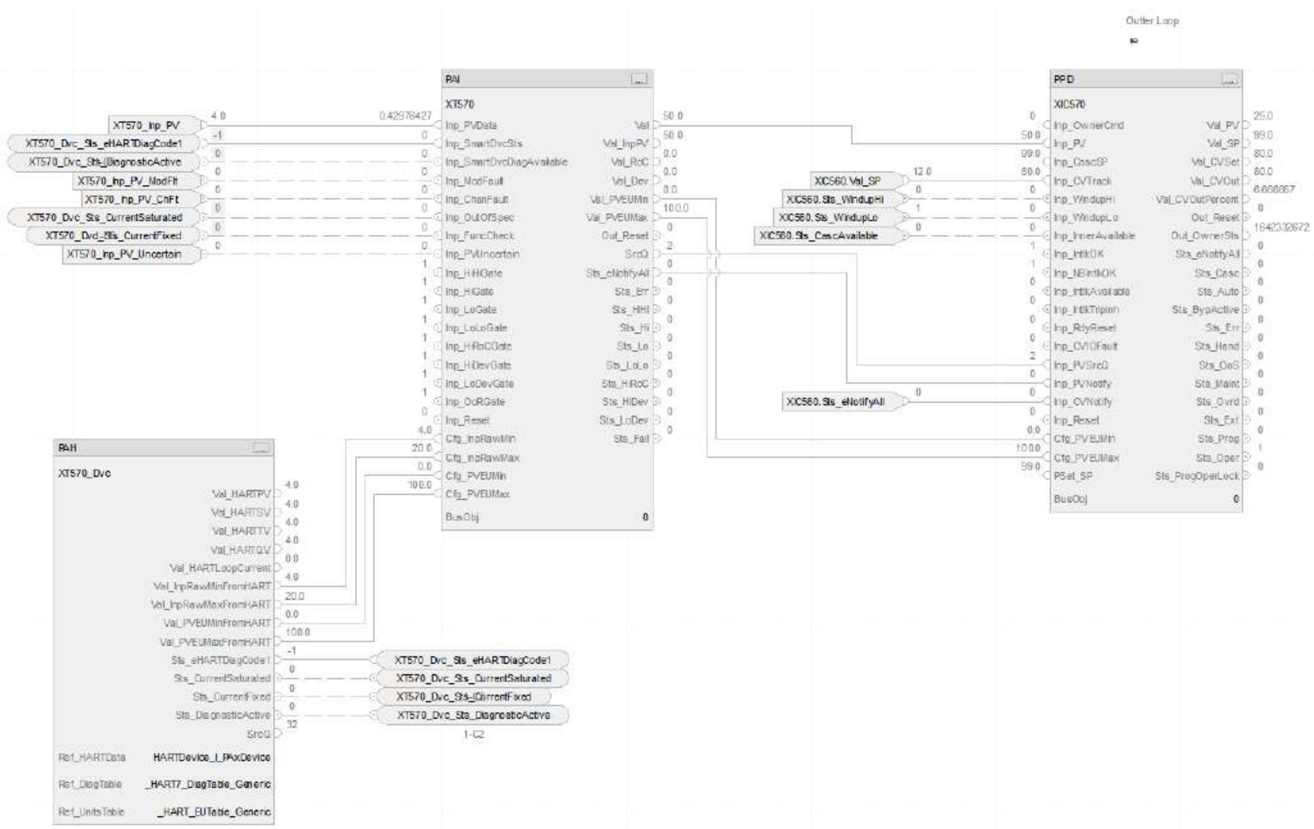
PPID Input References

Parameter	Description
XIC760.Inp_CVTrack	Inner loop CV to track if Cfg_UseCVTrack = 1 or if Inp_InnerAvailable = 0 (CVEU)
XIC760.Sts_WindupHi	1 = The inner loop winding up High, usually connects to Inp_WindupHi of outer loop
XIC760.Sts_WindupLo	1 = The inner loop winding up Low, usually connects to Inp_WindupLo of outer loop
XIC760.Sts_CascAvailable	1 = Inner loop is available. 0 = Inner loop is not available, PPID tracks Inp_CVTrack, typically inner loop SP or actuator position.
XIC760.Sts_eNotifyAll	Alarm status from inner loop: 0 = Not in alarm, acknowledged 1 = Not in alarm, unacknowledged or reset required 2 = Low severity alarm, acknowledged 3 = Low severity alarm, unacknowledged 4 = Medium severity alarm, acknowledged 5 = Medium severity alarm, unacknowledged 6 = High severity alarm, acknowledged 7 = High severity alarm, unacknowledged 8 = Urgent severity alarm, acknowledged 9 = Urgent severity alarm, unacknowledged

PPID Configuration Considerations

Operand	Type	Description
PlantPAx control	P_PID	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

CS_PPID_CASC - Outer Loop HART Sheet



The CS_PPID_CASC - Outer Loop HART sheet operates the same as the CS_PPID_CASC - Outer Loop sheet but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute for XT570 for XT100
- For more information, see HART Integration on page 31.

Notes:

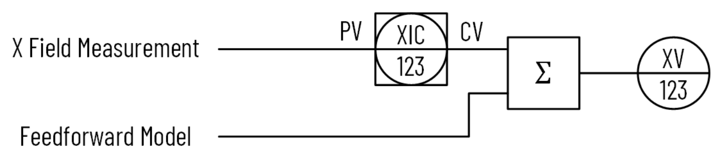
Process Proportional + Integral + Derivative (PPID) Feedforward Control Strategies

Use the PPID Feedforward control strategy when feedback control (PPID control strategy) alone is not adequate to maintain the process variable at the setpoint. Rather than rely on feedback to make corrective changes to a process only after some load change has driven the process variable away from setpoint, control schemes with feedforward monitor the relevant load(s) and use that information to preemptively make stabilizing changes to the final control element such that the process variable will not be affected.

To scale the CV to align with the associated IO module channel range or to accommodate a fail-open (FO) valve (or air to close) use either of the following options:

- Use a basic PPID with Analog Output control strategy
- Insert a scalar instruction between the PPID CV and the analog output channel reference

The PPID Feedforward control strategies are pre-configured to enable controller actions (CV Action, SP Action and Loop Mode Action) based on various shed conditions (Interlock trip, CV fail, PV fail, and SP fail).

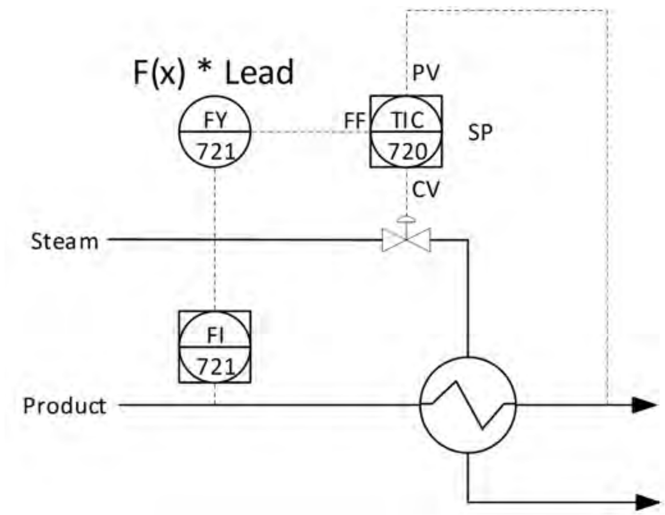


PPID Feedforward Example

Consider a control system that manipulates steam flow to a heat exchanger to maintain the discharge temperature of the product at a constant setpoint value. The outlet temperature suffers temporary deviations from setpoint if load conditions change. The feedback control system can eventually bring the exiting product's temperature back to setpoint, but it cannot begin corrective action until after a load has driven the product temperature away from the setpoint. To improve control, build both feedforward action and feedback action into the design. The feedforward action lets the control system take corrective action in response to load changes before the process variable is affected.

In this example, the dominant load in the system is product flow rate. Adapting this control system to include feedforward requires installing a product flow transmitter that is characterized to provide feedforward action to the PID controller maintaining temperature. With feedforward control action in place, the steam flow rate immediately changes with product flow rate, preemptively compensating for the increased or decreased heat demand of the product.

The feedforward component of the strategy directly affects the steam valve position in response to product flow. However, the temperature response to the manipulation of the PPID output generally includes a process lag. To overcome the process lag, the feedforward model typically includes a lead function.



When the product flow rate to this heat exchanger suddenly increases, the lead function adds a surge to the feedforward signal, quickly opening the steam valve sending a surge of steam to the exchanger to help overcome the process response lag. The feedforward action is not perfect with this lead function added, but it is substantially improved.

The following PPID control strategies are available as routines in the process library:

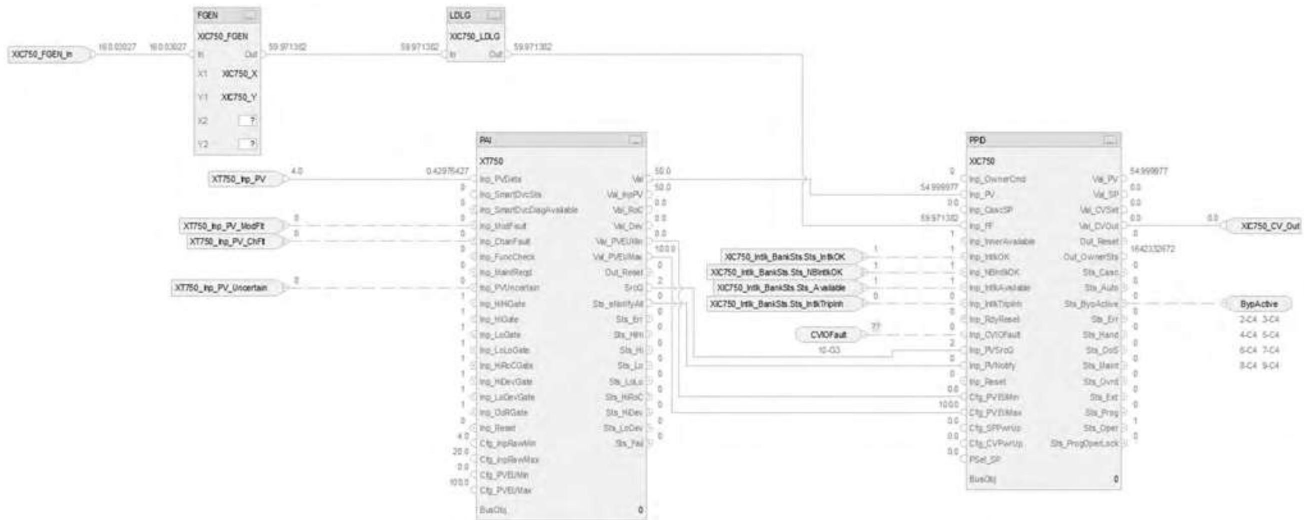
- CS_PPID_FF
- CS_PPID_FF_HART

Import the appropriate control strategy as a **routine** in your controller project.

The PPID Feedforward control strategy contains these sheets:

Sheet	Description
CS_PPID_FF	PPID instruction
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The PPID instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder
IO Faults	The logic monitors Control Variable faults.

CS_PPID_FF Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.

- Substitute XIC750 for the PV data instance of XT101
- Substitute XT750 for the remaining instances of XT101

PAI Outputs to PPID Inputs

Parameter	Description
Val	Value for PPID Inp_PV parameter Process Variable (PVEU)
Val_PVEUmin	Value for PPID Cfg_PVEUMin parameter PV minimum value for scaling from engineering units to %, PV at 0% (PVEU). Valid any float less than Cfg_PVEUMax.
Val_PVEUMax	Value for PPID Cfg_PVEUMax parameter PV maximum value for scaling from engineering units to %, PV at 100% (PVEU). Valid any float greater than Cfg_PVEUMin.
SrcQ	Value for PPID Inp_PVSrcQ parameter Inp_PV source status and quality: 0 = Good, live, confirmed good 1 = Good, live, assumed good 2 = Good, no feedback, assumed good 8 = Test, simulated 9 = Test, loopback 10 = Test, manually entered 16 = Uncertain, live, off-spec 17 = Uncertain, substituted at device 18 = Uncertain, substituted at instruction 19 = Uncertain, using last known good 20 = Uncertain, using replacement value 32 = Bad, signal failure 33 = Bad, channel fault 34 = Bad, module/communications fault 35 = Bad, invalid configuration
Sts_eNotifyAll	Value for PPID Inp_PVNotify parameter Related PV object alarm priority and acknowledgment status 0 = Not in alarm, acknowledged 1 = Not in alarm, unacknowledged or reset required 2 = Low severity alarm, acknowledged 3 = Low severity alarm, unacknowledged 4 = Medium severity alarm, acknowledged 5 = Medium severity alarm, unacknowledged 6 = High severity alarm, acknowledged 7 = High severity alarm, unacknowledged 8 = Urgent severity alarm, acknowledged 9 = Urgent severity alarm, unacknowledged

PPID Input References

Parameter	Description
XIC750_FGEN_In	FeedForward term (CVEU). Valid any float between -(Cfg_CVEUMax-Cfg_CVEUMin) and (Cfg_CVEUMax-Cfg_CVEUMin). Default is 0.0.
XIC750_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
XIC750_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
XIC750_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
XIC750_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
CVIOFault	Input connection from IO Faults sheet
XIC750_SPPwrUp	Loop SP on power-up (PVEU) used when Cfg_PwrUpLM is not 0. The value is clamped to the SP range (Cfg_SPLoLim, Cfg_SPHiLim). Valid any float between Cfg_PVEUMin and Cfg_PVEUMax.
XIC750_CWPwrUp	Loop CV on power-up (CVEU) used when Cfg_PwrUpLM is not 0. Value can be clamped to the configured limits (Cfg_CVLoLim, CfgHiLim) in cascade or auto, and in manual if so configured. Valid any float between Cfg_CVEUMin and Cfg_CVEUMax.
XIC750_PSet_SP	Program setting for SP, loop mode Auto (PVEU). Valid any float. Default is 0.0.

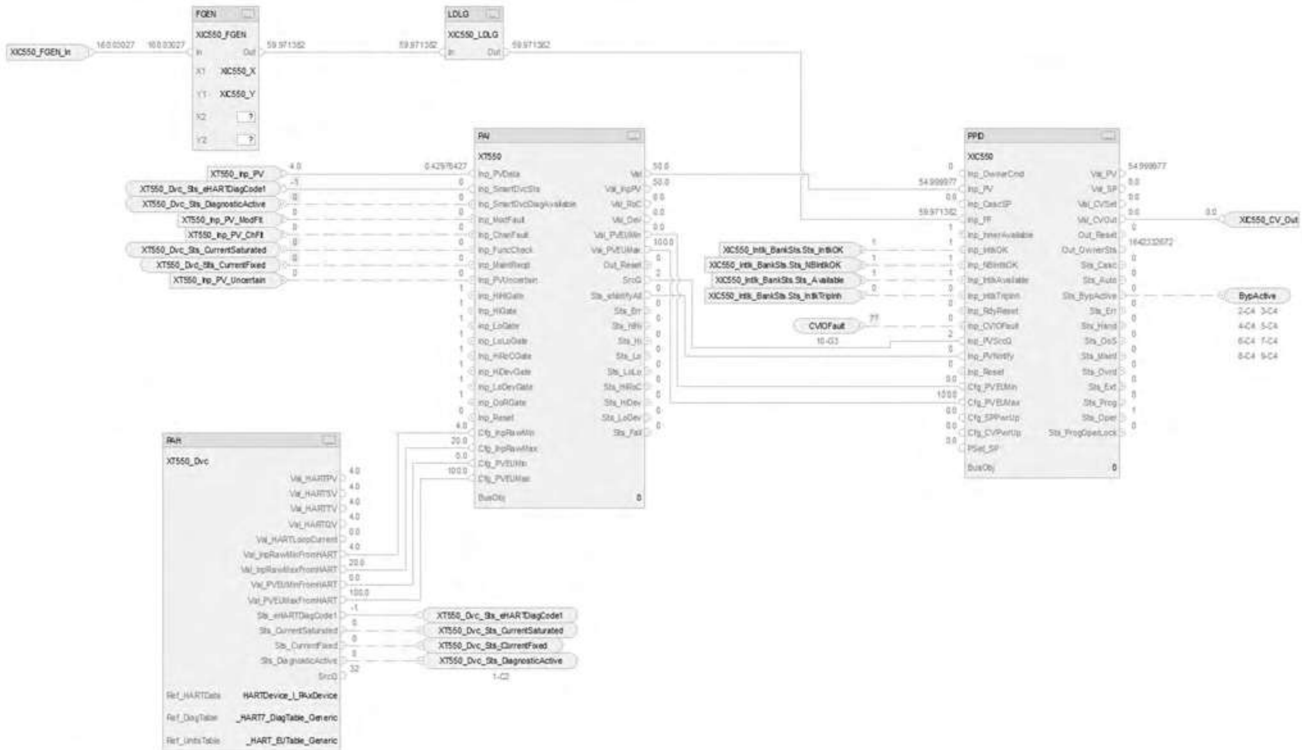
PPID Output References

Parameter	Description
XIC750_CV_Out	Control Variable output Loop CV after clamping and ramping (CVEU)
BypActive	Output connection to interlock bank sheet

PPID Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_PID	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

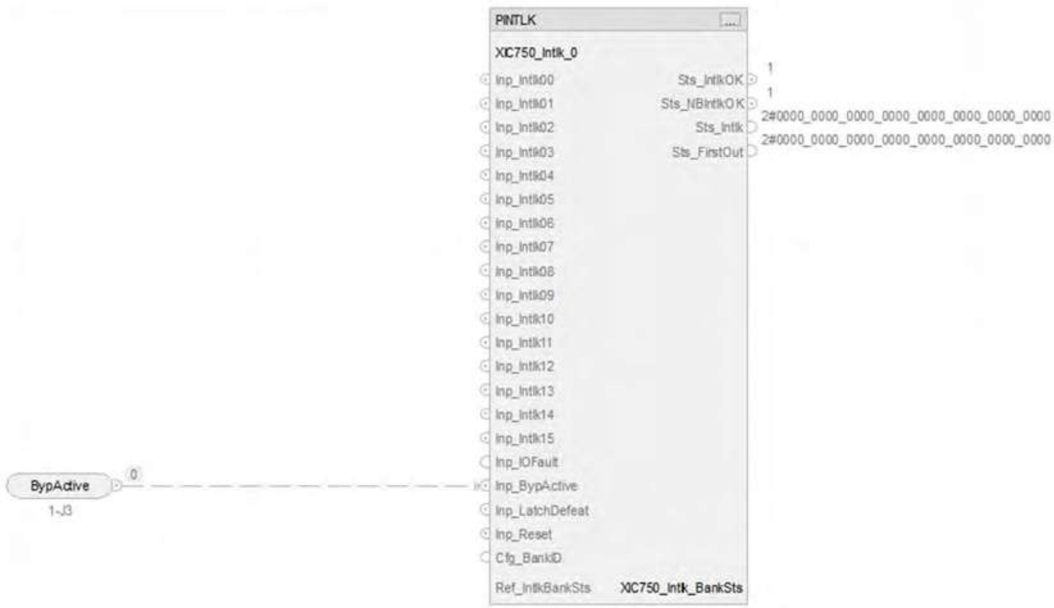
CS_PPID_FF HART Sheet



The CS_PPID_FF HART sheet operates the same as the CS_PPID_FF sheet but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute XIC550 for the PV data instance of XT101
- Substitute XT550 for the remaining instances of XT101
- For more information, see HART Integration on page 31.

Interlock Bank Sheet



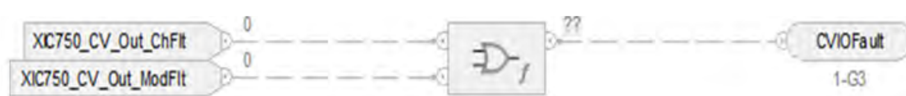
PINTLK Input Reference

Parameter	Description
BypActive	Input connection from CS_PPID_FF sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

IO Faults Sheet



Fault Input References

Parameter	Description
XIC750_Out_CV_ChFlt XIC550_Out_CV_ChFlt	Channel fault 1 = I/O channel fault or failure 0 = OK Source: PAI instruction
XIC750_Out_CV_ModFlt XIC550_Out_CV_ModFlt	Module fault 1 = I/O module failure or module communication status bad 0 = OK Source: PAI instruction

Fault Output Reference

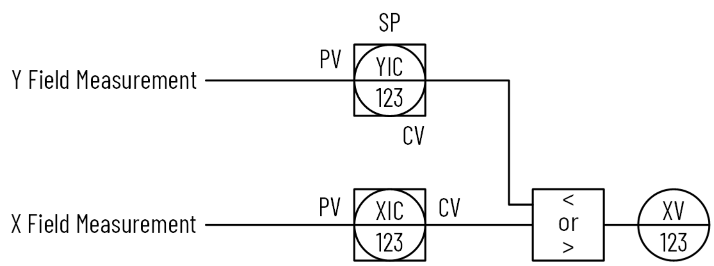
Parameter	Description
CVIOFault	Output connection to CS_PPID_FF sheet

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

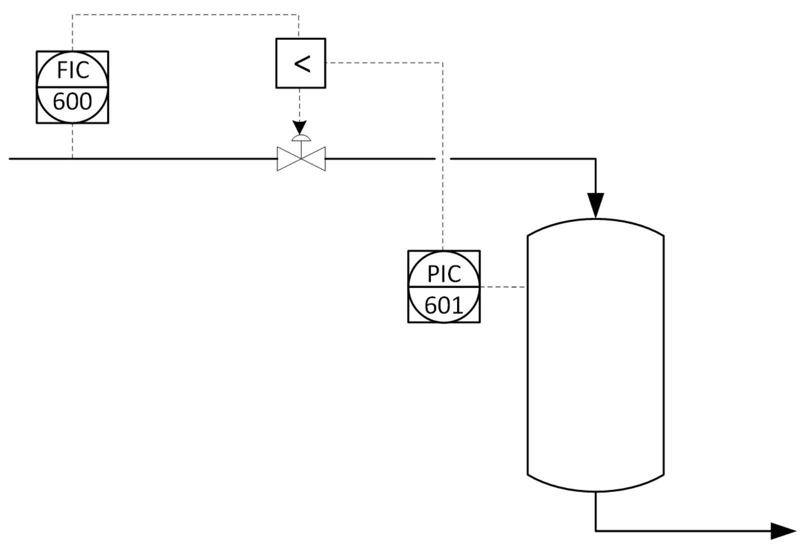
Process Proportional + Integral + Derivative (PPID) Override Control Strategies

The PPID Override control strategy selectively chooses the output of up to six PPID controllers based on configuration (High Select or Low Select) to drive an analog output device. The output(s) of the 'unselected' PPID controller(s) are kept within Kp*Error of the active PPID controller output to help ensure a quick response when another PPID's output becomes the limiting output.

To scale the CV to align with the associated I/O module channel range or to accommodate a fail-open (FO) valve (or air to close) use either of the following options:
- Use a basic PPID with Analog Output control strategy
- Insert a scalar instruction between the PPID CV and the analog output channel reference



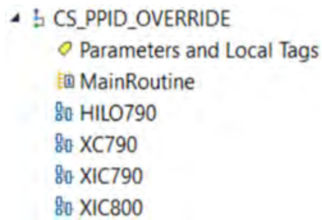
PPID Override Example



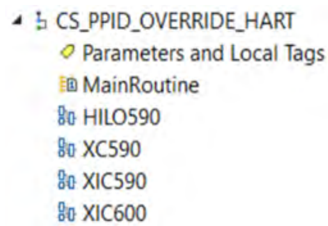
In this example, the primary control maintains a desired flow of product (FIC-600) into the vessel. To maintain the vessel integrity, it is desired to keep the vessel pressure below a set value. This is accomplished by using the override control strategy where the vessel pressure controller (PIC-601) restricts the valve opening if the vessel pressure exceeds the set value. The lower of the two CVs is selected to drive the final control element (FV-600).

These PPID Override control strategies (consisting of multiple routines) are available in the process library:

- CS_PPID_OVERRIDE



- CS_PPID_OVERRIDE



Import the **routines** for the appropriate control strategy in your controller project. Each control strategy contains multiple routines; each routine contains multiple Function Block sheets.

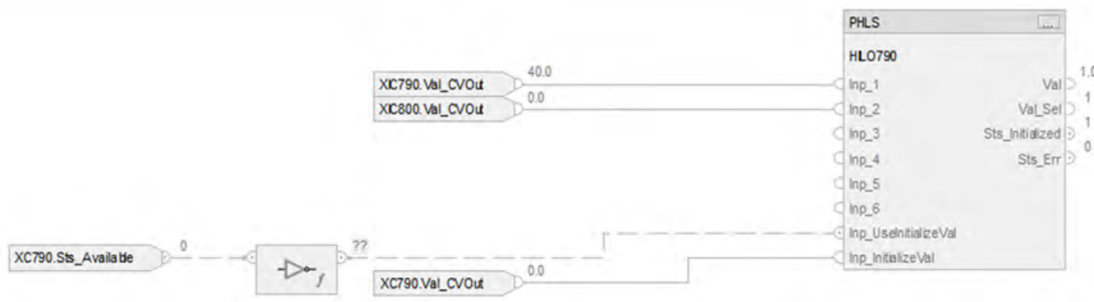
Each PPID Override control strategy contains these Routines:

- Override Low Select (HILO790 analog/HILO590 HART)
- The PHLS can be configured as High or Low Select within the instruction as required
- Process Analog Output (XC790 analog/XC590 HART)
- Process Analog Input to Process PID (XIC790 analog/XIC590 HART)
- Process Analog Input to Process PID (XIC800 analog/XIC600 HART)

ROUTINE Override Low Select

Sheet	Description
CS_PHLS	Process High or Low Selector instruction • HILO790 analog • HILO590 HART

CS_PHLS Sheet



The control strategy, as supplied, uses only two PID control loops. The control strategy can support up to six PID control loops by exposing additional inputs of the PHLS instruction:

- XIC790 (Inp_1) and XIC800 (Inp_2) for the analog option
- XIC590 (Inp_1) and XIC600 (Inp_2) for the HART option

The control strategy uses a subset of the PHLS control strategy. See Process High or Low Selector (PHLS) Control Strategies on page 115 for more details.

PHLS Input References

Parameter	Description
XIC790_Val_CVOut	Control Variable output 1 Source: Val_CVOut from PID loop 1 (XIC790)
XIC800_Val_CVOut	Control Variable output 2 Source: Val_CVOut from PID loop 1 (XIC800)
XC790_Sts_Available	Status available of downstream PAO 1 = Instruction is initialized
XC790_Val_CVOut	Control Variable output as initialization value from downstream block.

PHLS Output Reference

Parameter	Description
HILO790_Val	Control Variable output (selected minimum or maximum) for downstream block Destination: PAO input (PSet_CV)

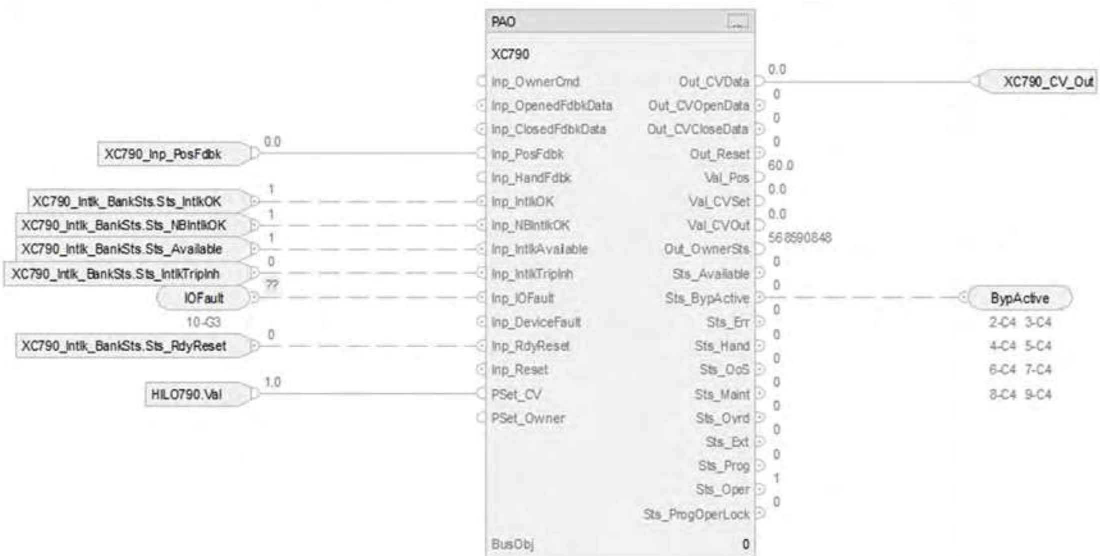
PHLS Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_HIGH_LOW_SELECT	Data structure required for proper operation of instruction

ROUTINE Process Analog Output (XC790 Analog/ XC590 HART)

Sheet	Description
CS_PAO	Process High or Low Selector instruction • XC790 analog • HC590 HART
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The PAO instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder
IO Faults	The logic monitors one analog output channel for I/O fault input and raises an alarm on an I/O fault.

CS_PAO Sheet



Input References to PAO and PAO Output References

See the Process Analog Output (PAO) Control Strategies on page 83 for details.

Substitute:

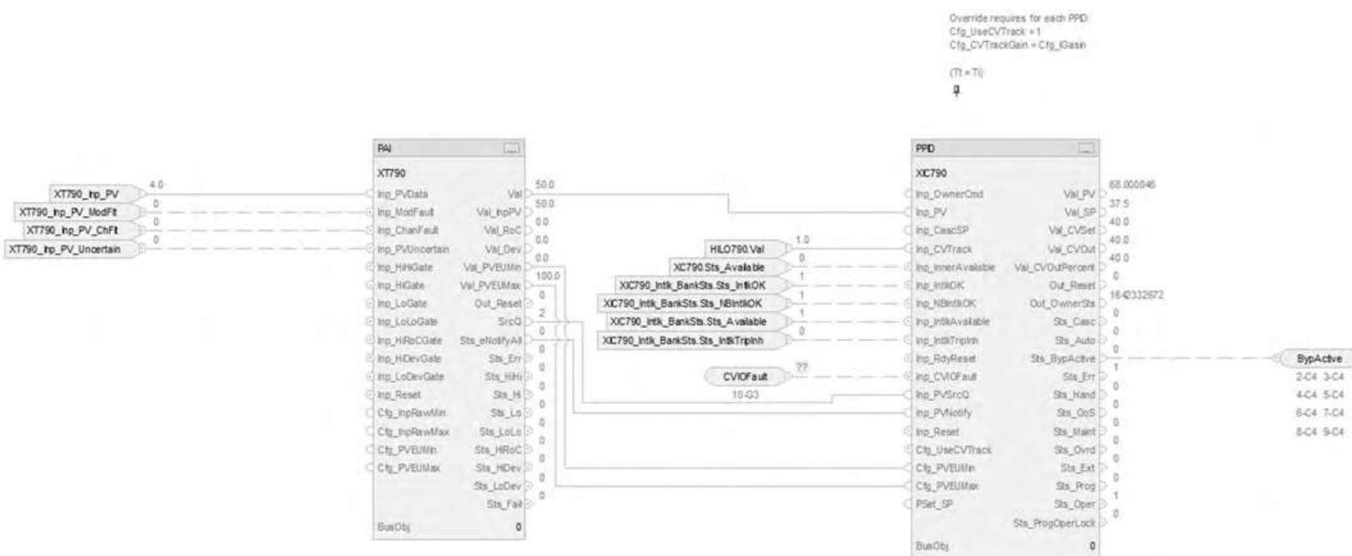
Analog Input	HART Input
XC790 for XC100	XC590 for XC101
HILO790 for XC100	HILO790 for XC100

ROUTINE: Process Analog Input to Process PID

There are two routines; each routine contains these sheets.

Sheet	Description
CS_PPID	Process PID instruction with override • XIC790/XIC800 analog • XIC590/XIC600 HART
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The PPID instruction monitors interlock conditions that cause output CV and SP to shed. CV shed can be configured to hold the last good CV value or to use the configured safe value. SP is shed to current PV. There are 8 interlock bank sheets; each bank exposes 16 interlocks but supports as many as 32 interlocks. Use those sheets and interlocks that you need; delete the remainder
IO Faults	The logic monitors Control Variable faults.

CS_PPID Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.

- Substitute XIC790 for the first instance of XT101
- Substitute XIC800 for the second instance of XT101

PAI Outputs to PPID Inputs

Parameter	Description
Val	Value for PPID Inp_PV parameter Process Variable (PVEU)
Val_PVEUmin	Value for PPID Cfg_PVEUMin parameter PV minimum value in engineering units (PVEU). Valid any float less than Cfg_PVEUMax.
Val_PVEUMax	Value for PPID Cfg_PVEUMax parameter PV maximum value in engineering units (PVEU). Valid any float greater than Cfg_PVEUMin.
SrcQ	Value for PPID Inp_PVSrcQ parameter Inp_PV source status and quality: 0 = Good, live, confirmed good 1 = Good, live, assumed good 2 = Good, no feedback, assumed good 8 = Test, simulated 9 = Test, loopback 10 = Test, manually entered 16 = Uncertain, live, off-spec 17 = Uncertain, substituted at device 18 = Uncertain, substituted at instruction 19 = Uncertain, using last known good 20 = Uncertain, using replacement value 32 = Bad, signal failure 33 = Bad, channel fault 34 = Bad, module/communications fault 35 = Bad, invalid configuration
Sts_eNotifyAll	Value for PPID Inp_PVNotify parameter Related PV object alarm priority and acknowledgment status: 0 = Not in alarm, acknowledged 1 = Not in alarm, unacknowledged or reset required 2 = Low severity alarm, acknowledged 3 = Low severity alarm, unacknowledged 4 = Medium severity alarm, acknowledged 5 = Medium severity alarm, unacknowledged 6 = High severity alarm, acknowledged 7 = High severity alarm, unacknowledged 8 = Urgent severity alarm, acknowledged 9 = Urgent severity alarm, unacknowledged

PPID Input References

Parameter	Description
HILO790_Val HILO800_Val	Control Variable output CV to track if Cfg_UseCVTrack = 1 or if Inp_InnerAvailable = 0 (CVEU). Valid any float. Source: PIC control loop
XC790_Sts_Available	Status available
XC800_Sts_Available	1 = Inner loop is available 0 = Inner loop is not available
XIC790_Intlk_BankSts.Sts_IntlkOK XIC800_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
XIC790_Intlk_BankSts.Sts_NBIntlkOK XIC800_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
XIC790_Intlk_BankSts.Sts_Available XIC800_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
XIC790_Intlk_BankSts.Sts_IntlkTripInh XIC800_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
CVIOFault	Input connection from IO Faults sheet

PPID Output References

Parameter	Description
XIC790_Val_CVOut	Control Variable output Loop CV after clamping and ramping (CVEU) Destination: Analog output channel or downstream REAL tag
BypActive	Output connection to interlock bank sheet

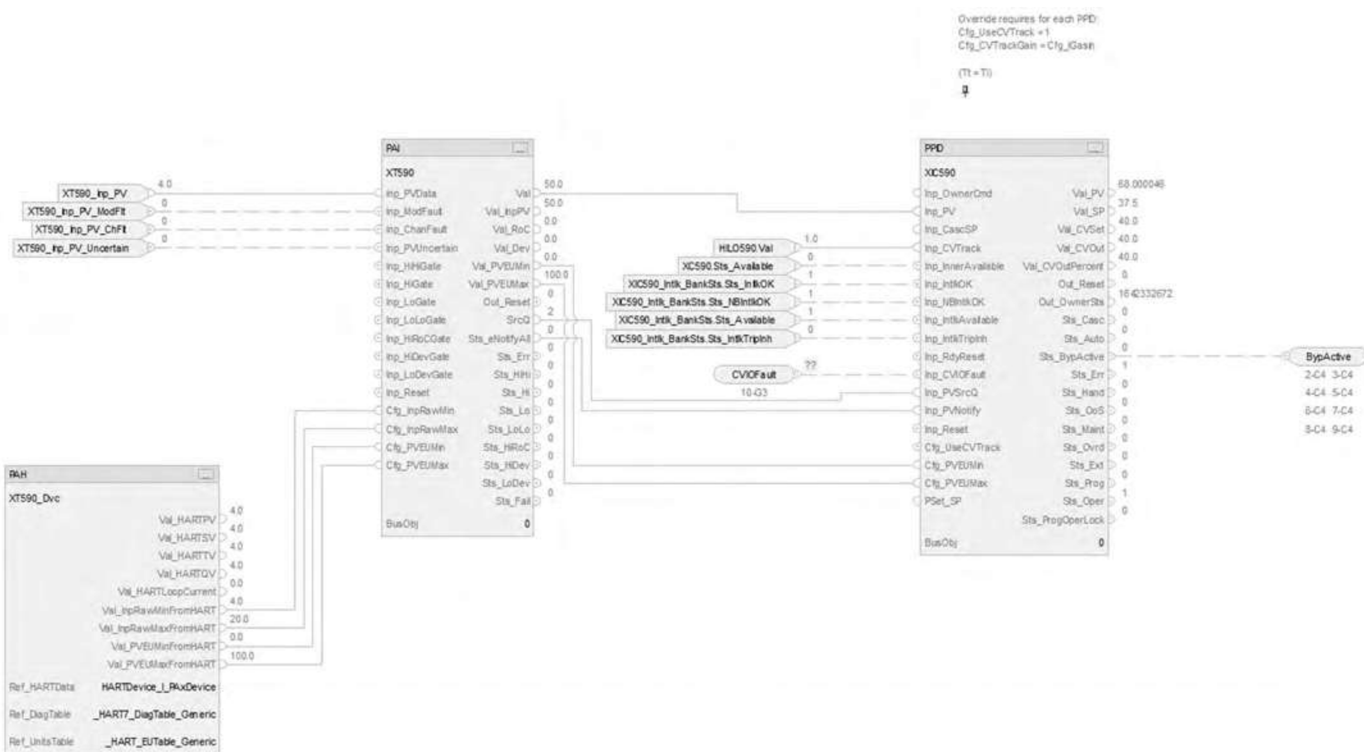
PPID Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_PID	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

Override requires these additional configuration operands for each PPID:

Parameter	Description
Cfg_UseCVTrack	For each PID control loop, set Cgf_UseCVTrack=1 1 = Use Inp_CVTrack reset feedback in tracking, for example, if PPID output is significantly faster than actuator or inner loop or in override select control
Cfg_CVTrackGain	For each PID control loop, set Cfg_CVTrackGain=Cfg_IGain (Tt=Ti) Tracking gain Kt (1/minute) for independent or tracking time constant Tt (minutes) for dependent gains for CV to track Inp_CVTrack if Cfg_UseCVTrack = 1.

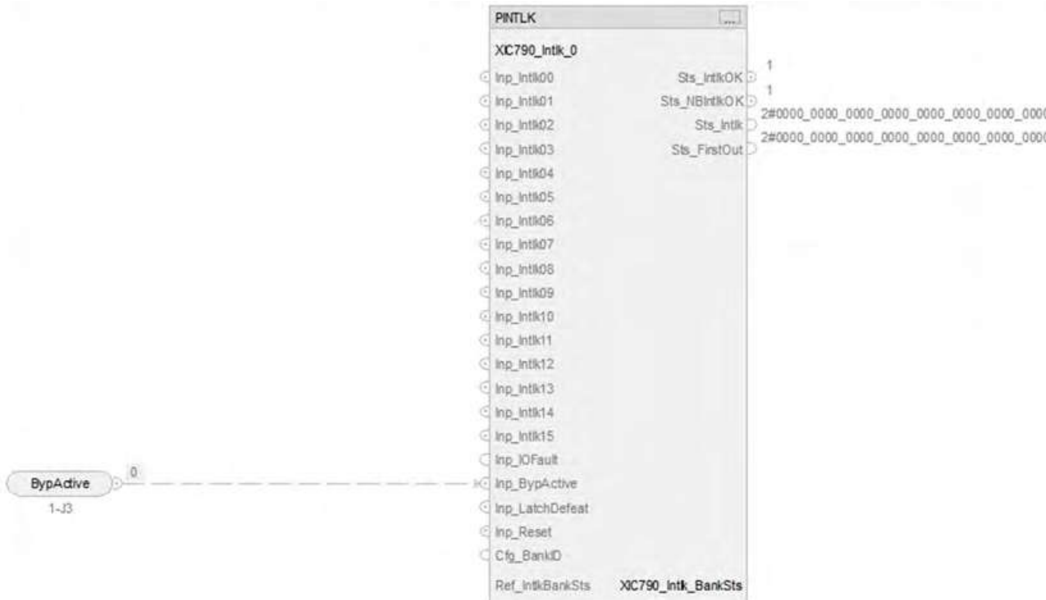
CS_PPID HART Sheet



The CS_PPID HART sheet operates the same as the CS_PPID sheet but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute XIC590 for the first instance of XT100
- Substitute XIC600 for the second instance of XT100
- For more information, see HART Integration on page 31.

Interlock Bank Sheet



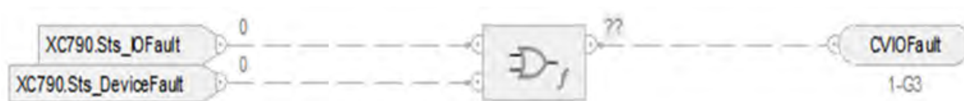
PINTLK Input Reference

Parameter	Description
BypActive	Input connection from CS_PPID_OVERRIDE sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

IO Faults Sheet



Faults Input References

Parameter	Description
XIC790_Out_CV_ChanFault	Control Variable output channel fault
XIC790_Out_CV_ModFault	Control Variable output data module fault

Fault Output Reference

Parameter	Description
CVIOFault	Output connection to CS_PPID_OVERRIDE sheet

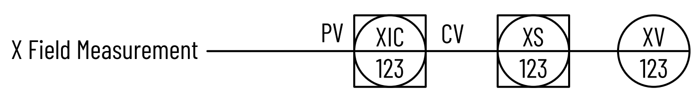
For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

Process Proportional + Integral + Derivative (PPID) Basic PPID with Process Analog Output (PAO) Control Strategies

This control strategy differs from the Basic PPID control strategy in that a PAO instruction is inserted between the output of the PPID and the reference signal to the analog output channel.

This strategy provides the capability to pulse outputs (pulse open and pulse close) to position a valve to the reference signal provided by the PPID CV.

You can also use this control strategy instead of the Basic PPID control strategy to scale the output to accommodate fail-open (FO) valves (or air to close). Alternatively, you can use a scalar instruction in place of the PAO to accommodate fail-open valves.

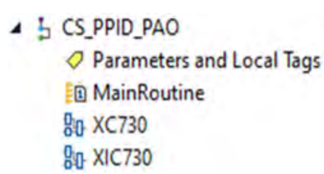


The following PPID control strategies are available as programs in the process library:

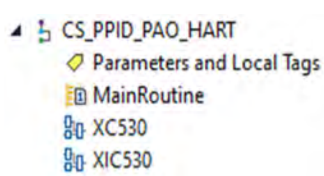
- CS_PPID_PAO
- CS_PPID_PAO_HART

The following PPID with PAO control strategies (consisting of multiple routines) are available in the process library:

- CS_PPID_PAO



- CS_PPID_PAO_HART



Import the **routines** for the appropriate control strategy in your controller project. Each control strategy contains multiple routines; each routine contains multiple Function Block sheets.

Each PPID with PAO control strategy contains these routines:

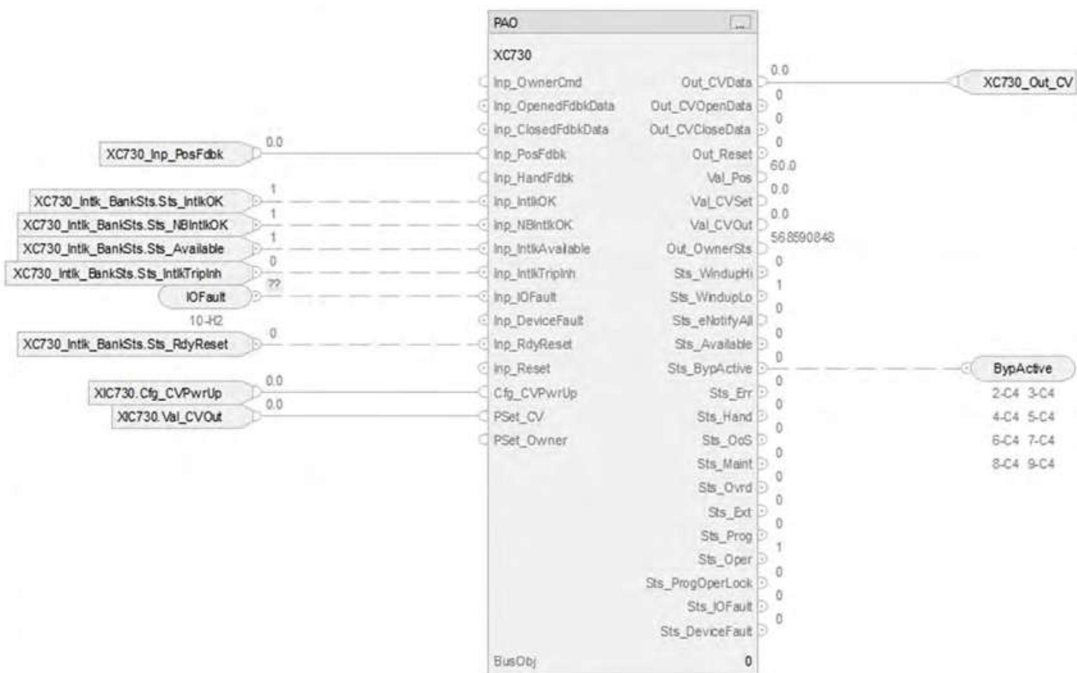
- PAO (XC730 analog/XC530 HART)
- PPID (XIC730 analog/XIC530 HART)

ROUTINE PAO (XC730 analog and XC530 HART)

Each routine contains these sheets.

Sheet	Description
CS_PPID_PAO	Process analog output to PPID XC730 analog XC530 HART
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The PAO instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder.
IO Faults	The logic monitors Control Variable faults.

CS_PAO Sheet



PAO Input References

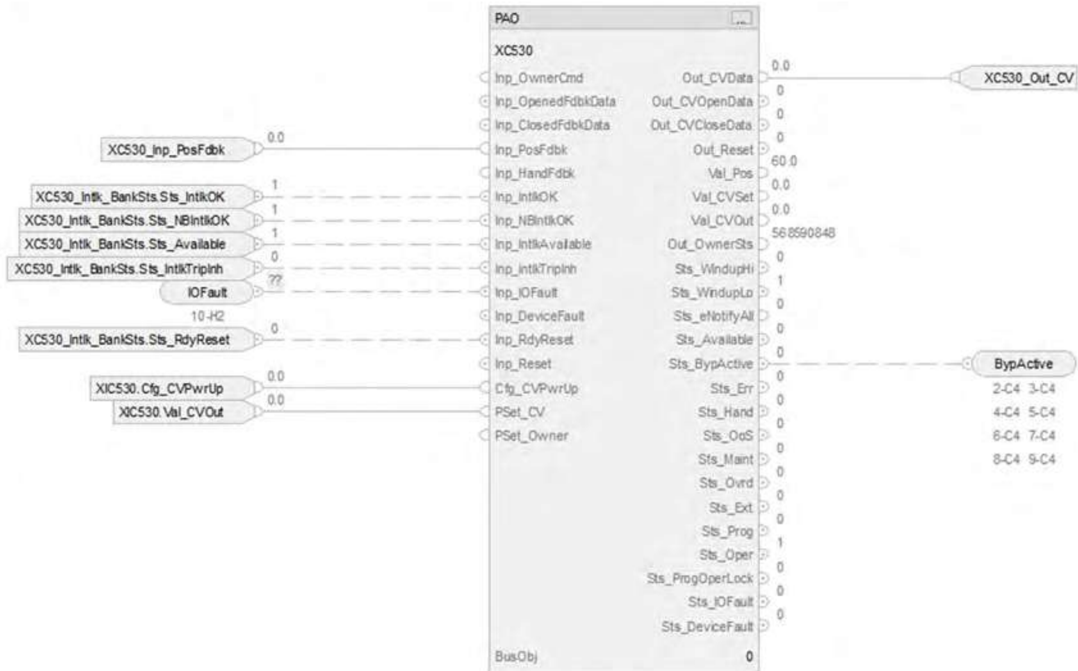
See the Process Analog Output (PAO) Control Strategies for details.

- Substitute XC730 for XC100

PAO Output References

Parameter	Description
XIC730_Out_CV	Control Variable output for PPID instructions CV output in raw (I/O Card) units. Extended properties of this member: Engineering Unit - Raw units (text) used for the analog output
BypActive	Output connection to interlock bank sheet

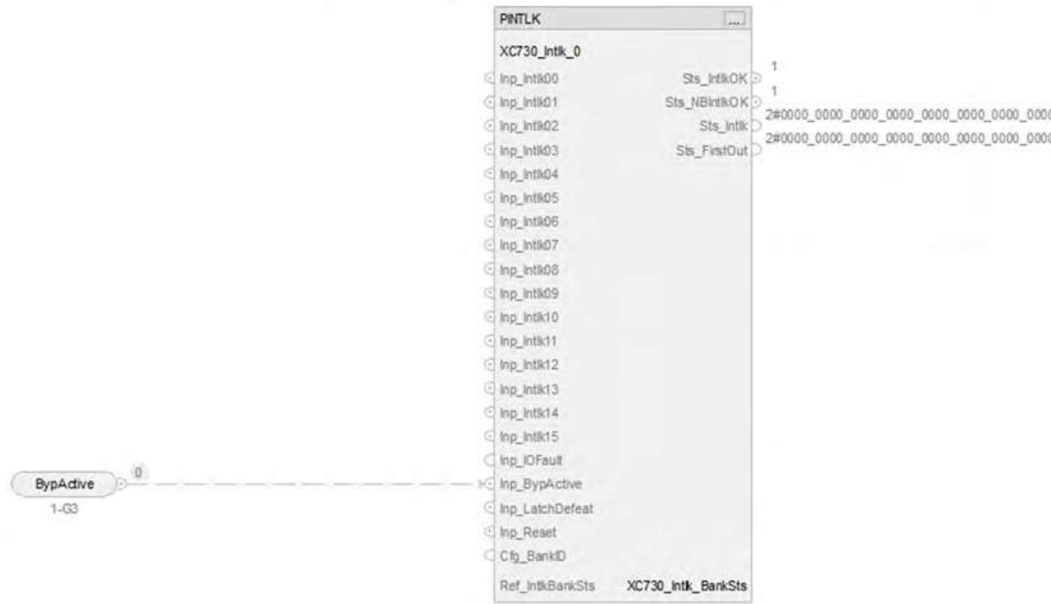
CS_PAO_HART Sheet



The CS_PAO HART sheet operates the same as the CS_PAO sheet but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute for XC530 for XT100
- For more information, see HART Integration on page 31.

Interlock Bank Sheet



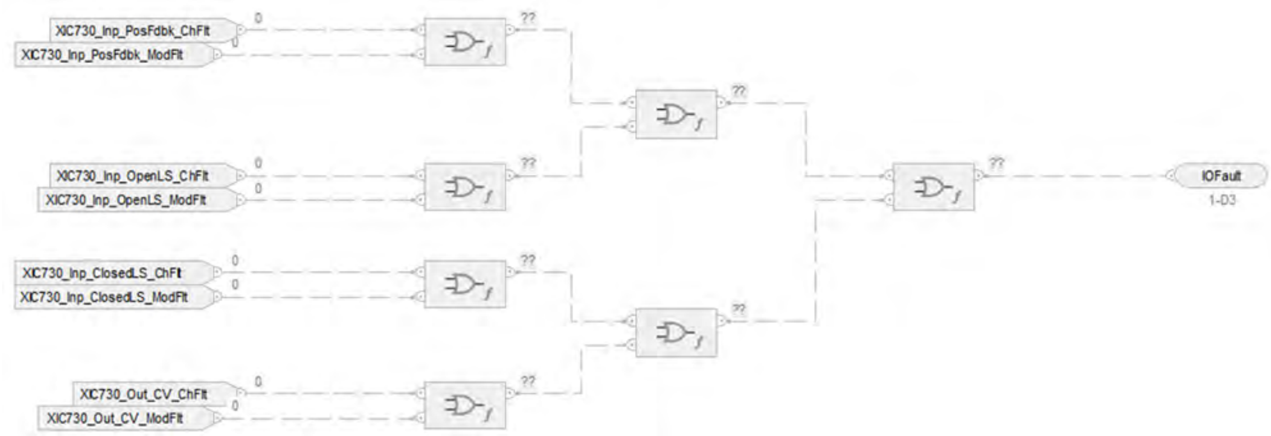
PINTLK Input Reference

Parameter	Description
BypActive	Input connection from CS_PAO sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

IO Faults Sheet



Faults Input References

Parameter	Description
XIC730_Inp_PosFdbk_ChFlt XIC530_Inp_PosFdbk_ChFlt	Tieback Input Channel Fault Source: PAO instruction
XIC730_Inp_PosFdbk_ModFlt XIC530_Inp_PosFdbk_ModFlt	1 = This or parent I/O communication faulted (module not in Running state). Use this parameter to trigger device I/O Fault action. Source: PAO instruction
XIC730_Inp_OpenLS_ChFlt XIC530_Inp_OpenLS_ChFlt	Opened Feedback Input Channel Fault Source: PAO instruction
XIC730_Inp_OpenLS_ModFlt XIC530_Inp_OpenLS_ModFlt	1 = This or parent I/O communication faulted (module not in Running state). Use this parameter to trigger device I/O Fault action. Source: PAO instruction
XIC730_Inp_ClosedLS_ChFlt XIC530_Inp_ClosedLS_ChFlt	Closed Feedback Input Channel Fault Source: PAO instruction
XIC730_Inp_ClosedLS_ModFlt XIC530_Inp_ClosedLS_ModFlt	1 = This or parent I/O communication faulted (module not in Running state). Use this parameter to trigger device I/O Fault action. Source: PAO instruction
XIC730_Out_CV_ChFlt XIC530_Out_CV_ChFlt	Control Variable Output Channel Fault 1 = I/O channel fault or failure 0 = OK Source: PAO instruction
XIC730_Out_CV_ModFlt XIC530_Out_CV_ModFlt	1 = This or parent I/O communication faulted (module not in Running state). Use this parameter to trigger device I/O Fault action. Source: PAO instruction

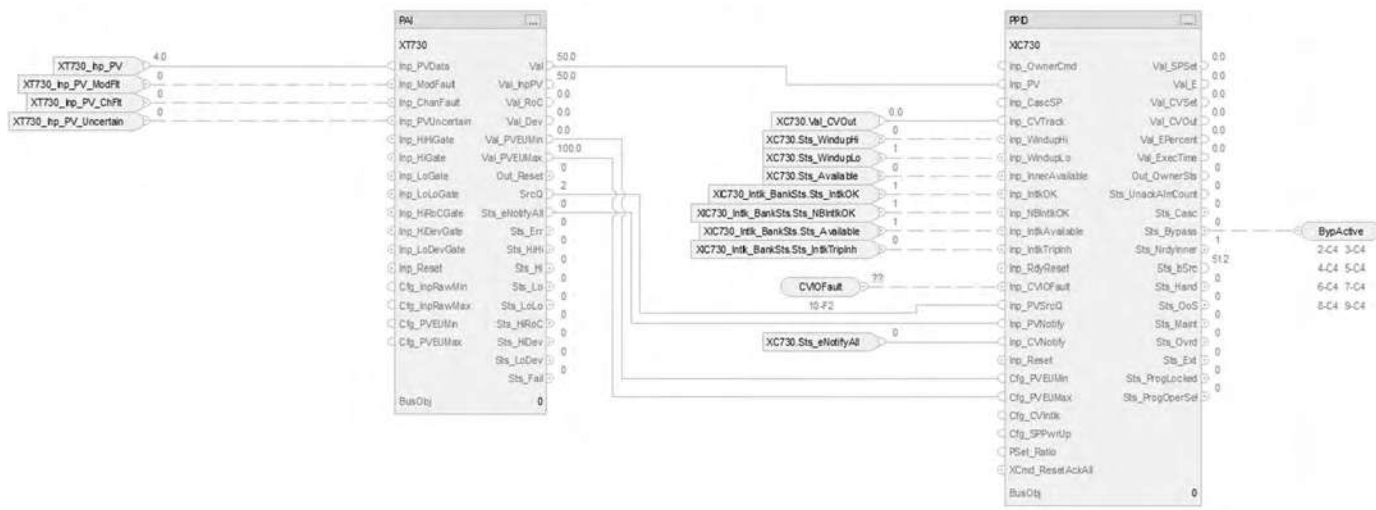
Fault Output Reference

Parameter	Description
IOFault	Output connection to CS_PAO sheet

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

ROUTINE PPID (XIC730 analog/XIC530 HART)

CS_PPID Sheet



PAI Input References

See [CS_PAI Sheet on page 71](#) for details.

- Substitute XT730 for XT101

PAI Outputs to PPID Inputs

Parameter	Description
Val	Value for PPID Inp_PV parameter Process Variable (PVEU)
SrcQ	Value for PPID Inp_PVSrcQ parameter Inp_PV source status and quality: 0 = Good, live, confirmed good 1 = Good, live, assumed good 2 = Good, no feedback, assumed good 8 = Test, simulated 9 = Test, loopback 10 = Test, manually entered 16 = Uncertain, live, off-spec 17 = Uncertain, substituted at device 18 = Uncertain, substituted at instruction 19 = Uncertain, using last known good 20 = Uncertain, using replacement value 32 = Bad, signal failure 33 = Bad, channel fault 34 = Bad, module/communications fault 35 = Bad, invalid configuration
Sts_eNotifyAll	Value for PPID Inp_PVNotify parameter Related PV object alarm priority and acknowledgment status:
	0 = Not in alarm, acknowledged
	1 = Not in alarm, unacknowledged or reset required
	2 = Low severity alarm, acknowledged
	3 = Low severity alarm, unacknowledged
	4 = Medium severity alarm, acknowledged
	5 = Medium severity alarm, unacknowledged
	6 = High severity alarm, acknowledged
	7 = High severity alarm, unacknowledged
	8 = Urgent severity alarm, acknowledged
	9 = Urgent severity alarm, unacknowledged

PPID Input References

Parameter	Description
XC730.Val_CVout	Loop CV after clamping and ramping (CVEU). Source: outer loop
XC730.Sts_WindupHi	Windup high signal. When true, the CV cannot integrate in a positive direction. The signal is typically obtained from the Windup hi output from an inner loop. Default is false.
XC730.Sts_WindupLo	Windup low signal. When true, the CV cannot integrate in a negative direction. The signal is typically obtained from the Windup low output from an inner loop. Default is false.
XC730.Sts_Available	1 = Inner loop (slave object) is available. 0 = Inner loop is not available, PPID tracks Inp_CVTrack, typically inner loop SP or actuator position. Default is true.
XIC730_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
XIC730_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
XIC730_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
XIC730_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
CVIOFault	Input connection from IO Faults sheet
XC730.Sts_eNotify	Alarm status from PAO: 0 = Not in alarm, acknowledged, 1 = Not in alarm, unacknowledged or reset required, 2 = Low severity alarm, acknowledged, 3 = Low severity alarm, unacknowledged, 4 = Medium severity alarm, acknowledged, 5 = Medium severity alarm, unacknowledged, 6 = High severity alarm, acknowledged, 7 = High severity alarm, unacknowledged, 8 = Urgent severity alarm, acknowledged, 9 = Urgent severity alarm, unacknowledged.

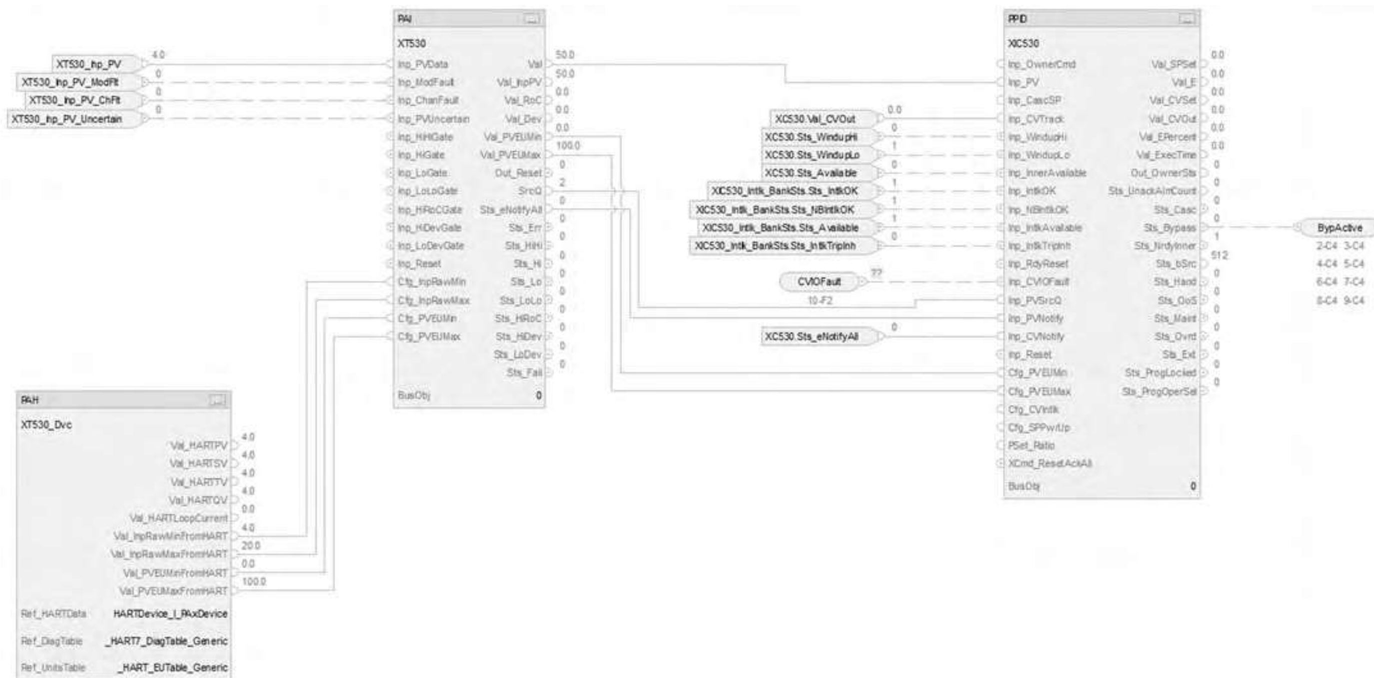
PPID Output References

Parameter	Description
BypActive	Output connection to interlock bank sheet

PPID Configuration Considerations

Operand	Type	Description
PlantPAx control	P_PID	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

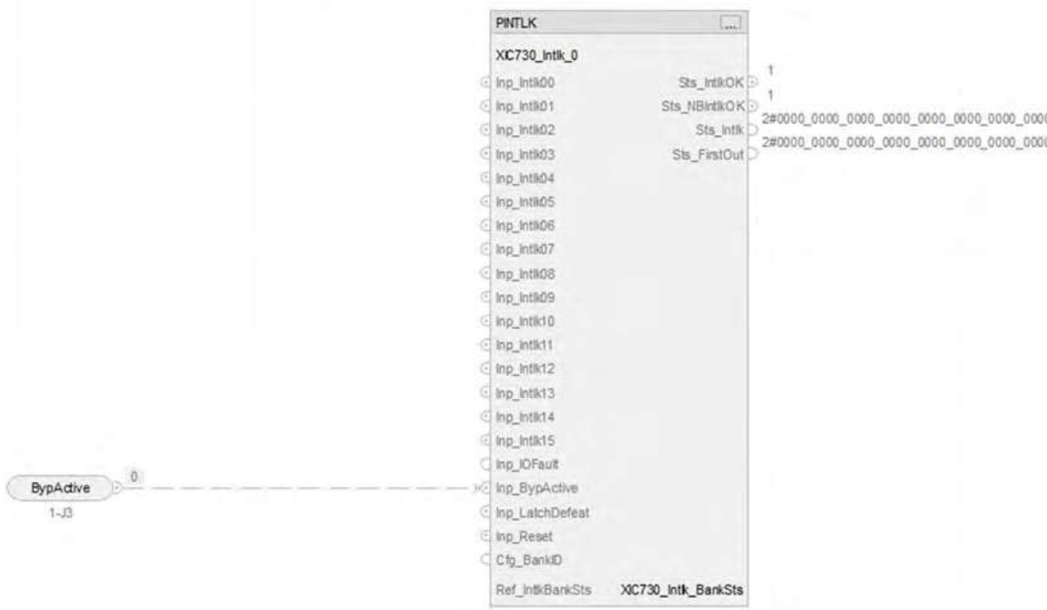
CS_PPID HART Sheet



The CS_PPID HART sheet operates the same as the CS_PPID sheet but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute for XT530 for XT100
- For more information, see HART Integration on page 31.

Interlock Bank Sheet



PINTLK Input Reference

Parameter	Description
BypActive	Input connection from CS_PPID sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

IO Faults Sheet

Faults Input References

Parameter	Description
XC730.Sts_IOFault XC530.Sts_IOFault	1 = IO Fault Status Bad 0 = OK There is a predefined default discrete Logix tag-based alarm for the status. Set standard configuration members of the discrete Logix tag-based alarm. Access alarm elements using this format: PAOTag.@Alarms.Alm_IOFault.AlarmElement Source: PAO instruction
XC730.Sts_DeviceFault XC530.Sts_DeviceFault	Device Fault status: 1 = Bad 0 = OK. There is a predefined default discrete Logix tag-based alarm for the status. Set standard configuration members of the discrete Logix tag-based alarm. Access alarm elements using this format: PAOTag.@Alarms.Alm_DeviceFault.AlarmElement Source: PAO instruction

Fault Output Reference

Parameter	Description
CVIOFault	Output connection to CS_PPID sheet

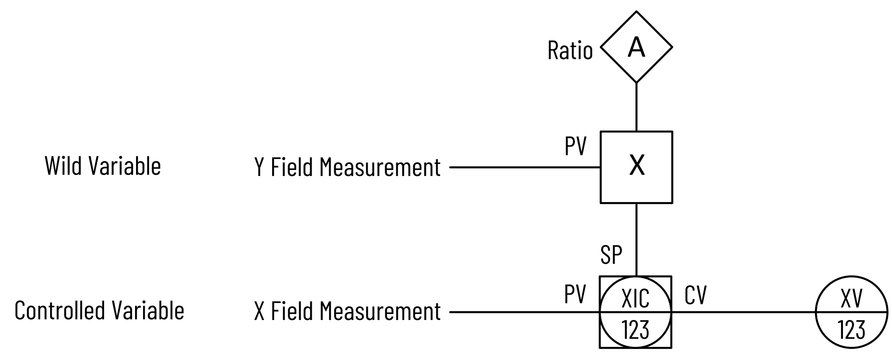
For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

Process Proportional + Integral + Derivative (PPID) Ratio Control Strategies

Use the PPID Ratio control strategy to add a material in a set proportion to another material.

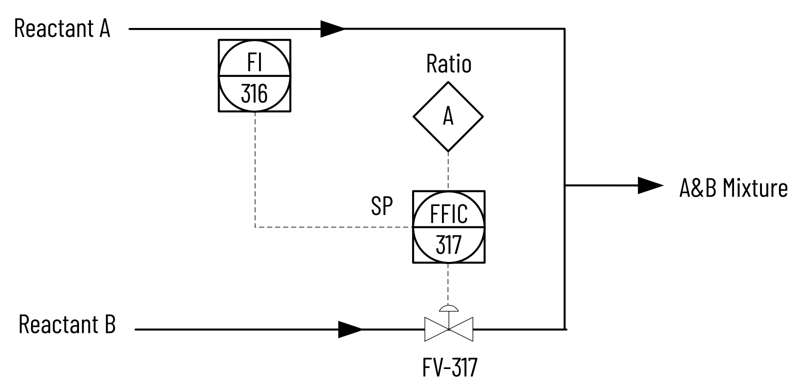
To scale the CV to align with the associated I/O module channel range or to accommodate a fail-open (FO) valve (or air to close) use either of the following options:

- Use a basic PPID with Analog Output control strategy
- Insert a scalar instruction between the PPID CV and the analog output channel reference



PPID Ratio Example

In this example, two reactants (A and B) are added to a tank in a constant ratio. The flow rate of reactant A might change over time because of some upstream process upsets. Use a PPID Ratio control strategy to automatically adjust the rate of the reactant B addition. In this example, reactant A is the uncontrolled or wild flow because it is not controlled by the PPID instruction. The flow of reactant B is the controlled flow.



To perform ratio control with a PPID instruction, set the Cfg_HasCasc and Cfg_HasRatio input parameters. Wire the uncontrolled flow into the Inp_CascSP input parameter. When in Cascade/Ratio mode, the uncontrolled flow is multiplied by either the OSet_Ratio, when in Operator control, or the PSet_Ratio, when in Program control, and the resulting value is used by the PPID instruction as the setpoint.

The following PPID control strategies are available as routines in the process library:

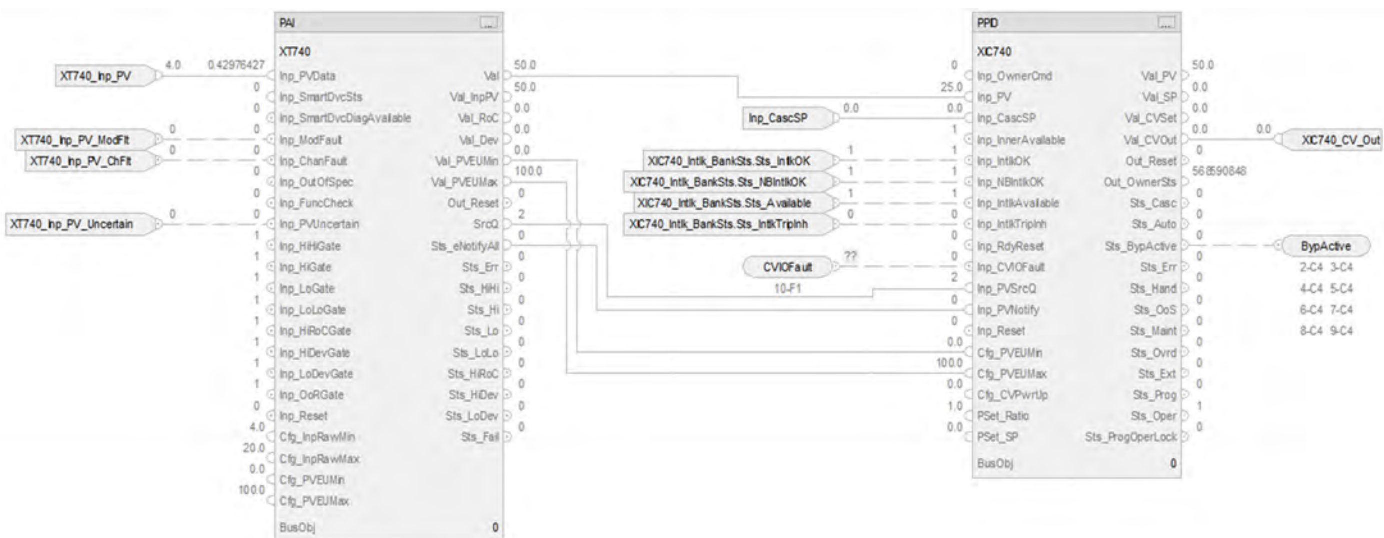
- CS_PPID_RATIO
- CS_PPID_RATIO_HART

Import the appropriate control strategy as a **routine** in your controller project.

The PPID Ratio control strategy contains these sheets:

Sheet	Description
CS_PPID_RATIO	PPID instruction
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The PPID instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder
IO Faults	The logic monitors Control Variable faults.

CS_PPID_RATIO Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.

- Substitute XIC740 for the PV data instance of XT101
- Substitute XT740 for the remaining instances of XT101

PAI Outputs to PPID Inputs

Parameter	Description
Val	Value for PPID Inp_PV parameter Process Variable (PVEU)
Val_PVEUmin	Value for PPID Cfg_PVEUMin parameter PV minimum value in engineering units (PVEU). Valid any float less than Cfg_PVEUMax.
Val_PVEUMax	Value for PPID Cfg_PVEUMax parameter PV maximum value in engineering units (PVEU). Valid any float greater than Cfg_PVEUMin.
SrcQ	Value for PPID Inp_PVSrcQ parameter Inp_PV source status and quality: 0 = Good, live, confirmed good 1 = Good, live, assumed good 2 = Good, no feedback, assumed good 8 = Test, simulated 9 = Test, loopback 10 = Test, manually entered 16 = Uncertain, live, off-spec 17 = Uncertain, substituted at device 18 = Uncertain, substituted at instruction 19 = Uncertain, using last known good 20 = Uncertain, using replacement value 32 = Bad, signal failure 33 = Bad, channel fault 34 = Bad, module/communications fault 35 = Bad, invalid configuration
Sts_eNotifyAll	Value for PPID Inp_PVNotify parameter Related PV object alarm priority and acknowledgment status: 0 = Not in alarm, acknowledged 1 = Not in alarm, unacknowledged or reset required 2 = Low severity alarm, acknowledged 3 = Low severity alarm, unacknowledged 4 = Medium severity alarm, acknowledged 5 = Medium severity alarm, unacknowledged 6 = High severity alarm, acknowledged 7 = High severity alarm, unacknowledged 8 = Urgent severity alarm, acknowledged 9 = Urgent severity alarm, unacknowledged

PPID Input References

Parameter	Description
XIC740_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
XIC740_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
XIC740_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
XIC740_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
CVIOFault	Input connection from IO Faults sheet

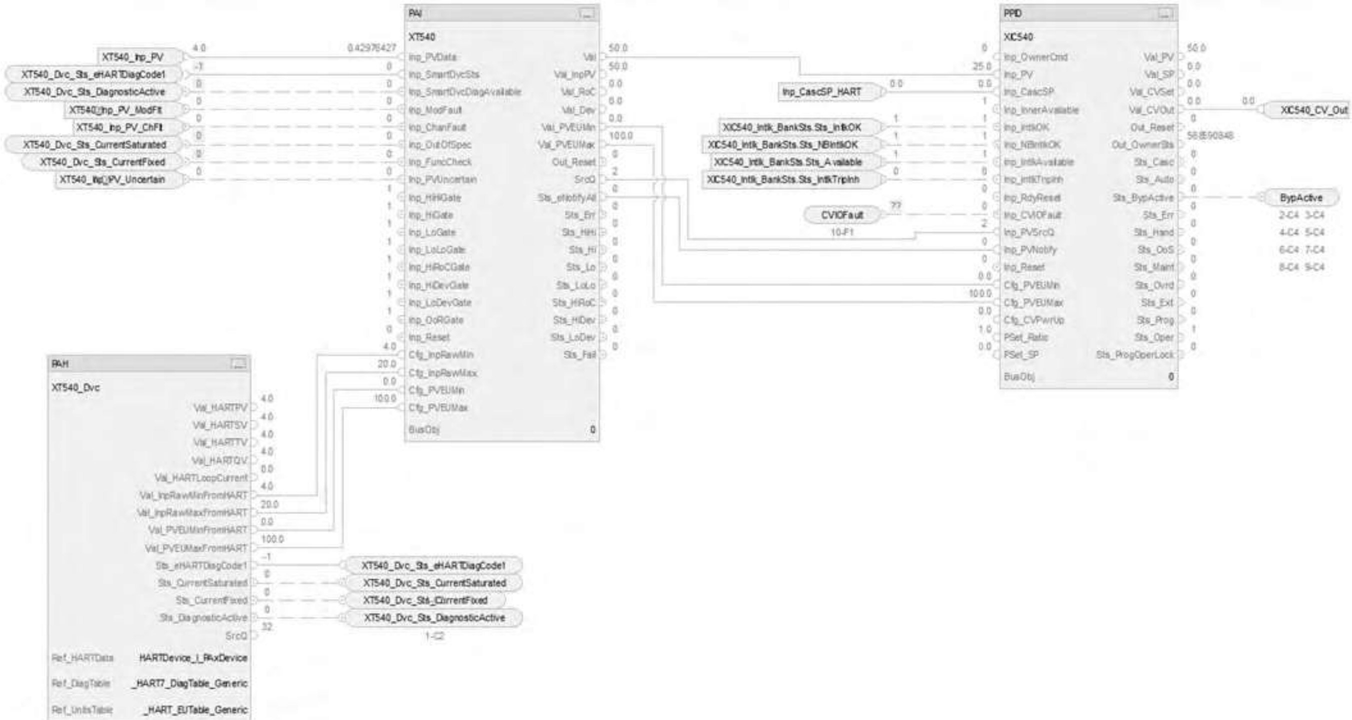
PPID Output References

Parameter	Description
XIC740_Out_CV	Control Variable output Loop CV after clamping and ramping (CVEU)
BypActive	Output connection to interlock bank sheet

PPID Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_PID	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

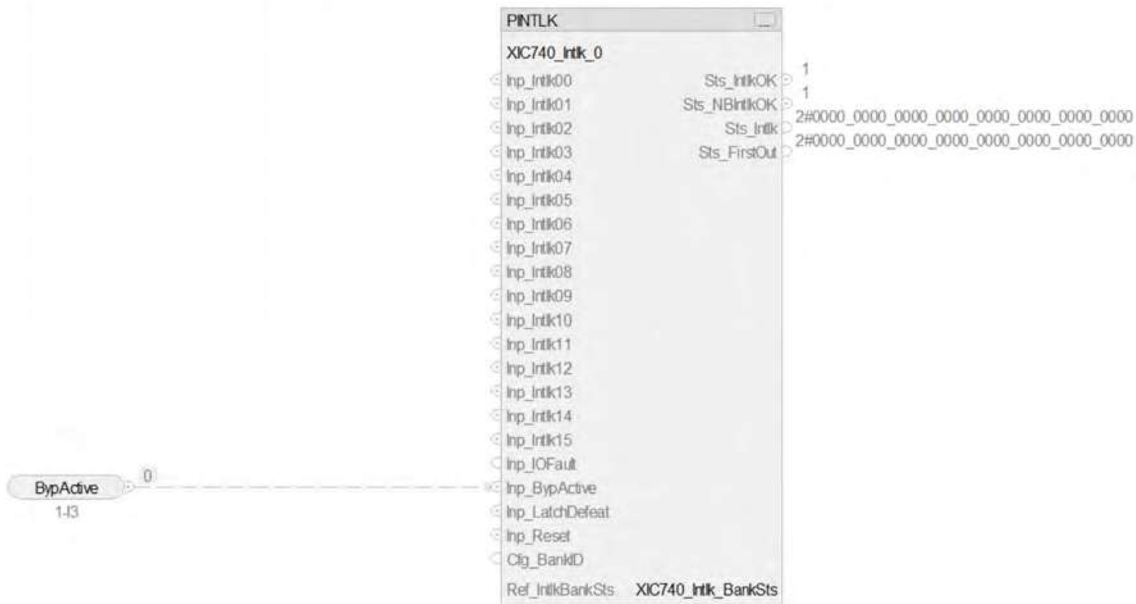
CS_PPID_RATIO HART Sheet



The CS_PPID_RATIO HART sheet operates the same as the CS_PPID_RATIO sheet but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute XIC540 for the PV data instance of XT101
- Substitute XT540 for the remaining instances of XT100
- For more information, see HART Integration on page 31.

Interlock Bank Sheet



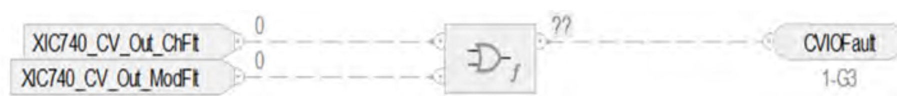
PINTLK Input Reference

Parameter	Description
BypActive	Input connection from CS_PPID_RATIO sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

IO Faults Sheet



Fault Input References

Parameter	Description
XIC740_Out_CV_ChFlt	Channel fault 1 = I/O channel fault or failure 0 = OK Source: PAI instruction
XIC740_Out_CV_ModFlt	Module fault 1 = I/O module failure or module communication status bad 0 = OK Source: PAI instruction

Fault Output Reference

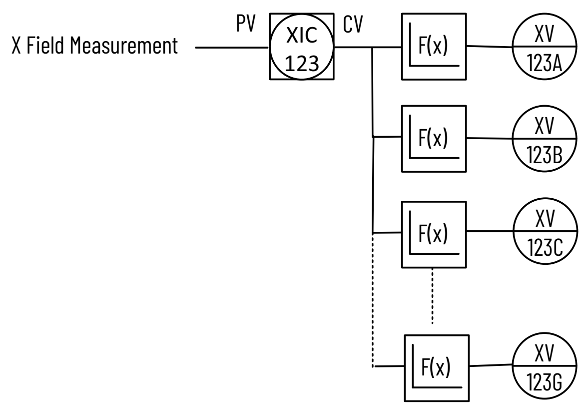
Parameter	Description
CVIOFault	Output connection to CS_PPID_RATIO sheet

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

Notes:

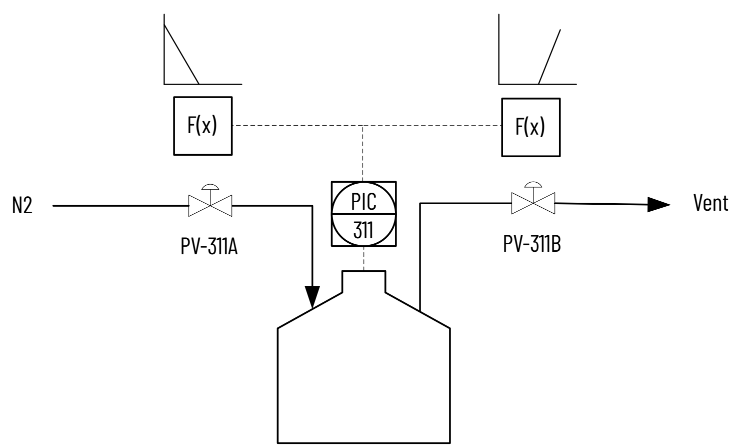
Process Proportional + Integral + Derivative (PPID) Split Range Control Strategies

This PPID Split Range control strategy as provided, manipulates two field devices to maintain one process variable (PV) at setpoint (SP). You can modify this strategy to manipulate up to eight field devices in one Split Range PPID control strategy.



A feature built into the Split Range control strategies is that the PPID instruction receives an indication whether its downstream objects can be controlled. If no downstream object is available for manipulation, the PPID tracks a configured selection (CV1 Initial Value or a fixed value).

PPID Split Range Example

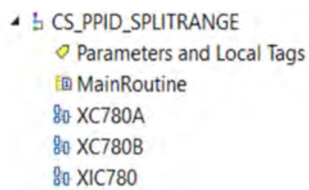


Blanket gas (an inert gas) often pressurizes tanks containing combustible material to ensure no admission of air. In the example above, when PIC-311 CV = 50%, both valves are closed. When the CV is less than 50%, the pressurizing valve (PV-311A) opens, and the vent valve (PV-311B) is kept closed. When the CV is greater than 50%, the vent valve (PV-311B) opens, and the pressuring valve is kept closed.

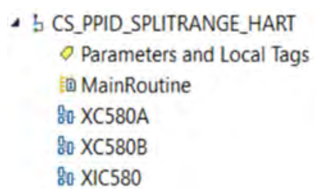
In practice, a gap can be used in the characterization to keep the valves from continuously cycling when the CV is near 50% (such as keep both valves closed when 48% < CV < 52%). The valves characterization (CV splitting) is done with the Process Analog Fanout (PFO) instruction. The action of the PPID is direct acting to accommodate the valves characterization (if pressure is above setpoint, the CV increases; if pressure is below setpoint, the CV decreases).

The following PPID Split Range control strategies (consisting of multiple routines) are available in the process library:

- CS_PPID_SPLITRANGE



- CS_PPID_SPLITRANGE_HART



Import the **routines** for the appropriate control strategy in your controller project. Each control strategy contains multiple routines; each routine contains multiple Function Block sheets. The control strategy, as supplied, uses only two analog outputs. The control strategy can support as many as eight analog outputs by exposing additional parameters in the PFO instruction and adding PAO routines.

Each PPID Split Range control strategy Program is built from multiple Routines:

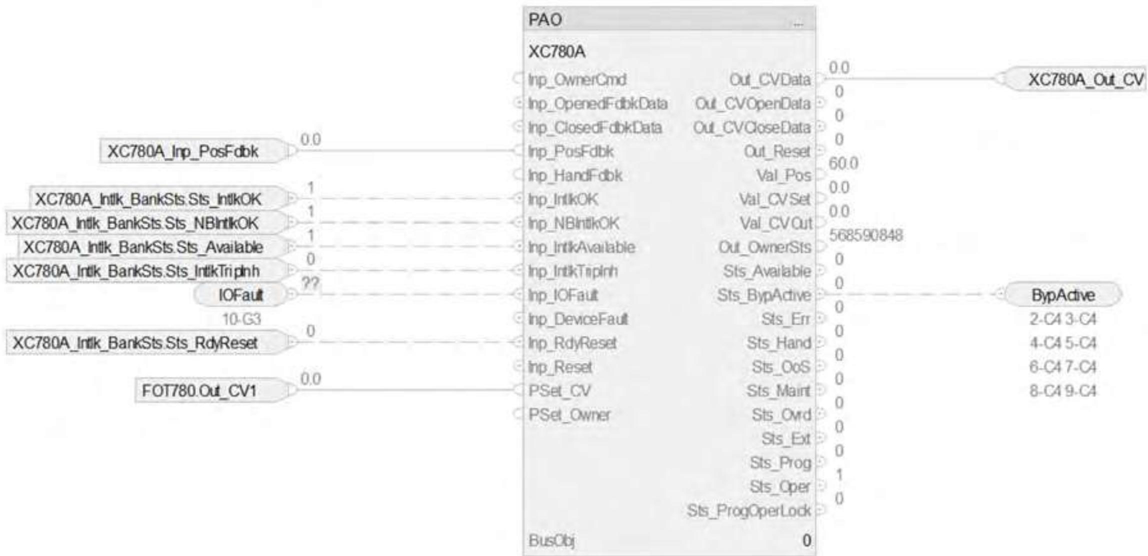
- Process Analog Output (XC780A analog/XC580A HART)
- Process Analog Output (XC780B analog/XC580B HART)
- Process Analog Input to Process PID with Fanout (XIC780 analog/XIC580 HART)

ROUTINE: Process Analog Output (XC780A and XC780B analog/XC580A and XC580B HART)

There are two routines; each routine contains these sheets:

Sheet	Description
CS_PAO	Process High or Low Selector instruction • XC780A and XC780B analog • HC580A and XC580B HART
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The PAO instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder.
IO Faults	The logic monitors one analog output channel for I/O fault input and raises alarm on an I/O fault.

CS_PAO Sheet



Input References to PAO

See the Process Analog Output (PAO) Control Strategies on page 83 for details.

Substitute:

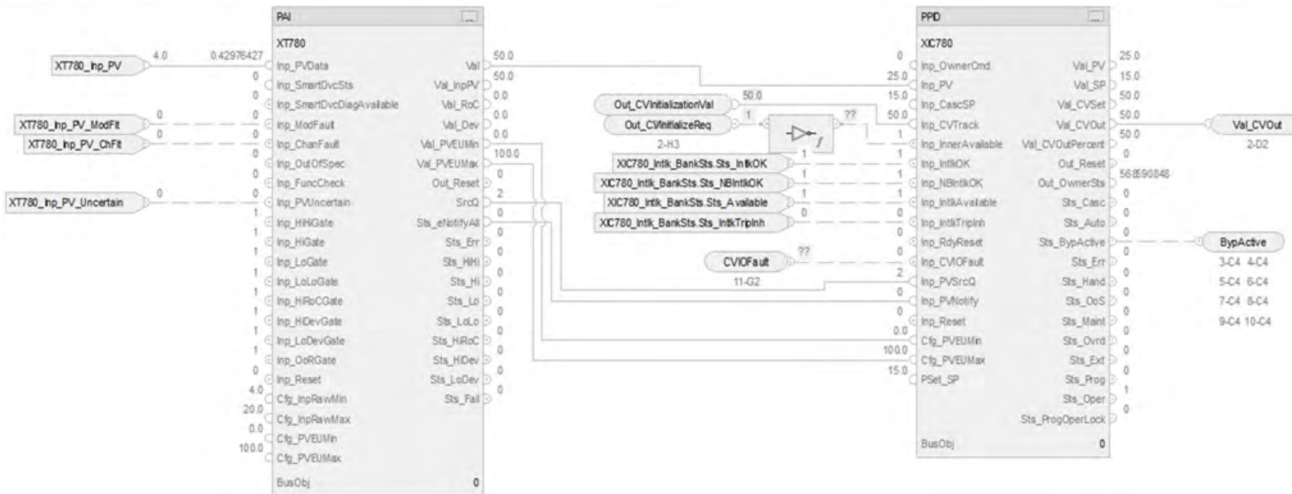
Analog Input	HART Input
XC780 for XC100 • XC780A for first instance • XC780B for second instance	XC580 for XC101 • XC580A for first instance • XC580B for second instance
FOT780 for XC100	FOT580 for XC100

ROUTINE: Process Analog Input to Process PID with Fanout

There are two routines (One without HART and another with HART); each routine contains these sheets.

Sheet	Description
CS_PPID	Process PID instruction with override XIC790/XIC800 analog XIC590/XIC600 HART
PFO	Process fan out
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The PPID instruction monitors interlock conditions which cause output CV and SP to shed. CV shed can be configured to hold the last good CV value or to use the configured safe value. SP is shed to current PV. There are 8 interlock bank sheets; each bank exposes 16 interlocks but supports as many as 32 interlocks. Use those sheets and interlocks that you need; delete the remainder
IO Faults	The logic monitors Control Variable faults.

CS_PPID Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.

- Substitute XIC780 for the PV data instance of XT101
- Substitute XT780 for the remaining instances of XT101

PAI Outputs to PPID Inputs

Parameter	Description
Val	Value for PPID Inp_PV parameter Process Variable (PVEU)
Val_PVEUmin	Value for PPID Cfg_PVEUMin parameter PV minimum value in engineering units (PVEU). Valid any float less than Cfg_PVEUMax.
Val_PVEUMax	Value for PPID Cfg_PVEUMax parameter PV maximum value in engineering units . (PVEU). Valid any float greater than Cfg_PVEUMin.
SrcQ	Value for PPID Inp_PVSrcQ parameter Inp_PV source status and quality: 0 = Good, live, confirmed good 1 = Good, live, assumed good 2 = Good, no feedback, assumed good 8 = Test, simulated 9 = Test, loopback 10 = Test, manually entered 16 = Uncertain, live, off-spec 17 = Uncertain, substituted at device 18 = Uncertain, substituted at instruction 19 = Uncertain, using last known good 20 = Uncertain, using replacement value 32 = Bad, signal failure 33 = Bad, channel fault 34 = Bad, module/communications fault 35 = Bad, invalid configuration
Sts_eNotifyAll	Value for PPID Inp_PVNotify parameter Related PV object alarm priority and acknowledgment status: 0 = Not in alarm, acknowledged 1 = Not in alarm, unacknowledged or reset required 2 = Low severity alarm, acknowledged 3 = Low severity alarm, unacknowledged 4 = Medium severity alarm, acknowledged 5 = Medium severity alarm, unacknowledged 6 = High severity alarm, acknowledged 7 = High severity alarm, unacknowledged 8 = Urgent severity alarm, acknowledged 9 = Urgent severity alarm, unacknowledged

Input References to PPID

Parameter	Description
Out_CVInitializationVal	Initialization value to PPID Source: PFO instruction
Out_CVInitializeReq	Initialization request to PPID Source: PFO instruction
XIC780_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
XIC780_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
XIC780_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
XIC780_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
CVIOFault	Input connection from IO Faults sheet
XIC780_PSet_SP	Program setting for SP, loop mode Auto (PVEU). Valid any float.

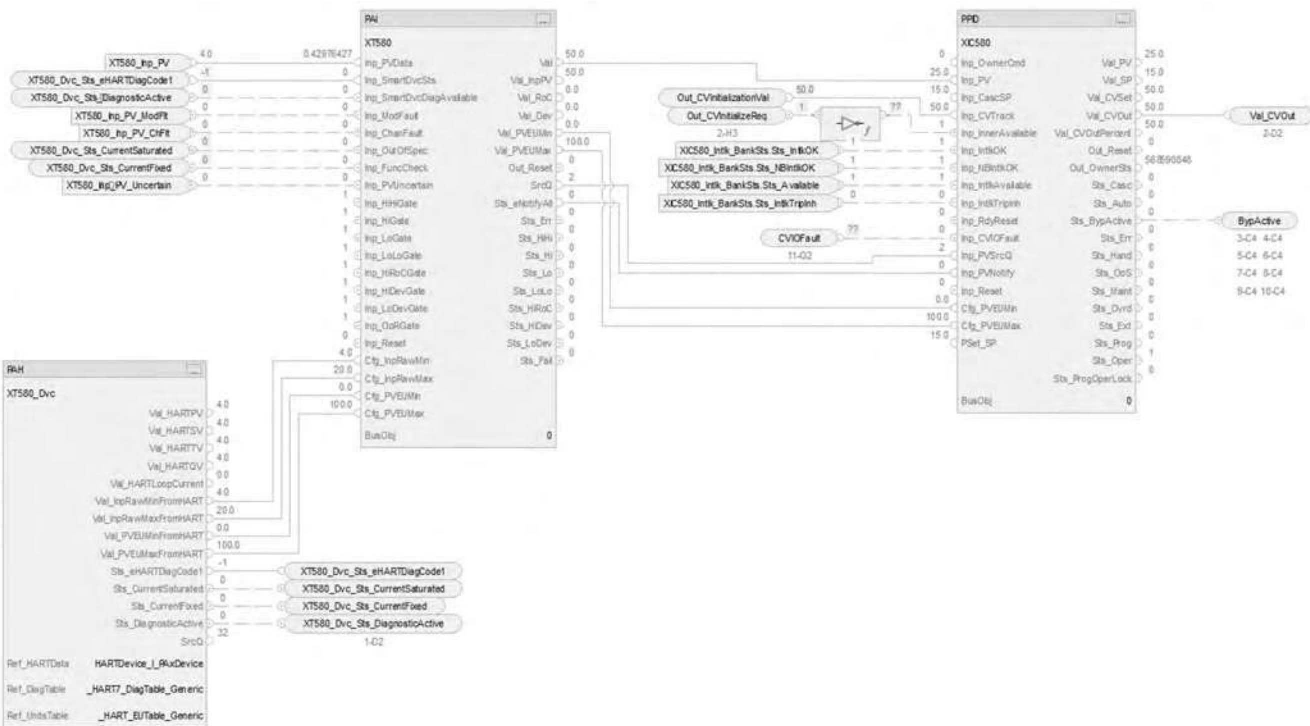
PPID Output References

Parameter	Description
Val_CVOut	Control Variable output Loop CV after clamping and ramping (CVEU) Destination: PFO instruction
BypActive	Output connection to interlock bank sheet

PPID Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_PID	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

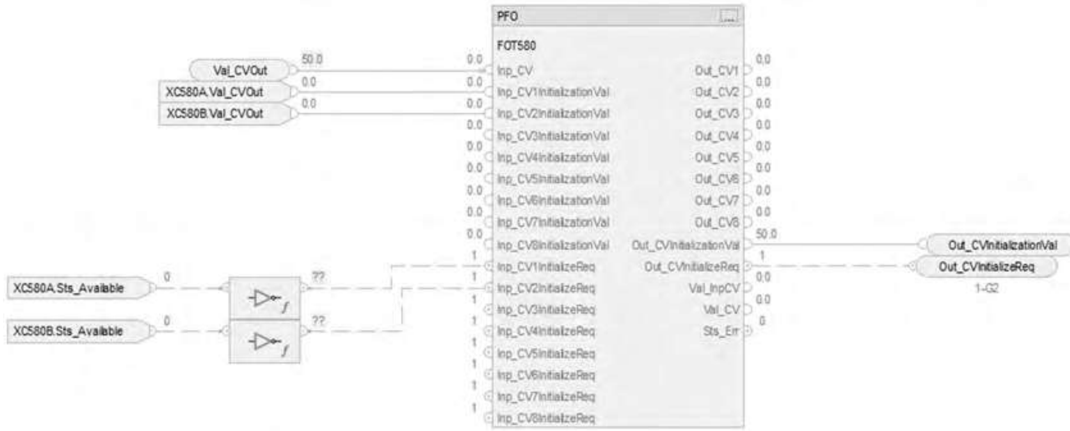
CS_PPID HART Sheet



The CS_PPID HART sheet operates the same as the CS_PPID sheet but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute XIC580 for the PV data instance of XT100
- Substitute XT580 for the remaining instances of XT100
- For more information, see HART Integration on page 31.

PFO Sheet



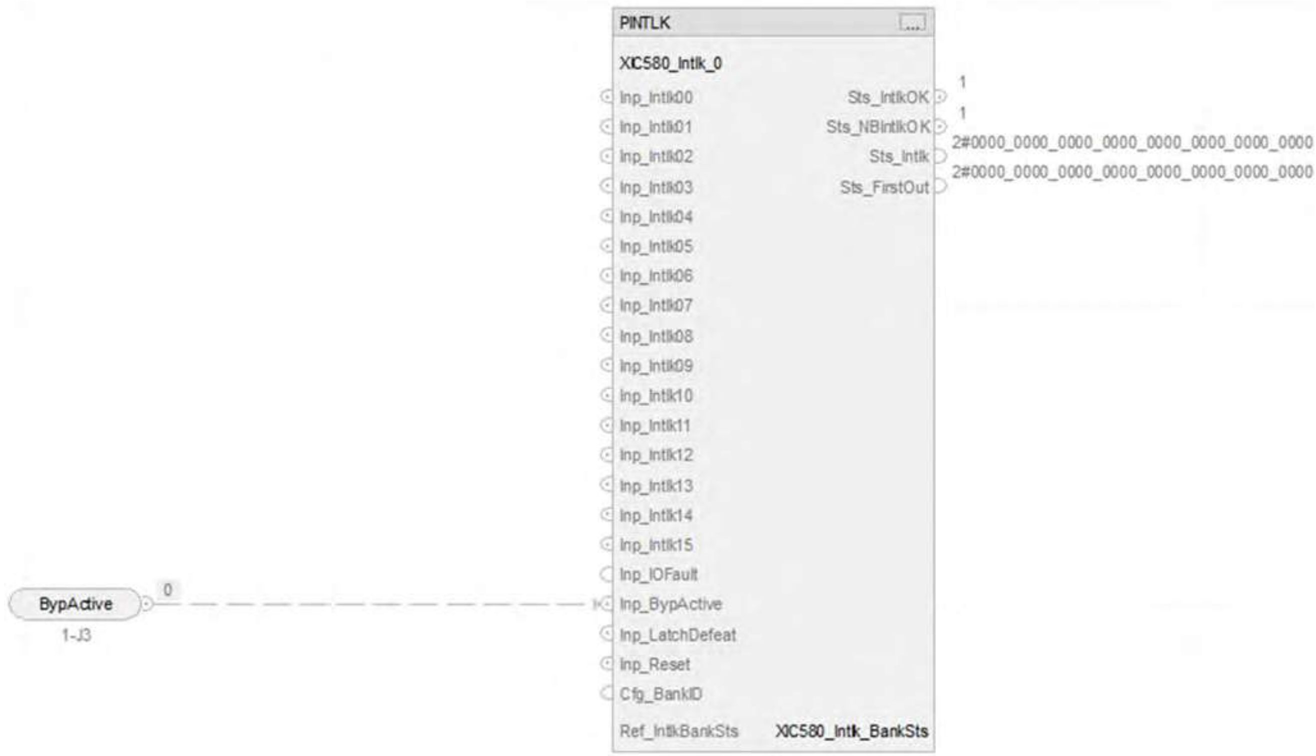
PFO Input References

Parameter	Description
Val_CVOut	Control Variable output Loop CV after clamping and ramping (CVEU)
XC780A.Val_CVOut XC780B.Val_CVOut XC580A.Val_CVOut XC580B.Val_CVOut	Value of CV Output after optional rate limiting, in engineering units. Extended Properties of this member: Engineering Unit - Engineering units (text) used for the analog output. Source: PAO instructions
XC780A.Sts_Available XC780B.Sts_Available XC580A.Sts_Available XC580B.Sts_Available	1 = Analog output available for control by program Source: PAO instructions

PFO Output References

Parameter	Description
Out_CVInitializationVal	Initialization value for PPID
Out_CVInitializeReq	Initialization request for PPID

Interlock Bank Sheet



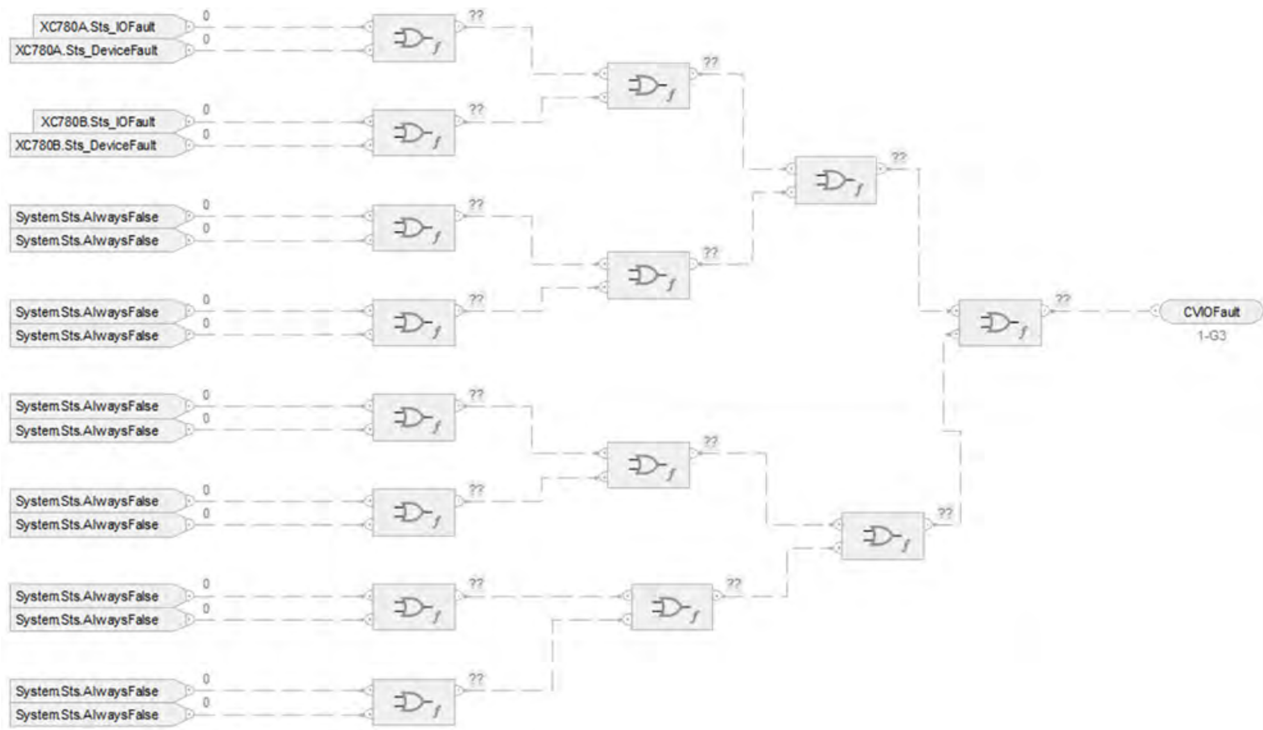
PINTLK Input Reference

Parameter	Description
BypActive	Input connection from CS_PPID_SPLITRANGE sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

IO Faults Sheet



Faults Input References

Parameter	Description
XC780A.Sts_IOFault XC780B.Sts_IOFault XC580A.Sts_IOFault XC580B.Sts_IOFault	1 = IO Fault Status Bad 0 = OK There is a predefined default discrete Logix tag-based alarm for the status. Set standard configuration members of the discrete Logix tag-based alarm. Access alarm elements using this format: PAOTag.@Alarms.Alm_IOFault.AlarmElement
XC780B.Sts_DeviceFault XC780B.Sts_DeviceFault XC580B.Sts_DeviceFault XC580B.Sts_DeviceFault	1 = Device Fault Status Bad 0=OK There is a predefined default discrete Logix tag-based alarm for the status. Set standard configuration members of the discrete Logix tag-based alarm. Access alarm elements using this format: PAOTag.@Alarms.Alm_DeviceFault.AlarmElement
System.Sts.AlwaysFalse	raP_UDT_Opr_ System.Sts.AlwaysFalse

Fault Output Reference

Parameter	Description
CVIOFault	Output connection to CS_PPID_SPLITRANGE sheet

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

Notes:

Chapter 30

Process Pressure/Temperature Compensated Flow (PPTC) Control Strategies

The Pressure/Temperature Compensated Flow (PPTC) instruction calculates the flow of a gas at standard / design temperature and pressure, essentially a mass flow rate, given a volumetric flow rate or differential pressure measurement. The design temperature and pressure are specific to each instrument, and since the thermodynamic conditions of the actual gas flow rarely align with the design conditions, temperature and pressure compensation is often used to compensate the actual flow measurement so that the measurement is adjusted to design conditions (essentially normalizing the gas flow to design conditions). This instruction requires measurements of the actual temperature and pressure of the flowing gas.

The following PPTC control strategy folders are available as routines in the process library:

- CS_PPTC
- CS_PPTC_HART

Within each control strategy folder there are two routines available to use:

- (RA-LIB)CS_PPTC_DP_5_00-03_Routine – Uncompensated flow provided using DP transmitter
- (RA-LIB)CS_ PPTC_DP 5_00-03_Routine - Uncompensated flow provided using linear flow transmitter

For the compensation to work correctly, like units for temperature and pressure must align and the final calculation is applied to absolute values. That is, Inp_Pact and Cfg_PStd must both be in the same units (Cfg_POffset is added to both values to convert to absolute pressure). Also, Inp_Tact and Cfg_TStd must both be in the same units (Cfg_TOffset is added to both values to convert to absolute temperature) _

The PPTC Add-On Instruction is intended as a calculation function only, between other blocks, and no HMI components are provided. If a faceplate or alarms are needed, the calculated output from the instruction can be sent to a PAI (analog input) instruction for alarming and display.

The following PPTC control strategies are available as routines in the process library:

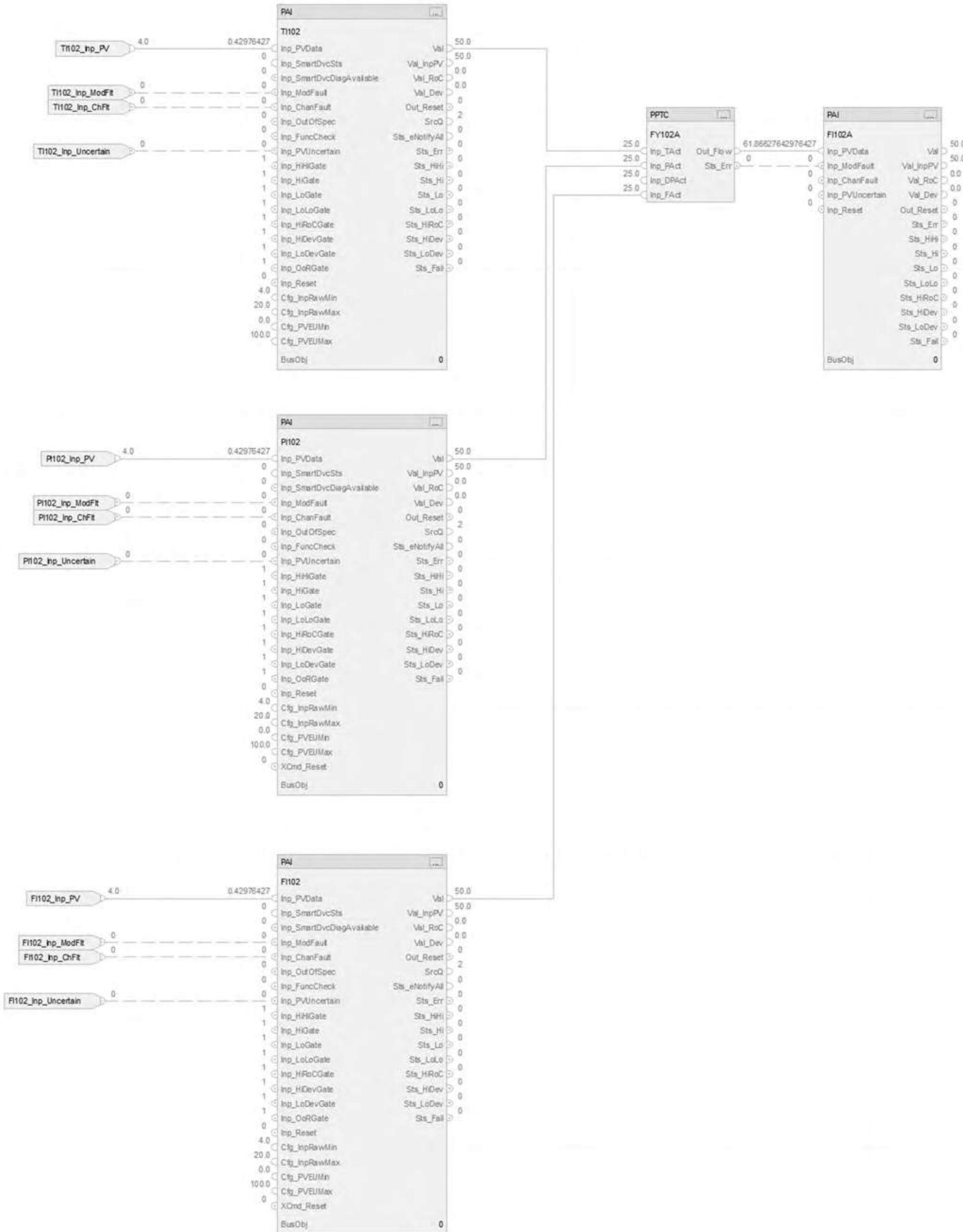
- CS_PPTC
- CS_PPTC_HART

Import the appropriate control strategy as a **routine** in your controller project.

Each PPTC control strategy contains one Function Block sheet:

Sheet	Description
CS_PPTC	Process Pressure/Temperature Compensated Flow instruction
CS_PPTC_HART	Process Pressure/Temperature Compensated Flow instruction with HART input

CS_PPTC Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.

PAI Instruction	Substitute
PPTC Inp_TAct Actual (measured) temperature	Substitute TI102/TI202 with desired instrument name
PPTC Inp_PAct Actual (measured) pressure	Substitute PI102/PI202 with desired instrument name
PPTC Inp_DPAct Actual (measured) differential pressure (square root)	Substitute PDIT202 with desired instrument name
PPTC Inp_FAct Actual (measured) uncompensated flow (linear)	Substitute FI102 with desired instrument name

To configure the flow calculation method, see the Advanced properties page for the PPTC instruction. Select one of the following:

- Differential pressure (PPTC Inp_DPAct)
- Flow input (PPTC Inp_Fact)

PAI Outputs

Parameter	Description
Val for PPTC Inp_TAct	Actual (measured) temperature
Val for PPTC Inp_PAct	Actual (measured) pressure
Val for PPTC Inp_DPAct	Actual (measured) differential pressure
Val PPTC Inp_FAct	Actual (measured) uncompensated flow

PPTC Outputs

Parameter	Description
Out_Flow	Compensated flow (at standard temperature and pressure: mass flow)
Sts_Err	1 = Error in configuration: See detail bits (Sts_Errxxx) for reason

PPTC Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_PRESS_TEMP_COMPENSATED	Instance of data structure (backing tag) required for proper operation of instruction

CS_PPTC_HART Sheet

The CS_PPTC_HART control strategy operates the same as the CS_PPTC control strategy but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- For more information, see HART Integration on page 31.

PAI Instruction	Substitute
PPTC Inp_TAct Actual (measured) temperature	Substitute TI1012/TI201 with desired instrument name
PPTC Inp_PAct Actual (measured) pressure	Substitute PI101/PI201with desired instrument name
PPTC Inp_DPAct Actual (measured) differential pressure	Substitute PDIT201 with desired instrument name
PPTC Inp_FAct Actual (measured) uncompensated flow	Substitute FI101 with desired instrument name

Notes:

Process Tank Strapping Table (PTST) Control Strategies

Use a PTST control strategy to calculate the volume of product in an upright cylindrical tank, given the level of the product and the tank calibration table. The instruction can compensate for:

- Free water at the bottom of the tank, given a product/water interface level.
- Thermal expansion of the tank shell, given the coefficient of linear expansion of the shell material and product and ambient temperatures.
- A floating tank roof, given the product density is provided.

The PTST instruction is intended only as a calculation function, between other blocks, and so no HMI components are provided.

The following PTST control strategy folders are available as routines in the process library:

- CS_PTST
- CS_PTST_HART

Within each control strategy folder there are two routines available to use:

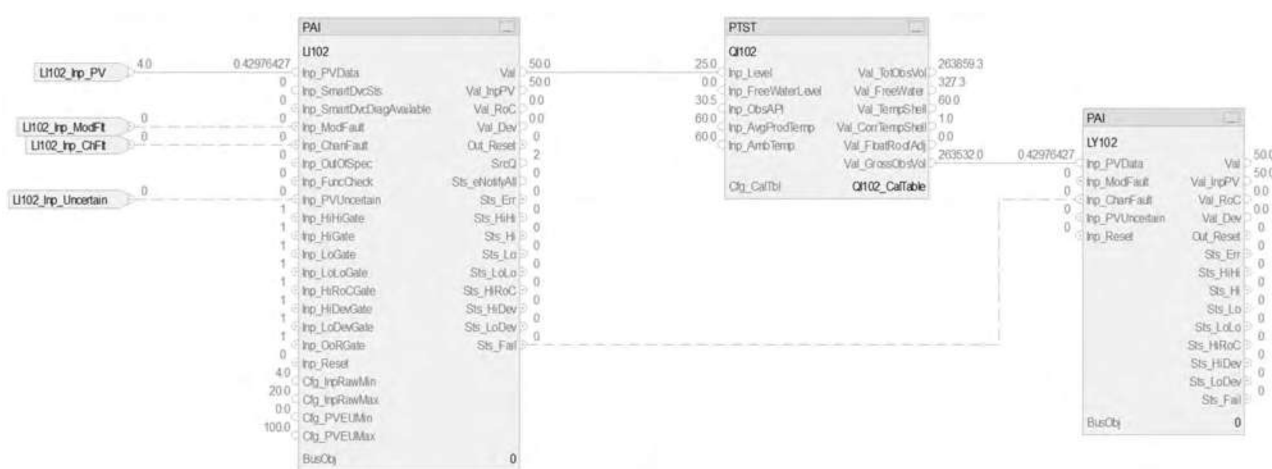
- (RA-LIB)CS_PTST_– Level input with no compensation
- (RA-LIB)CS_PTST_InpEnbl_ - Level input with compensation inputs exposed

Import the appropriate control strategy as a **routine** in your controller project.

Each PTST control strategy contains one Function Block sheet:

Sheet	Description
CS_PTST	Process Tank Strapping Table instruction
CS_PTST_HART	Process Tank Strapping Table instruction with HART input

CS_PTST Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.
- Substitute QI102 with the desired tag name
- Substitute LI102 with the desired instrument tag name
- Substitute LY102 with the desired tag name.

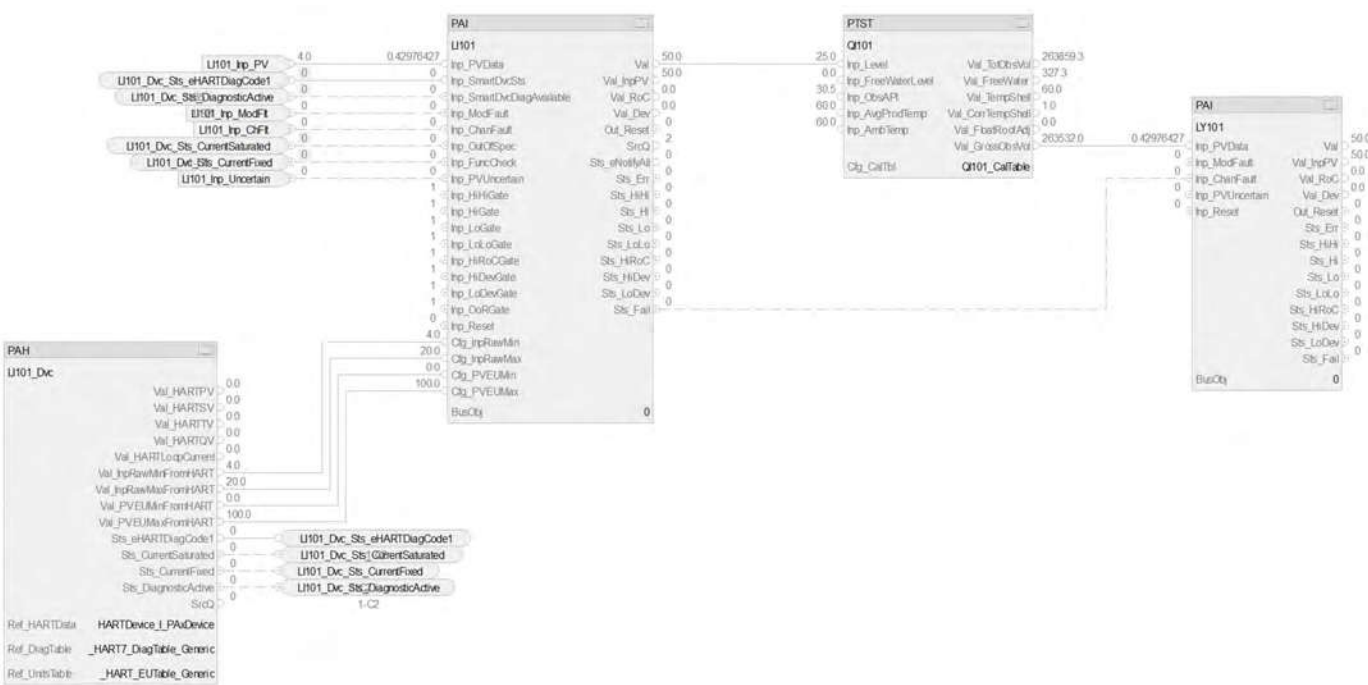
PAI Outputs

Parameter	Description
Val	Input to Inp_Level of PTST instruction Tank innage level, in feet or meters
Sts_Fail	Input to Inp_ChanFault of secondary PAI instruction 1 = I/O channel fault or failure 0 = OK

PTST Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_TANK_STRAPPING_TABLE	Instance of data structure (backing tag) required for proper operation of instruction
Cfg_CalTbl	P_STRAPPING_TABLE_ROW	Array for tank calibration table, level to volume

CS_PTST_HART Sheet



The CS_PTST_HART control strategy operates the same as the CS_PTST control strategy but relies on HART input data.
- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute QI102 with the desired tag name
- Substitute LI102 with the desired instrument tag name
- Substitute LY102 with the desired tag name.
- For more information, see HART Integration on page 31.

Process Valve Hand Operated (PVLVHO) Control Strategy

Use the PVLVHO control strategy to monitor a hand (locally) operated valve and display its current state. The valve can have any type of actuator – handwheel, lever, motor, solenoid, pneumatic, hydraulic – but it is normally operated at the valve and only monitored by the control system via open and closed limit switches.

This PVLV control strategy does not provide operator access to control the valve, but it does provide an optional Trip output. The Trip state is generated by interlock conditions not being met and the output can be used to de-energize a valve control circuit to drive the valve to its default (fail) position. If the trip function is used, the PVLV instruction checks to make sure that the valve reaches the configured trip position (open or closed) if a trip command is executed.

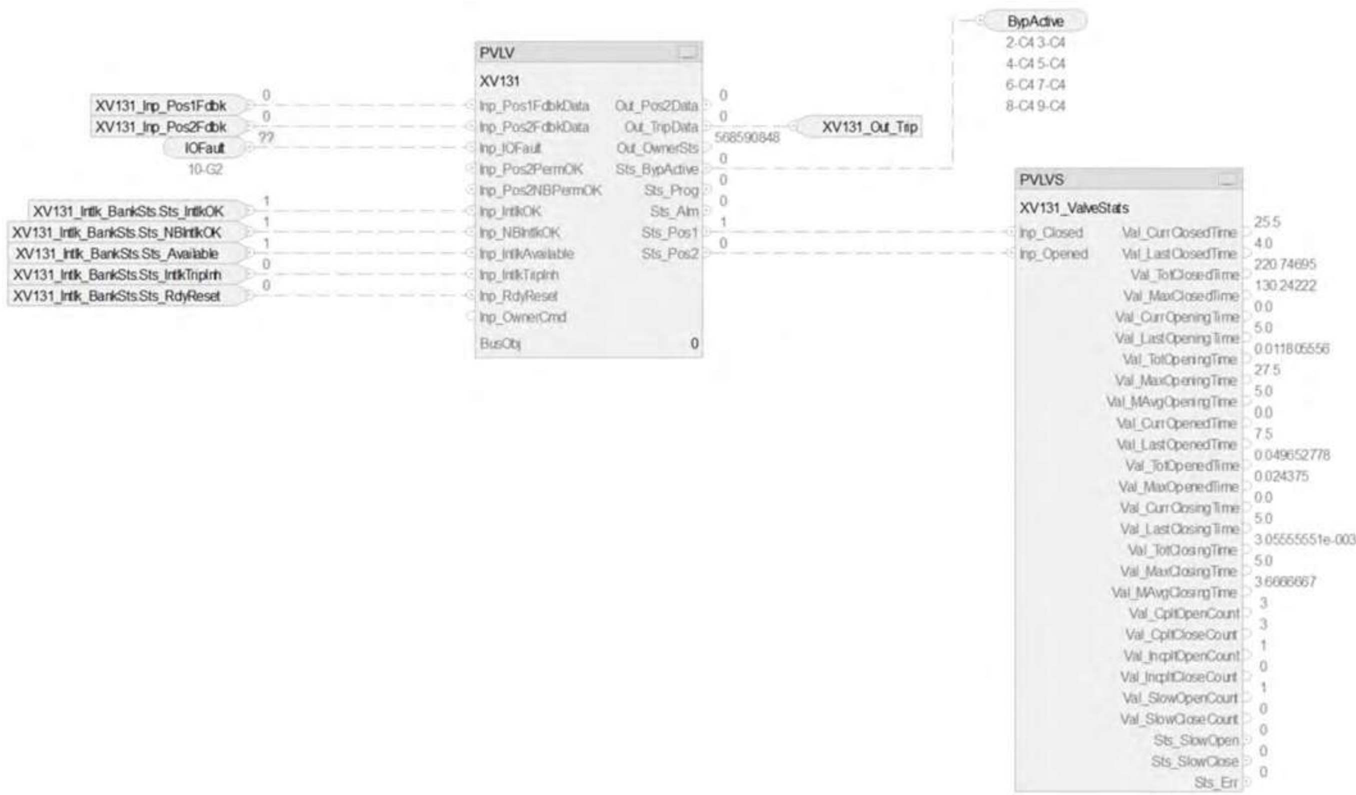
The CS_PVLVHO control strategy is available as a routine in the process library.

Import the control strategy as a **routine** in your controller project.

The PVLVHO control strategy contains these Function Block sheets:

Sheet	Description
CS_PVLVHO	Process Valve instruction, hand operated
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The PVLV instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder
IO Faults	The logic monitors one analog output channel for I/O fault input and raises an alarm on an I/O fault.

CS_PVLVHO Sheet



PVLV Input References

Parameter	Description
XV131_Inp_Pos1Fdbk	Feedback from Position 1 limit switch of the device 1 = Device confirmed Position 1
XV131_Inp_Pos2Fdbk	Feedback from Position 2 limit switch of the device 1 = Device confirmed Position 2
IOFault	Input connection from IO Faults sheet
XV131_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
XV131_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
XV131_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
XV131_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
XV131_Intlk_BankSts.Sts_RdyReset	Interlock bank status 1 = A latched interlock (returned to OK) is ready to be reset

PVLV Output References

Parameter	Description
XV131_Out_Trip	1 = Trip valve to safe/fail state
BypActive	Output connection to interlock bank sheet

PVLV Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_VALVE_DISCRETE	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

PVLV Output References to PVLVS

Parameter	Description
Sts_Pos1	1 = Valve requested to Position 1 and is confirmed Position 1
Sts_Pos2	1 = Valve requested to Position 1 and is confirmed Position 2

Interlock Bank Sheet

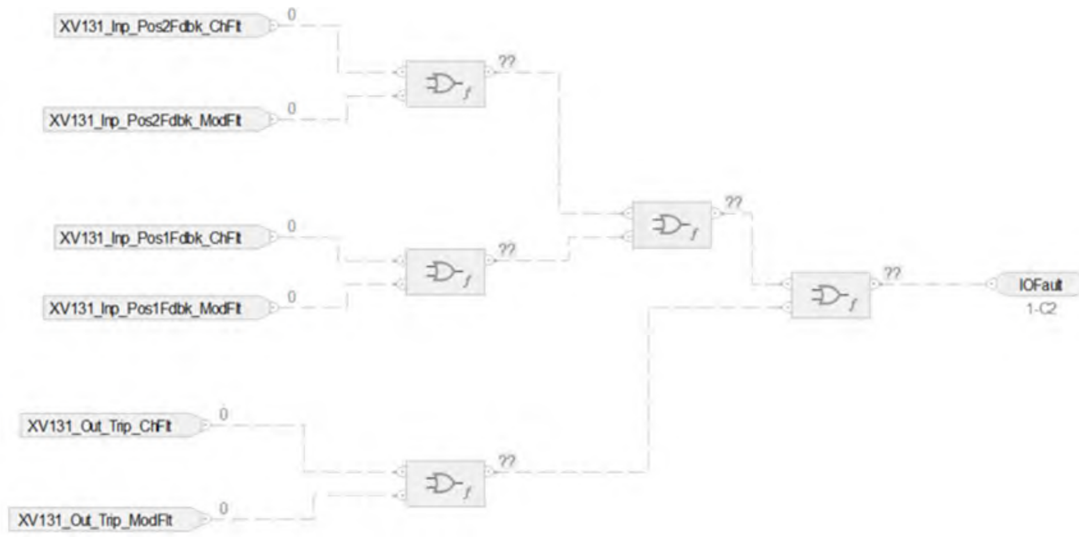
PINTLK Input Reference

Parameter	Description
BypActive	Input connection from CS_PVLVHO sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

IO Faults Sheet



Fault Input References

Parameter	Description
XV131_Inp_Pos1Fdbk_ChFlt	Tieback input 1 channel fault
XV131_Inp_Pos1Fdbk_ModFlt	Tieback input 1 module fault
XV131_Inp_Pos2Fdbk_ChFlt	Tieback input 2 channel fault
XV131_Inp_Pos2Fdbk_ModFlt	Tieback input 2 module fault
XV131_Out_Trip_ChFlt	Output channel fault
XV131_Out_Trip_ModFlt	Output module fault

Fault Output References

Parameter	Description
IOFault	Output connection to CS_PVLVHO sheet

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

Process Valve Motor Operated (PVLVMO) Control Strategy

Use the PVLVMO control strategy to operate (open and close) a motor-operated valve. Since a motor-operated valve has no spring return (to return the valve to the fail-safe state), two digital outputs are required(one to move the valve towards the open position; and another to move the valve to the closed position).

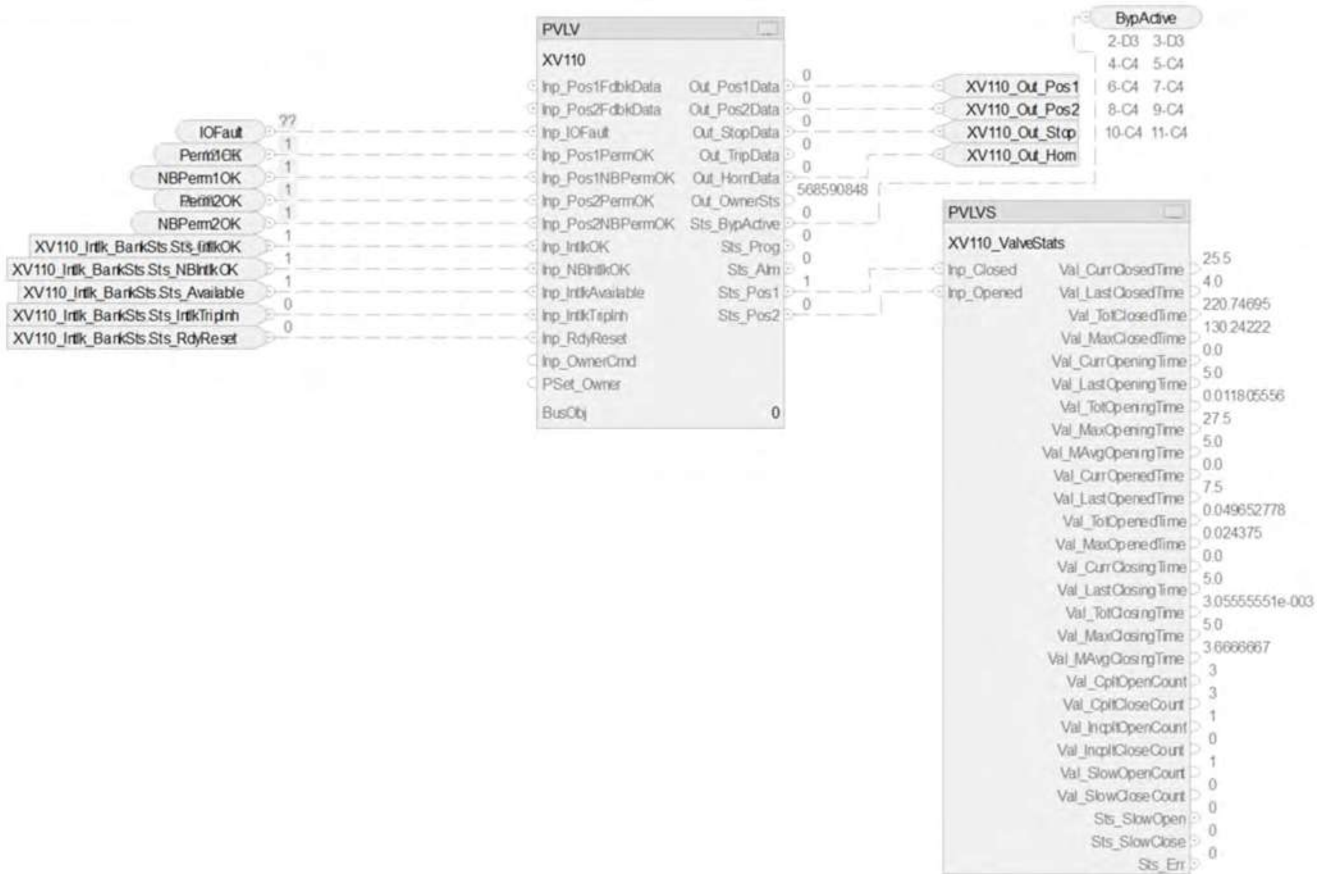
The CS_PVLVMO control strategy is available as a routine in the process library.

Import the control strategy as a **routine** in your controller project.

The PVLVMO control strategy contains these Function Block sheets:

Sheet	Description
CS_PVLVMO	Process Valve instruction, motor operated
Position 1 Permissives Position 2 Permissives	Process Permissives instruction The Process Permissives (PPERM) instruction collects, or sums up, the permissive conditions that let a piece of equipment energize. Position 1 and Position 2 permissives are applied to the commands to energize towards those positions. Permissive conditions must be true to energize equipment. Once the equipment is energized, permissives are ignored.
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The PVLV instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder
IO Faults	The logic monitors one analog output channel for I/O fault input and raises an alarm on an I/O fault.

CS_PVLVMO Sheet



PVLV Input References

Parameter	Description
IOFault	Input connection from IO Faults sheet
Perm1OK	Input connection from Position 1 Permissives sheet 1 = On permissives OK, device can turn On
NBPerm1OK	Input connection from Position 1 Permissives sheet 1 = Non-bypassable On permissives OK, device can turn On
Perm2OK	Input connection from Position 2 Permissives sheet 1 = On permissives OK, device can turn On
NBPerm2OK	Input connection from Position 2 Permissives sheet 1 = Non-bypassable On permissives OK, device can turn On
XV110_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
XV110_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
XV110_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
XV110_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
XV131_Intlk_BankSts.Sts_RdyReset	Interlock bank status 1 = A latched interlock (returned to OK) is ready to be reset

PVLV Output References

Parameter	Description
XV110_Out_Pos1	1 = Activate to move valve to Position 1
XV110_Out_Pos2	1 = Activate to move valve to Position 2
XV110_Out_Horn	1 = Sound audible before commanded valve start
BypActive	Output connection to permissives and interlock bank sheets

PVLV Configuration Considerations

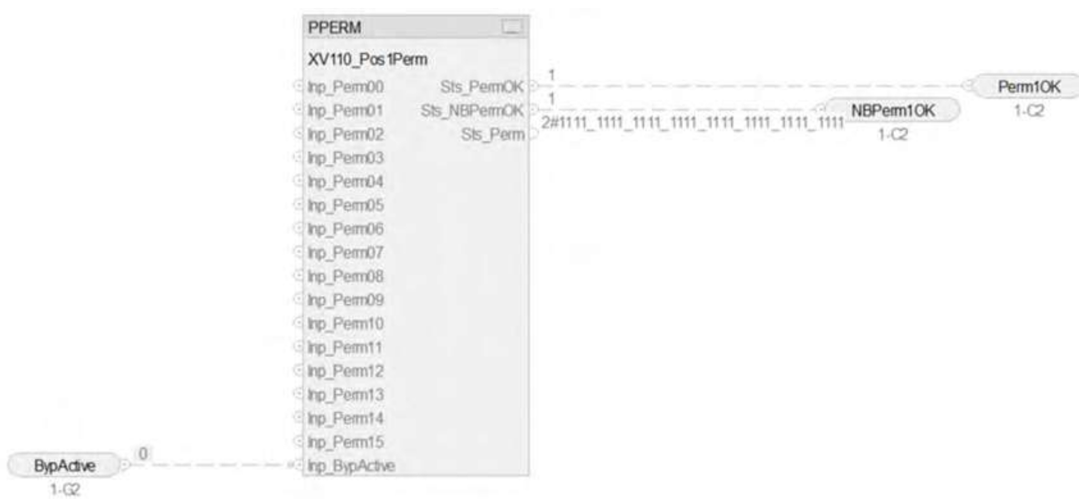
Operand	Type	Description
PlantPAx® control	P_VALVE_DISCRETE	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

PVLV Output References to PVLVS

Parameter	Description
Sts_Pos1	1 = Valve requested to Position 1 and is confirmed Position 1
Sts_Pos2	1 = Valve requested to Position 1 and is confirmed Position 2

Permissive Sheet

Image applies for Position 1, there is a similar sheet for Position 2.



PPERM Input References

Parameter	Description
BypActive	Input connection from the CS_PVLVMO Sheet.

PPERM Output References

Parameter	Description
Perm1OK Perm2OK	Overall permissive status (1 = OK to energize)
NBPerm1OK NBPerm2OK	Non-bypassable permissive status (1 = all non-bypassable permissives OK to energize)

Interlock Bank Sheet

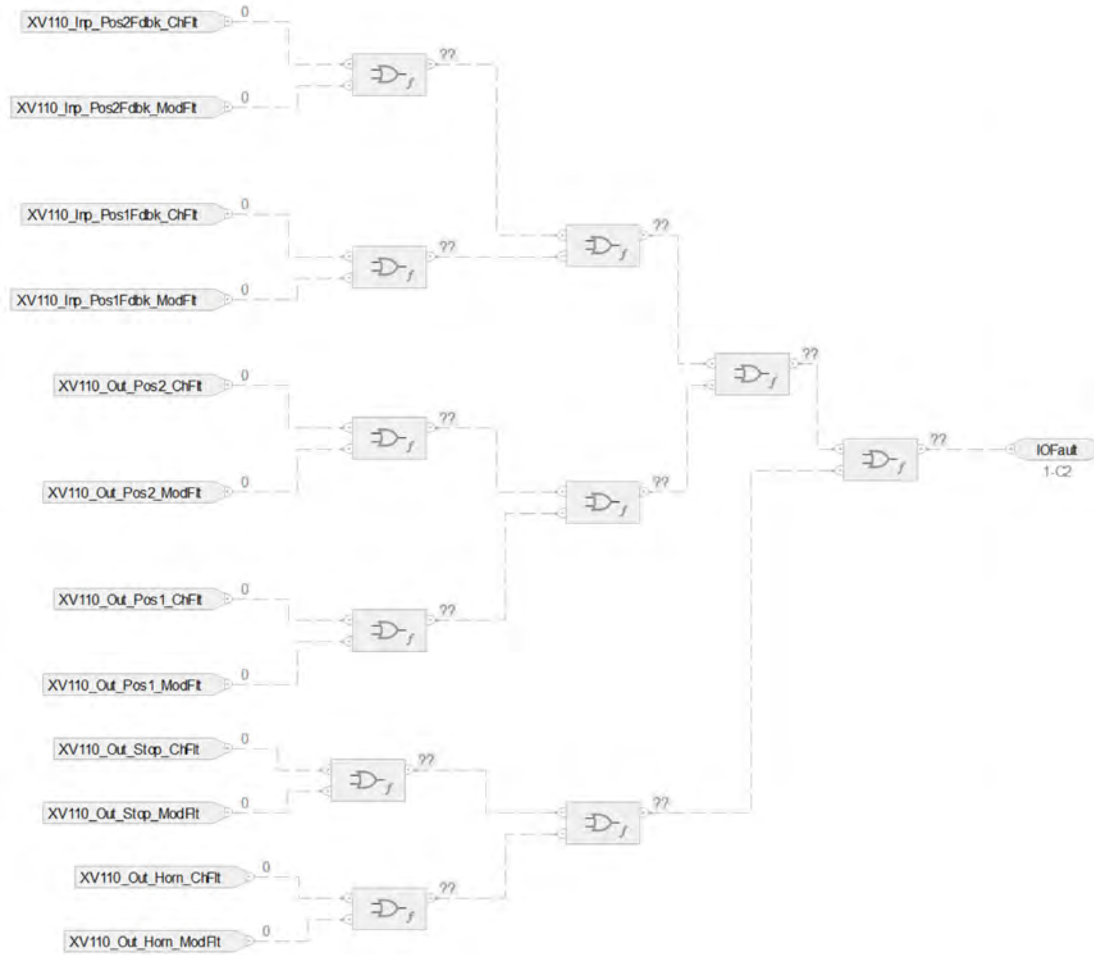
PINTLK Input Reference

Parameter	Description
BypActive	Input connection from CS_PVLVMO sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

IO Faults Sheet



Faults Input References

Parameter	Description
XV110_Inp_Pos1Fdbk_ChFlt	Tieback input 1 channel fault
XV110_Inp_Pos1Fdbk_ModFlt	Tieback input 1 module fault
XV110_Inp_Pos2Fdbk_ChFlt	Tieback input 2 channel fault
XV110_Inp_Pos2Fdbk_ModFlt	Tieback input 2 module fault
XV110_Out_Pos1_ChFlt	Position 1 channel fault
XV110_Out_Pos1_ModFlt	Position 1 module fault
XV110_Out_Pos2_ChFlt	Position 2 channel fault
XV110_Out_Pos2_ModFlt	Position 2 module fault
XV110_Out_Horn_ChFlt	Sound audible for output channel fault
XV110_Out_Horn_ModFlt	Sound audible for output module fault

Fault Output References

Parameter	Description
IOFault	Output connection to CS_PVLVMO sheet

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

Notes:

Process Mix Proof Valve (PVLVMP) Control Strategy

The Process Mix Proof Valve (PVLVMP) instruction controls and monitors feedback from a mix proof valve in various modes and states, and monitors for fault conditions. This instruction supports mix proof valves with or without additional connections for cleaning (CIP, clean-in-place) or steaming (SIP, sanitize in place).

Use the PVLVMP control strategy to control one mix proof valve in various modes and states, while monitoring position feedback inputs to verify that the valve reaches the commanded position. An alarm can be provided on failure to reach the commanded position.

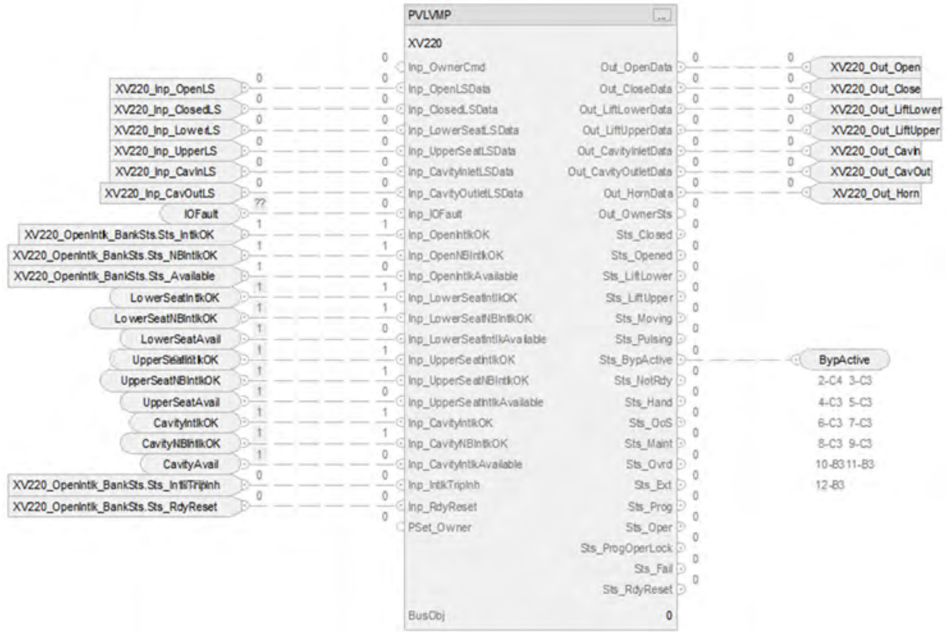
The CS_PVLVMP control strategy is available as a routine in the process library.

Import the control strategy as a **routine** in your controller project.

The PVLVMP control strategy contains these Function Block sheets:

Sheet	Description
CS_PVLVMP	Process Mix Proof Valve instruction
Permissives	Process Permissives instruction The Process Permissives (PPERM) instruction collects, or sums up, the permissive conditions that let a piece of equipment energize. In most cases, permissive conditions must be true to energize equipment. Once the equipment is energized, permissives are ignored.
Open Interlock Cavity Interlocks Upper Seat Interlocks Lower Seat Interlocks	The PVLVMP instruction monitors bypassable and non-bypassable Interlocks that force the Output instead of 'analog output' and to the configured safe state. • Open Interlock has 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. • Cavity Interlocks, Upper Seat Interlocks, and Lower Seat Interlocks each have one interlock sheet that exposes 16 of the available 32 interlocks per bank by default. • Use the sheets and interlocks that you need; delete the remainder.
IO Faults	The logic monitors the input and output modules and channels that are used to interface with the device for fault conditions and raises an alarm on an I/O fault.

CS_PVLVMP Sheet



PVLVMP Input References

Parameter	Description
XV220_Inp_OpenLS	Valve Open Limit Switch, 1 = confirmed open
XV220_Inp_Closed_LS	Valve Closed Limit Switch, 1 = confirmed closed
XV220_Inp_LowerLS	Valve Lower Seat Lift Limit Switch, 1 = confirmed lower seat lifted
XV220_Inp_UpperLS	Valve Upper Seat Lift Limit Switch, 1 = confirmed upper seat lifted
XV220_Inp_CavInLS	Valve cavity inlet limit switch: 1 = Confirmed cavity inlet opened.
XV220_Inp_CavOutLS	Valve cavity outlet limit switch: 1 = Confirmed cavity output closed
IOFault	Input connection from I/O Faults sheet
PermOK	Input connection from Permissives sheet 1 = Permissives OK, valve can move from the closed position
NBPermOK	Input connection from Permissives sheet 1 = Non-bypassable permissives OK, valve can move from the closed position
XV220_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to move valve from the closed position 0 = Close valve
XV220_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to move valve from the closed position 0 = Close valve
XV220_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
XV220_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit – closes valve but does not raise trip alarm
XV220_Intlk_BankSts.Sts_RdyReset	Interlock bank status 1 = A latched interlock (returned to OK) is ready to be reset

PVLVMP Output References

Parameter	Description
XV220_Out_Open	Output to Open valve, 1 = Open
XV220_Out_Closed	Output to Close valve, 1 = Close
XV220_Out_LiftLower	Output to Lift lower valve seat, 1 = Lift
XV220_Out_LiftUpper	Output to Lift upper valve seat, 1 = Lift
XV220_Out_Horn	1 = Sound audible before commanded valve action
XV220_Out_CavIn	Cavity In Output
XV220_Out_CavOut	Cavity Out Output
BypActive	Output connection to permissive and interlock bank sheets

PVLVMP Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_DISCRETE_MIX_PROOF	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

Open Interlock Bank Sheet

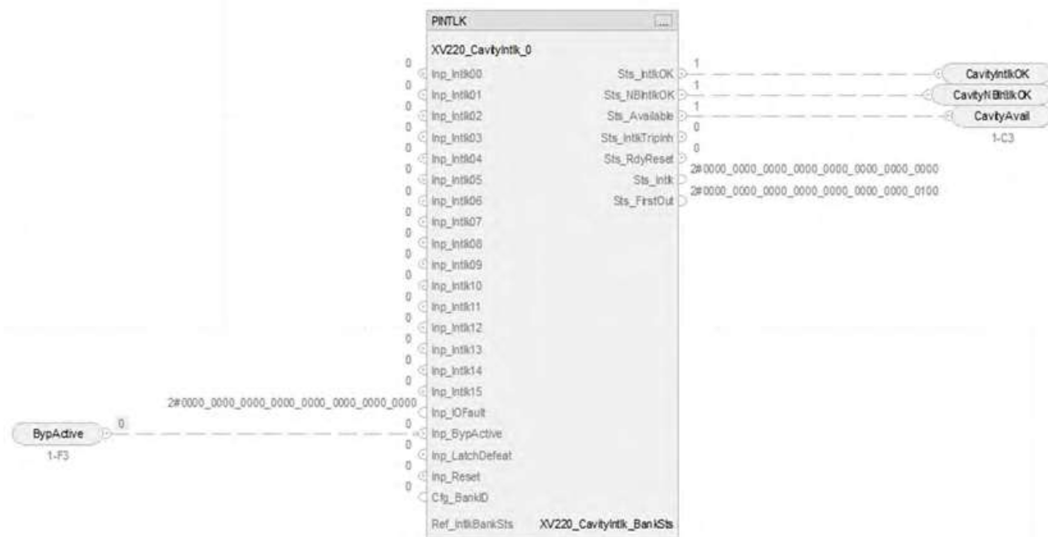
PINTLK Input References

Parameter	Description
BypActive	Input connection from CS_PVLVMP sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

Cavity Interlocks Bank Sheet



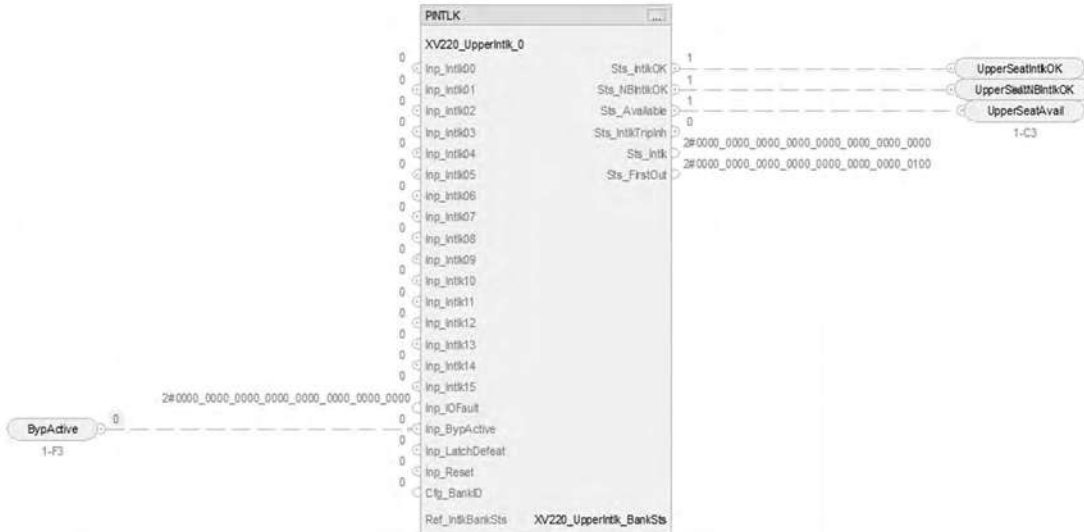
PINTLK Input References

Parameter	Description
BypActive	Input connection from CS_PVLVMP sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

Upper Seat Interlocks Bank Sheet



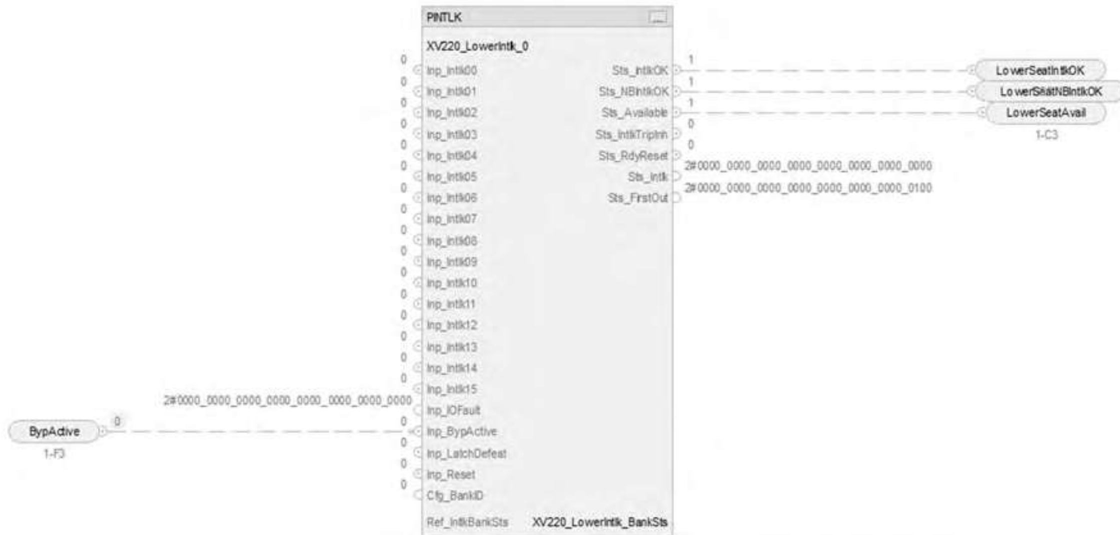
PINTLK Input References

Parameter	Description
BypActive	Input connection from CS_PVLVMP sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

Lower Seat Interlocks Bank Sheet



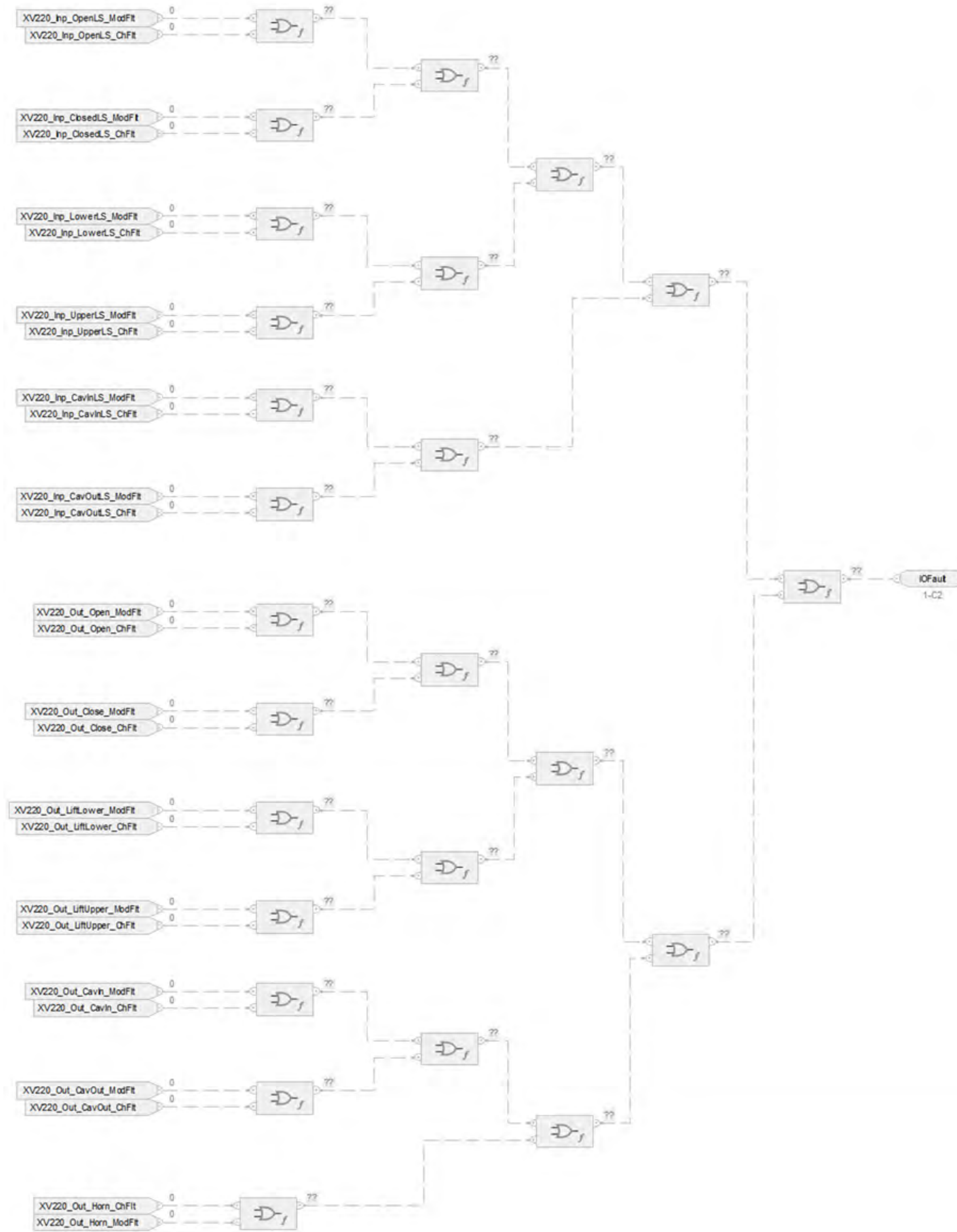
PINTLK Input References

Parameter	Description
BypActive	Input connection from CS_PVLVMP sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

I/O Faults Sheet



Fault Input References

Parameter	Description
XV220_Inp_OpenLS_ModFlt	
XV220_Inp_OpenLS_ChFlt	
XV220_Inp_Closed_LS_ModFlt	
XV220_Inp_Closed_LS_ChFlt	
XV220_Inp_LowerLS_ModFlt	
XV220_Inp_LowerLS_ChFlt	
XV220_Inp_UpperLS_ModFlt	
XV220_Inp_UpperLS_ChFlt	
XV220_Inp_CavInLS_ModFlt	Valve cavity inlet limit switch module fault
XV220_Inp_CavInLS_ChFlt	Valve cavity inlet limit switch channel fault
XV220_Inp_CavOutLS_ModFlt	Valve cavity outlet limit switch module fault
XV220_Inp_CavOutLS_ChFlt	Valve cavity outlet limit switch channel fault
XV220_Out_Open_ModFlt	Open limit switch module fault
XV220_Out_Open_ChFlt	Open limit switch channel fault
XV220_Out_Close_ModFlt	Closed limit switch module fault
XV220_Out_Close_ChFlt	Closed limit switch channel fault
XV220_Out_LiftLower_ModFlt	Lift lower limit switch module fault
XV220_Out_LiftLower_ChFlt	Lift lower limit switch channel fault
XV220_Out_LiftUpper_ModFlt	Lift upper limit switch module fault
XV220_Out_LiftUpper_ChFlt	Lift upper limit switch channel fault
XV220_Out_CavIn_ModFlt	
XV220_Out_CavIn_ChFlt	
XV220_Out_CavOut_ModFlt	
XV220_Out_CavOut_ChFlt	
XV220_Out_Horn_ChFlt	Audible output device channel fault
XV220_Out_Horn_ModFlt	Audible output device module fault

Fault Output References

Parameter	Description
IOFault	Output connection to CS_PVLVMP sheet

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

Process Valve Solenoid Operated (PVLVSO) Control Strategy

Use the PVLVSO control strategy to operate (open and close) one solenoid-operated valve. Generally, a solenoid-operated valve only requires one output to energize a solenoid providing pneumatic energy to an actuator that moves the valve from its fail-safe position. When this output is de-energized, a spring forces the valve back to its fail-safe position. When using this control strategy, one must consider whether the valve is Fail Closed (FC) or Fail Open (FO). For the more common FC valve, the output XV101_Out_Pos2 must be used to drive the field device. If the valve is a FO valve, the output XV101_Out_Pos1 must be used to drive the field device.

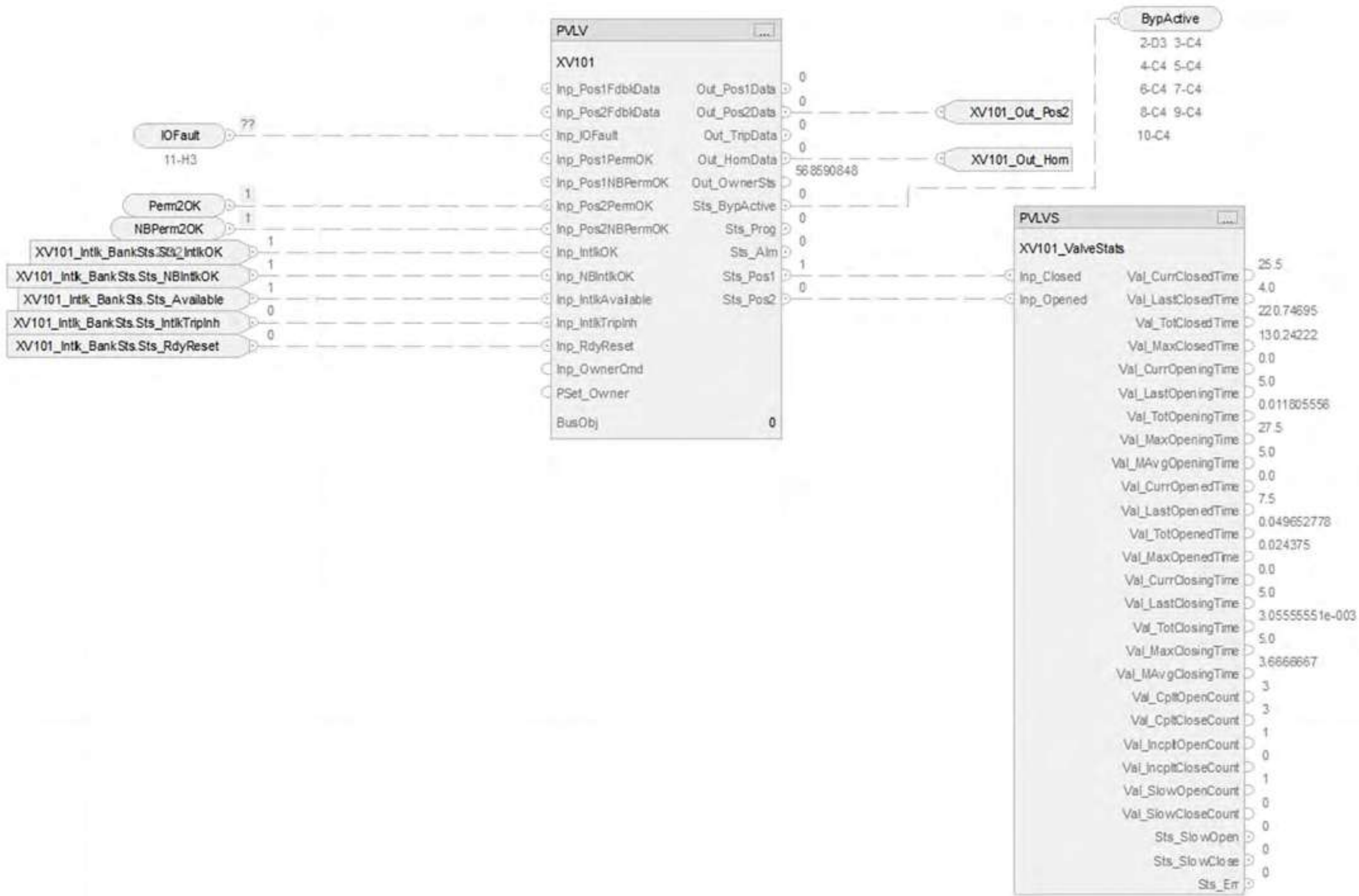
The CS_PVLVSO control strategy is available as two routines in the process library:

- One routine for fail position 1, and one routine for fail position 2. Fail position 1 and 2 could be Close/Open, Up/ Down, Left/ Right depending on the application.
- Import the control strategy as a **routine** in your controller project.

The PVLVSO control strategy contains these Function Block sheets:

Sheet	Description
CS_PVLVSO	Process Valve instruction, solenoid operated
Permissives	Process Permissives instruction The Process Permissives (PPERM) instruction collects, or sums up, the permissive conditions that let a piece of equipment energize. In most cases, permissive conditions must be true to energize equipment. Once the equipment is energized, permissives are ignored.
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The PVLV instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder
IO Faults	The logic monitors one analog output channel for I/O fault input and raises an alarm on an I/O fault.

CS_PVLVSO_FailPos1 Sheet



PVLV Input References

Parameter	Description
XV101_Inp_Pos1Fdbk XV101_Inp_Pos2Fdbk	Feedback from Position limit switches of the device 1 = Device confirmed Position 1
IOFault	Input connection from IO Faults sheet
Perm2OK	Input connection from Position 2 Permissives sheet 1 = On permissives OK, device can turn On
NBPerm2OK	Input connection from Position 2 Permissives sheet 1 = Non-bypassable On permissives OK, device can turn On
XV101_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
XV101_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
XV101_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
XV101_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
XV101_Intlk_BankSts.Sts_RdyReset	Interlock bank status 1 = A latched interlock (returned to OK) is ready to be reset

PVLV Output References

Parameter	Description
XV101_Out_Pos2	1 = Activate to move valve to Position 2
XV101_Out_Pos1	1 = Activate to move valve to Position 1
XV101_Out_Horn	1 = Sound audible before commanded valve start
BypActive	Output connection to permissives and interlock bank sheets

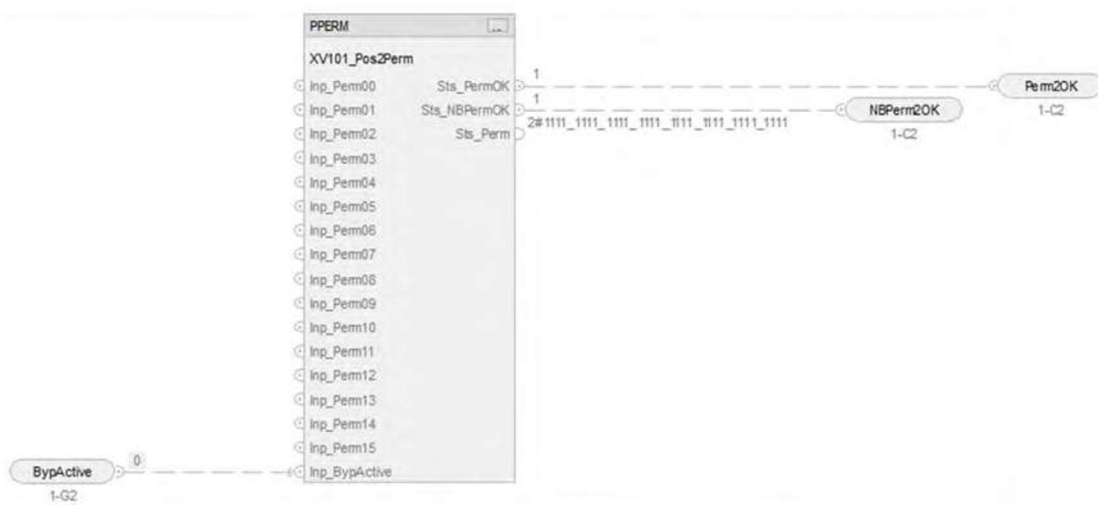
PVLV Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_VALVE_DISCRETE	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

PVLV Output References to PVLVS

Parameter	Description
Sts_Pos1	1 = Valve requested to Position 1 and is confirmed Position 1
Sts_Pos2	1 = Valve requested to Position 1 and is confirmed Position 2

Position 2 Permissive Sheet



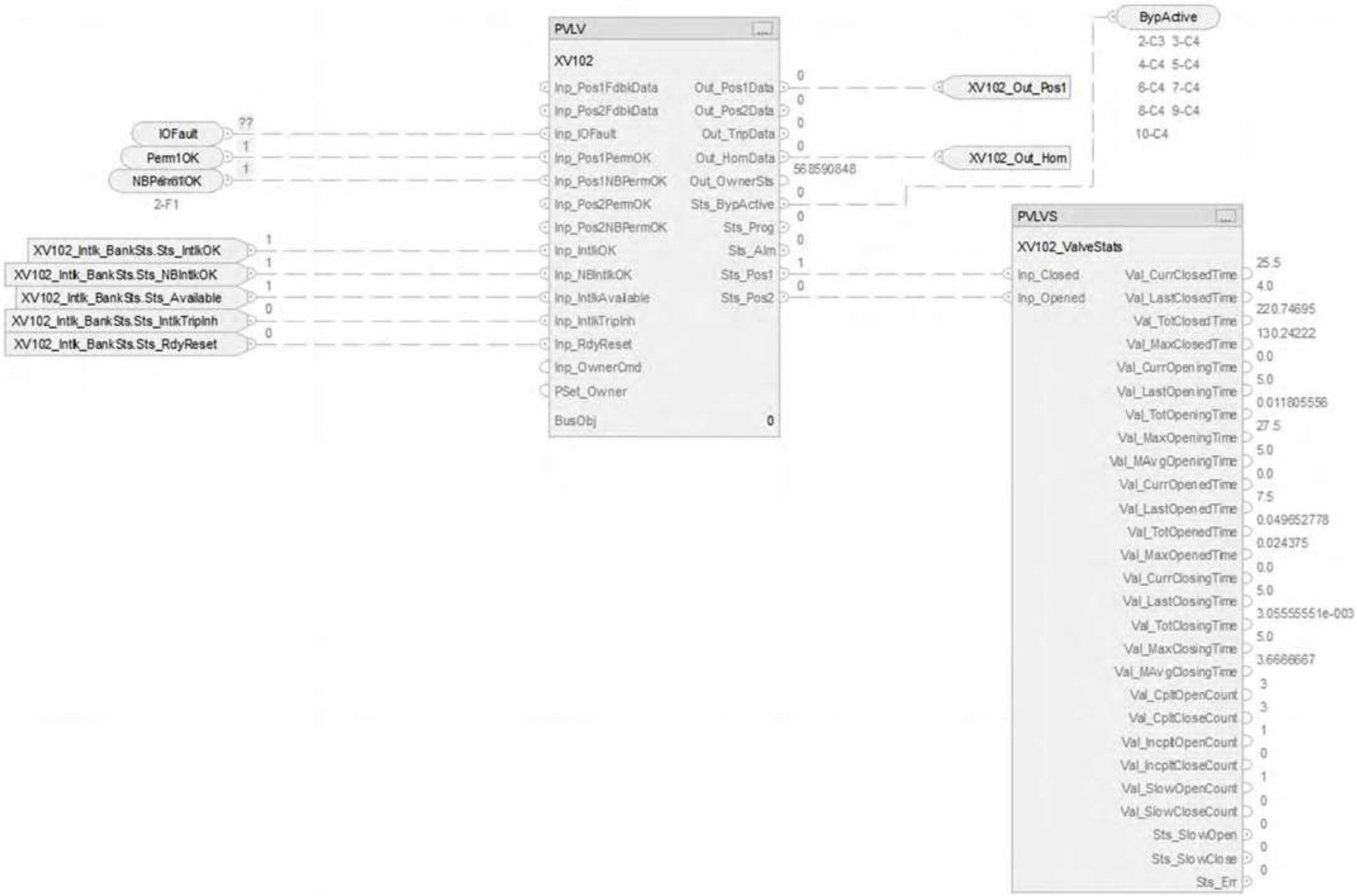
PPERM Input References

Parameter	Description
BypActive	Input connection from the interlock bank sheet

PPERM Output References

Parameter	Description
Perm2OK	Overall permissive status (1 = OK to energize)
NBPerm2OK	Non-bypassable permissive status (1 = all non-bypassable permissives OK to energize)

CS_PVLVSO_FailPos2 Sheet



PVLV Input References

Parameter	Description
XV102_Inp_Pos1Fdbk XV102_Inp_Pos2Fdbk	Feedback from Position limit switches of the device 1 = Device confirmed Position 1
IOFault	Input connection from IO Faults sheet
Perm2OK	Input connection from Position 2 Permissives sheet 1 = On permissives OK, device can turn On
NBPerm2OK	Input connection from Position 2 Permissives sheet 1 = Non-bypassable On permissives OK, device can turn On
XV102_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
XV102_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
XV102_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
XV102_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
XV102_Intlk_BankSts.Sts_RdyReset	Interlock bank status 1 = A latched interlock (returned to OK) is ready to be reset

PVLV Output References

Parameter	Description
XV102_Out_Pos2	1 = Activate to move valve to Position 2
XV102_Out_Pos1	1 = Activate to move valve to Position 1
XV102_Out_Horn	1 = Sound audible before commanded valve start
BypActive	Output connection to permissives and interlock bank sheets

PVLV Configuration Considerations

Operand	Type	Description
PlantPAx control	P_VALVE_DISCRETE	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

PVLV Output References to PVLVS

Parameter	Description
Sts_Pos1	1 = Valve requested to Position 1 and is confirmed Position 1
Sts_Pos2	1 = Valve requested to Position 1 and is confirmed Position 2

Position 1 Permissive Sheet

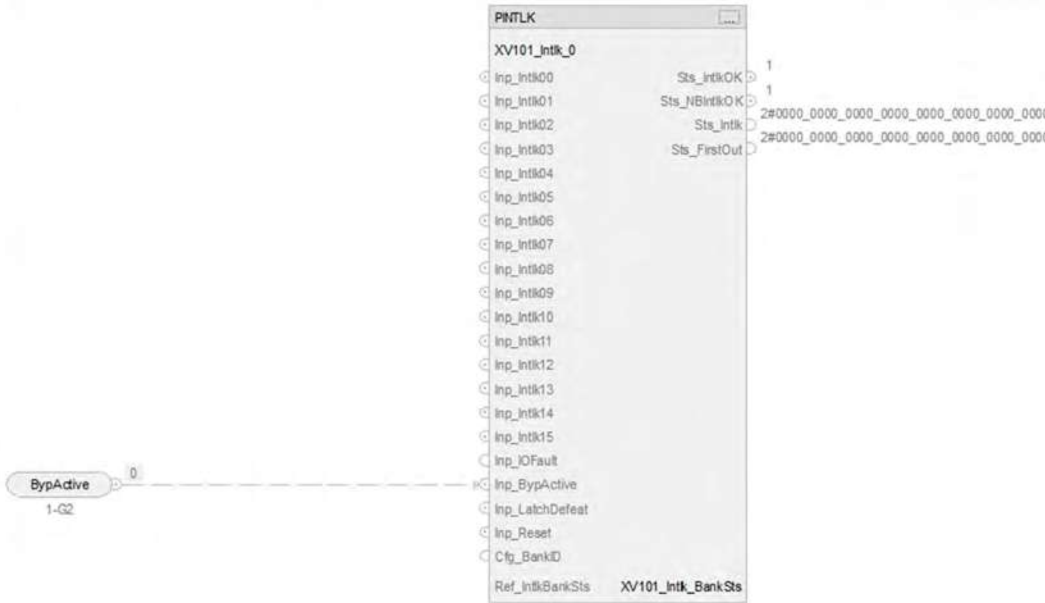
PPERM Input References

Parameter	Description
BypActive	Input connection from the interlock bank sheet

PPERM Output References

Parameter	Description
Perm1OK	Overall permissive status (1 = OK to energize)
NBPerm1OK	Non-bypassable permissive status (1 = all non-bypassable permissives OK to energize)

Interlock Bank Sheet-XV101



PINTLK Input Reference

Parameter	Description
BypActive	Input connection from CS_PVLVSO sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

Interlock Bank Sheet-XV102

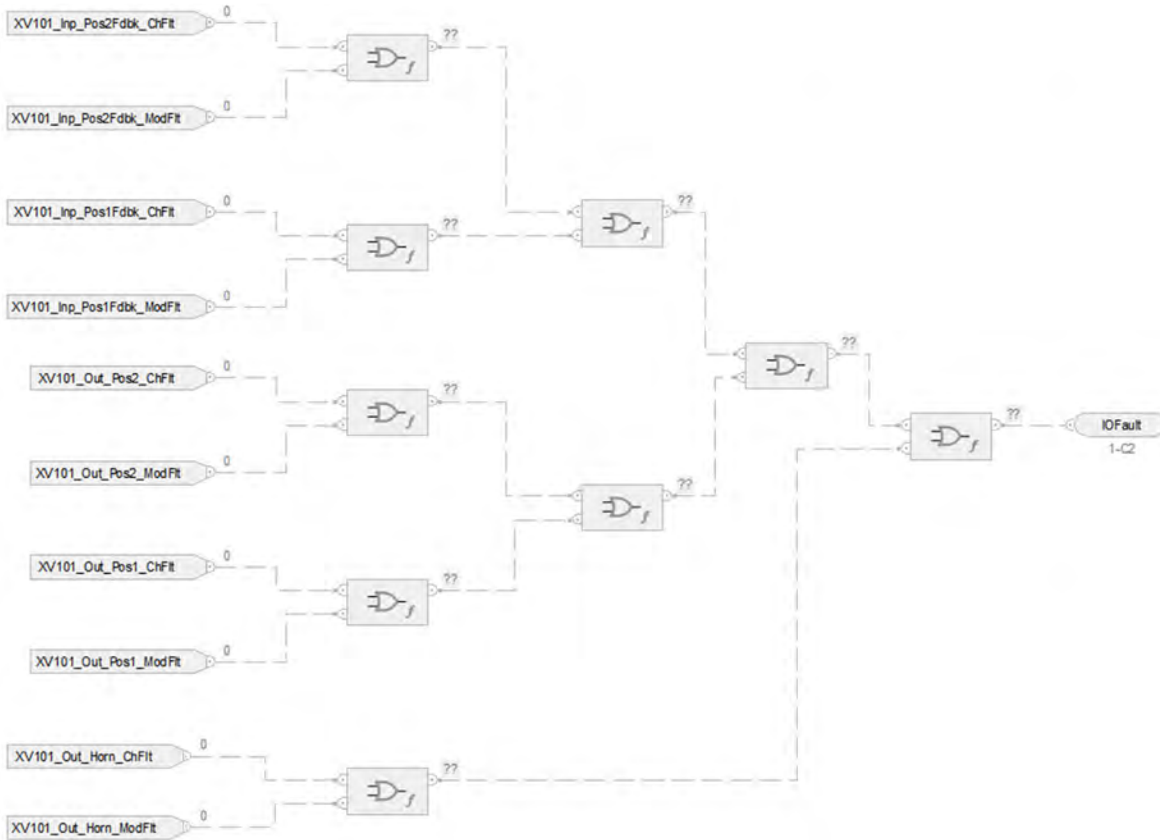
PINTLK Input Reference

Parameter	Description
BypActive	Input connection from CS_PVLVSO sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

IO Faults Sheet-XV101



Fault Input References

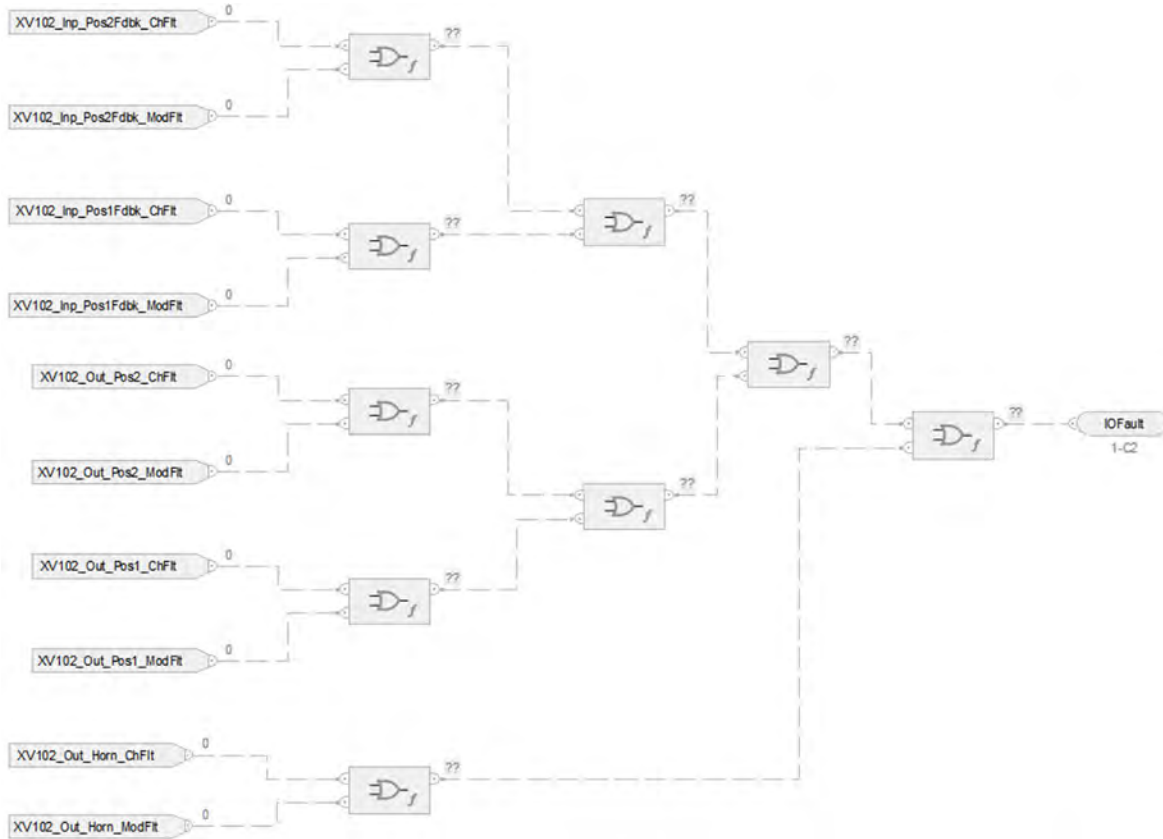
Parameter	Description
XV101_Inp_Pos2Fdbk_ChFlt	Tieback input 2 channel fault
XV101_Inp_Pos2Fdbk_ModFlt	Tieback input 2 module fault
Local:6:I.Fault.8	Discrete input fault
Local_06.Sts_IOFault	Discrete input communication faulted
Local:7:I.Fault.8	Discrete output fault
Local_07.Sts_IOFault	Discrete output communication faulted
XV101_Out_Pos1_ChFlt	Position 1 channel fault
XV101_Out_Pos1_ModFlt	Position 1 module fault
XV101_Out_Horn_ChFlt	Sound audible for output channel fault
XV101_Out_Horn_ModFlt	Sound audible for output module fault

Fault Output References

Parameter	Description
IOFault	Output connection to CS_PVLVSO sheet

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

IO Faults Sheet-XV102



Fault Input References

Parameter	Description
XV102_Inp_Pos2Fdbk_ChFlt	Tieback input 2 channel fault
XV102_Inp_Pos2Fdbk_ModFlt	Tieback input 2 module fault
Local:6:I.Fault.8	Discrete input fault
Local_06.Sts_IOFault	Discrete input communication faulted
Local:7:I.Fault.8	Discrete output fault
Local_07.Sts_IOFault	Discrete output communication faulted
XV102_Out_Pos1_ChFlt	Position 1 channel fault
XV102_Out_Pos1_ModFlt	Position 1 module fault
XV102_Out_Horn_ChFlt	Sound audible for output channel fault
XV102_Out_Horn_ModFlt	Sound audible for output module fault

Fault Output References

Parameter	Description
IOFault	Output connection to CS_PVLVSO sheet

For examples on how to map data to input tags see: PlantPAx Control Strategies on page 17.

Notes:

Process Variable Speed Drive (PVSD) Control Strategies

Use a PVSD control strategy to monitor and control a variable speed motor using an AC (variable frequency) or DC drive. Use the instruction to run or jog the motor forward or reverse. The drive interface can be through a Device Object Interface or through individual pins.

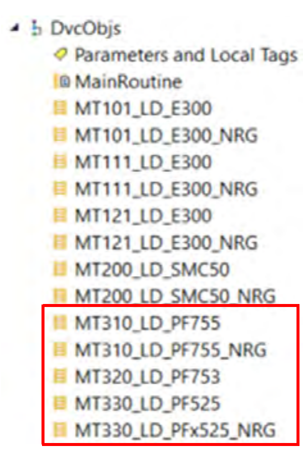
The following PVSD control strategies are available as routines in the process library:

Drive Type	Control Strategy
PowerFlex® 525	CS_PVSD525 (basic) CS_PVSD525_Hand (Hand command source) CS_PVSD525_Energy (energy parameters)
PowerFlex 753	CS_PVSD753 (basic) CS_PVSD753_Hand (Hand command source)
PowerFlex 755	CS_PVSD755 (basic) CS_PVSD755_Hand (Hand command source) CS_PVSD755_Energy (energy parameters)

Import the appropriate control strategy as a **routine** in your controller project.

Also, import the appropriate device object as a routine in your controller project. These objects are from the Power Device Library and must be downloaded separately from the PlantPAx® Process Library.

Each '_NRG' object uses the Energy object to group energy parameters for the device. Use this object with the corresponding, energy-related control strategy.



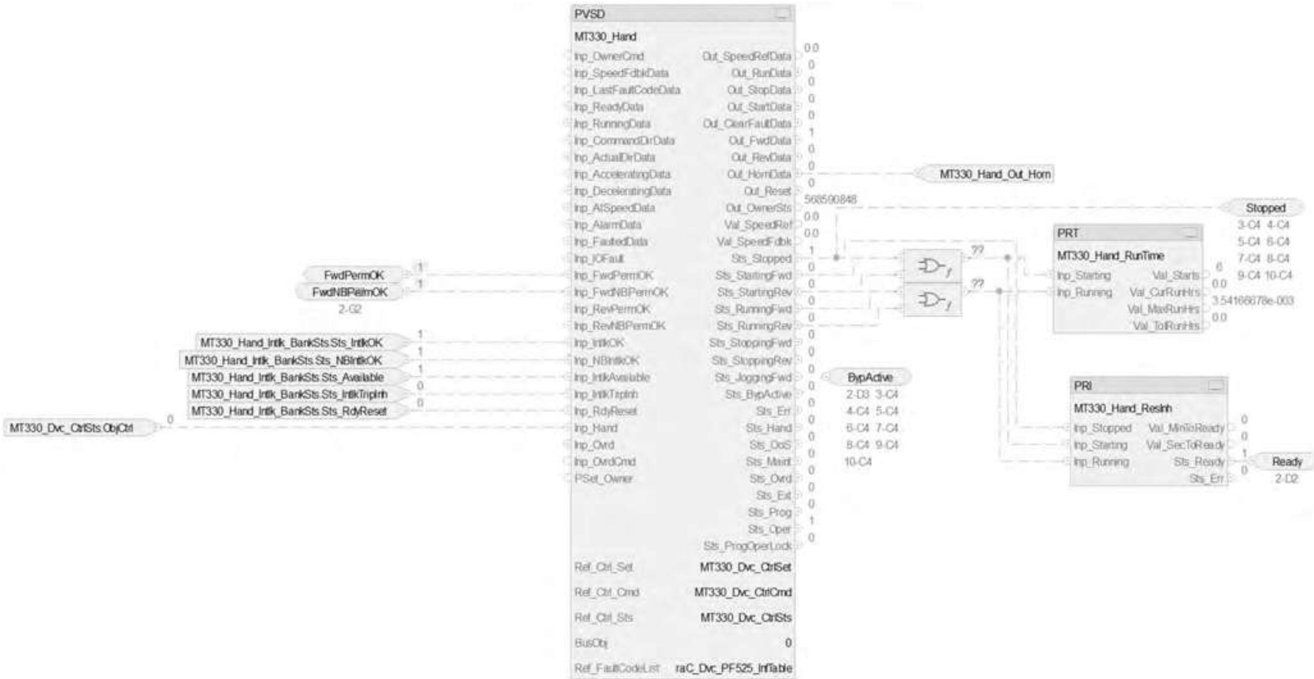
The PVSD control strategies contain these Function Block sheets:

Sheet	Description
CS_PVSD	Process Variable Speed Drive instruction
Permissive	Process Permissives instruction The Process Permissives (PPERM) instruction collects, or sums up, the permissive conditions that let a piece of equipment energize. In most cases, permissive conditions must be true to energize equipment. Once the equipment is energized, permissives are ignored.
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The PVSD instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder

In the input and output reference descriptions on each sheet, [device] = one of the following:

Drive Type	Type
PowerFlex 525	MT330
PowerFlex 753	MT320
PowerFlex 755	MT310

CS_PVSD Sheet



PVSD Input References

Parameter	Description
FwdPermOK	Input connection from Forward Permissives sheet 1 = On permissives OK, device can turn On
FwdNBPermOK	Input connection from Forward Permissives sheet 1 = Non-bypassable On permissives OK, device can turn On
[device]_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
[device]_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
[device]_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
[device]_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
[device]_Intlk_BankSts.Sts_RdyReset	Interlock bank status 1 = A latched interlock (returned to OK) is ready to be reset
[device]_Dvc_CtrlSts.ObjCtrl	Hand command source only 1 = Acquire Hand (typically hardwired local) 0 = Release Hand

PVSD Output References

Parameter	Description
[device]_Out_Horn	1 = Sound audible before commanded state change
BypActive	Output connection to permissives and interlock bank sheets
Ready	Output connection to the permissive sheet
Stopped	Output connection to interlock bank sheet

The Boolean OR performs a bitwise OR based on these PVSD outputs:

- Sts_Stopped
- Sts_StartingFwd
- Sts_StartingRev
- Sts_RunningFwd
- Sts_RunningRev

The result feeds these instructions:

Instruction	Description
Process Run Time and Start Counter (PRT)	The PRT instruction records the total run time and number of instances the drive starts.
Process Restart Inhibit (PRI)	The PRI instruction helps prevent the drive from starting repeatedly. Continual starts or start attempts in a short period overheat the motor windings and damage the motor.

PVSD Configuration Considerations

Operand	Type	Description
PlantPAx control	P_VARIABLE_SPEED_DRIVE	Instance of data structure (backing tag) required for proper operation of instruction
BusObj	BUS_OBJ	Bus component for organization control • 0 if not using organization • Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.
Ref_Ctrl_Set	RAC_ITF_DVC_PWRVELOCITY_SET	Velocity Automation Device Object Settings Interface Preconfigured in the device object ladder routine
Ref_Ctrl_Cmd	RAC_ITF_DVC_PWRVELOCITY_CMD	Velocity Automation Device Object Command Interface Preconfigured in the device object ladder routine
Ref_Ctrl_Sts	RAC_ITF_DVC_PWRVELOCITY_STS	Velocity Automation Device Object Status Interface Preconfigured in the device object ladder routine
Ref_FaultCodeList	RAC_CODEDESCRIPTION[400]	Fault Code to Fault Description lookup table for the drive Preconfigured in the device object ladder routine

Permissive Sheet

PPERM Input References

Parameter	Description
Ready	Input connection from the CS_PVSD sheet
BypActive	Input connection from the interlock bank sheet

PPERM Output References

Parameter	Description
FwdPermOK	Overall permissive status (1 = OK to energize)
FwdNBPermOK	Non-bypassable permissive status (1 = all non-bypassable permissives OK to energize)

Interlock Bank Sheet

PINTLK Input Reference

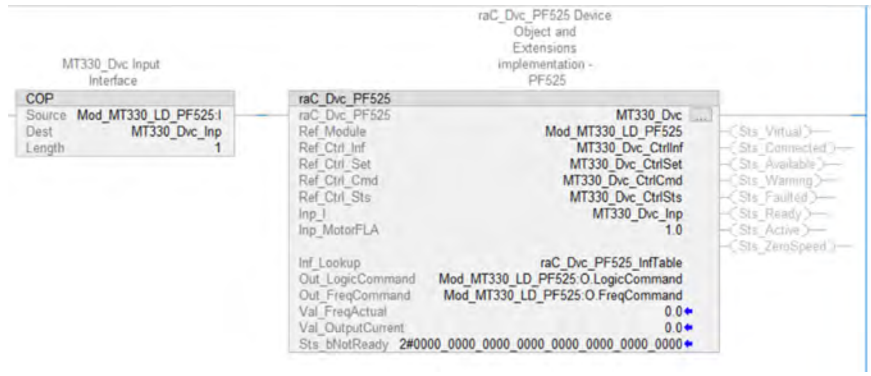
Parameter	Description
BypActive	Input connection from CS_PVSD sheet
Stopped	Input connection from the CS_PVSD sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

Drive Device Objects

Drive



Drive with Energy Parameters

Notes:

Ramp/Soak (RMPS) Control Strategy

Ramp/Soak refers to the ramping of a controller setpoint to a final target at a predefined rate where it is held for a specified time. This strategy is typically used to control temperature where temperature is "ramped" (increased or decreased) at a predefined rate, and once reaching the target temperature the setpoint is "soaked" (held at temperature for a specified time). The RMPS control strategy can be used to manage multiple segments of alternating ramp and soak periods.

The following RMPS control strategies are available as routines in the process library:

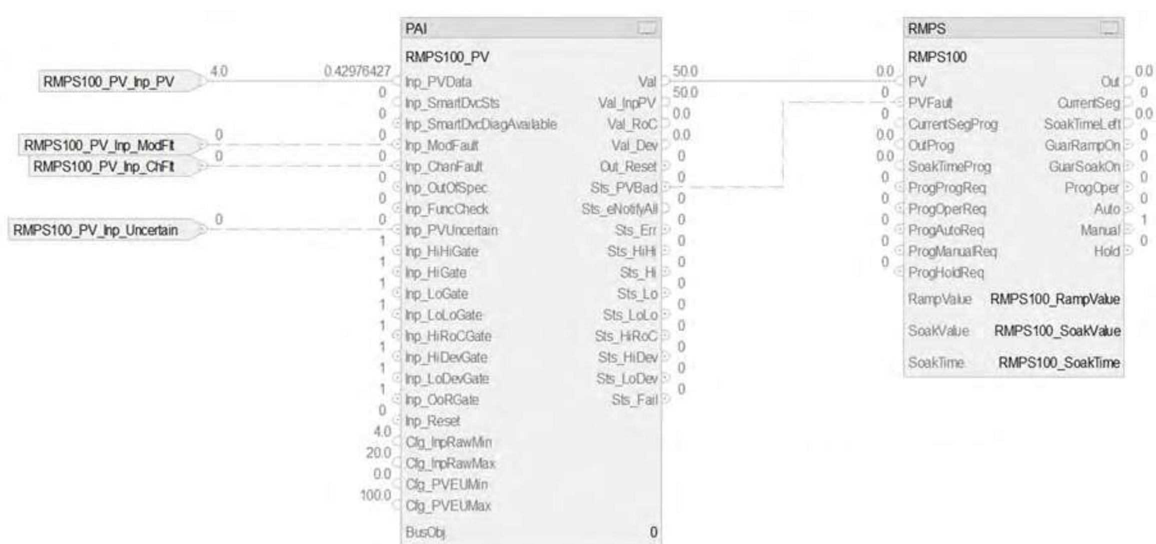
- CS_RMPS
- CS_RMPS_HART

Import the appropriate control strategy as a **routine** in your controller project.

The RMPS control strategy contains one Function Block sheet:

Sheet	Description
CS_RMPS	Ramp/Soak instruction
CS_RMPS_HART	Ramp/Soak instruction with HART input

CS_RMPS Sheet



PAI Input References

See CS_PAI Sheet on page 71 for details.

PAI Outputs to RMPS Inputs

Parameter	Description
Val	Analog input value in engineering units (after Substitute PV, if used). Extended Properties of this member: Units - Engineering units (text) used for the analog input.
Sts_PVBad	Quality of PV value 1 = PV quality is flagged as Bad

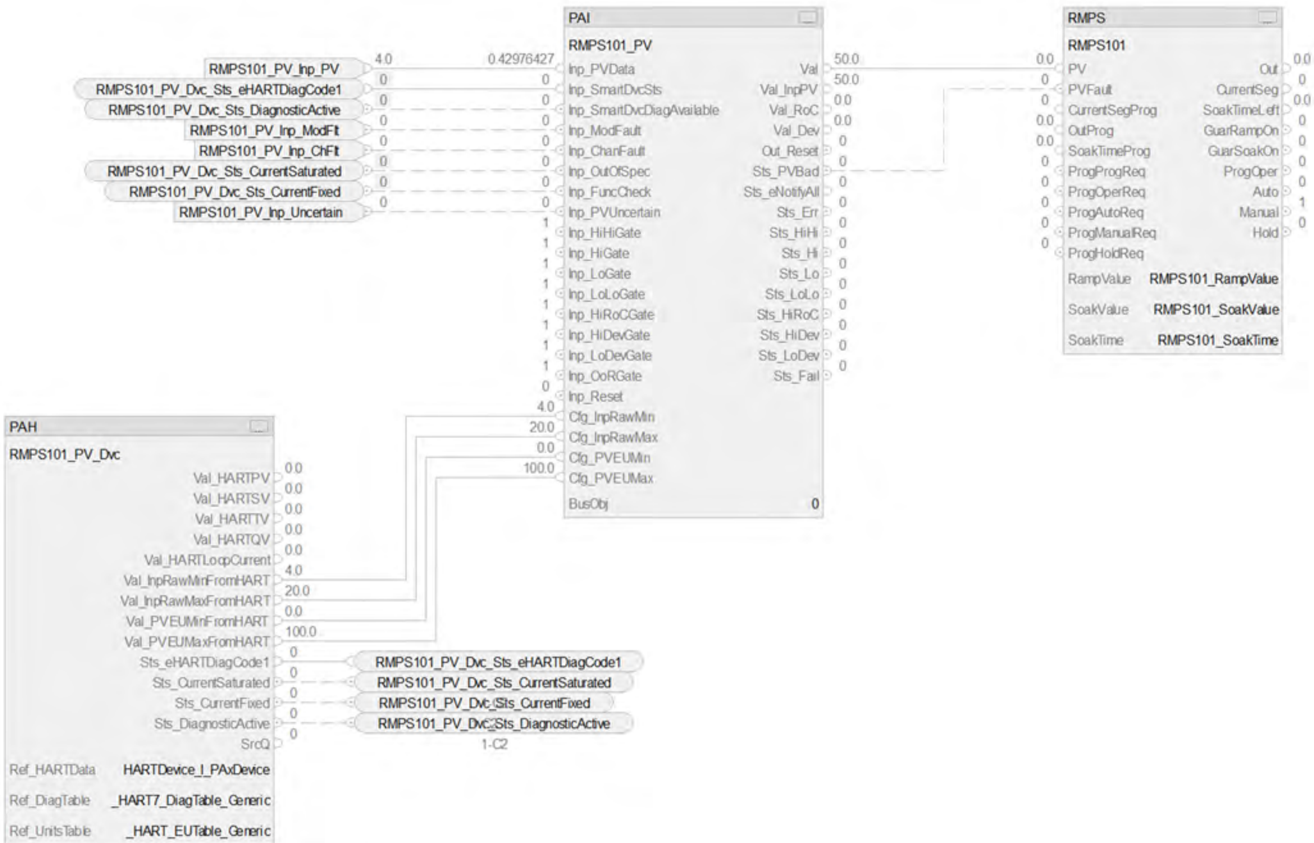
RMPS Output Reference

Parameter	Description
RMPS100_Out	The output of the ramp/soak instruction

RMPS Configuration Considerations

Operand	Type	Description
RMPS Tag	RAMP_SOAK	Instance of data structure (backing tag) required for proper operation of instruction. RMPS100 in this example corresponds to an instance of coordinating ramp/soak segments.
RampValue	REAL array	Enter a ramp value for each segment (0 to NumberOfSegs-1). Ramp values are entered as time in minutes or as a rate in units/minute. The TimeRate parameter reflects which method is used to specify the ramp. If a ramp value is invalid, the instruction sets the appropriate bit in Status and changes to Operator Manual or Program Hold mode. The array must be at least as large as NumberOfSegs. Valid = 0.0 to maximum positive float
SoakValue	REAL array	Enter a soak value for each segment (0 to NumberOfSegs-1). The array must be at least as large as NumberOfSegs. Valid = any float
SoakTime	REAL array	Enter a soak time for each segment (0 to NumberOfSegs-1). Soak times are entered in minutes. If a soak value is invalid, the instruction sets the appropriate bit in Status and changes to Operator Manual or Program Hold mode. The array must be at least as large as NumberOfSegs.

CS_RMPS_HART Sheet



The CS_RMPS_HART control strategy operates the same as the CS_RMPS control strategy but relies on HART input data.

- For information on PAH outputs to PAI inputs, see CS_PAI_HART Sheet on page 72.
- Substitute for RMPX101 for XT100
- For more information, see HART Integration on page 31.

Notes:

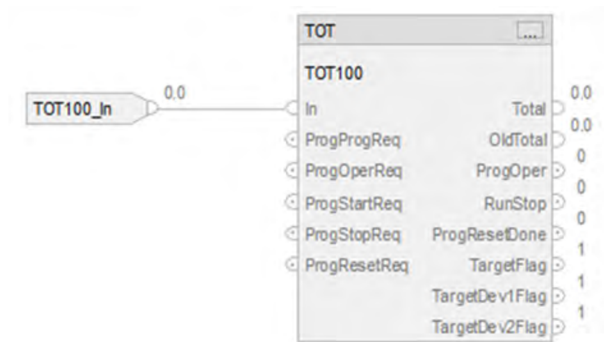
Totalizer (TOT) Control Strategy

Use the TOT control strategy to provide a time-scaled accumulation of an analog input value.

The CS_TOT control strategy is available as a routine in the process library. Import the appropriate control strategy as a **routine** in your controller project.

The TOT control strategy contains one Function Block sheet:

CS_TOT Sheet



TOT Input Reference

Parameter	Description
In	The analog signal input to the instruction

TOT Configuration Considerations

Operand	Type	Description
TOT Tag	TOTALIZER	Instance of data structure (backing tag) required for proper operation of instruction

Notes:

Process Area Control Strategy

The Process Area Add-On Instruction groups Units together, aggregates status from Unit objects, and broadcasts commands to Unit objects.

You can consolidate the status from groups of equipment, and display the consolidated status on an HMI. These status items include:

- Alarm Status
- Alarm Severity
- Mode
- Configuration Errors
- Prompt Status

You can also manage any of the following functions for a group of equipment with a global set of commands:

- Mode
- Alarm Acknowledge
- Alarm Reset
- Enable/Disable Alarms

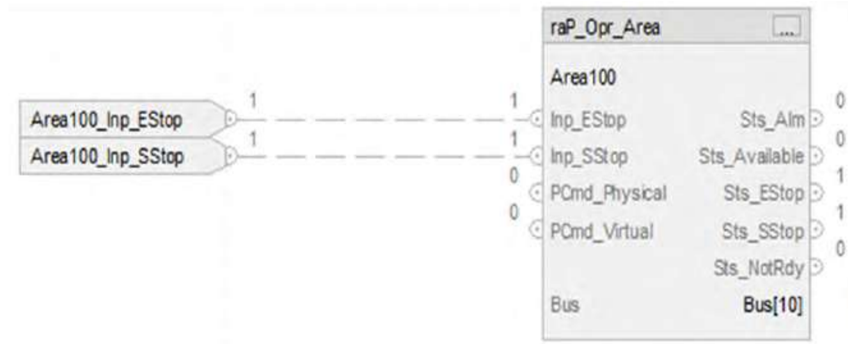
The CS_raP_Opr_Area control strategy is available as a routine in the process library. Import the appropriate control strategy as a **routine** in your controller project.

The Process Area control strategy contains these routines:

Routine	Description
Area100	Function Block control strategy routine.
ExtddAlarms	Contains instances of external alarms and trigger logic.

CS_raP_Opr_Area Sheet

The Area100 routine contains the CS_raP_Opr_Area sheet.



CS_raP_Opr_Area Input Reference

Parameter	Description
Area100_Inp_EStop	1 = Emergency stop input ok.
Area100_Inp_SStop	1 = Software stop input ok.

CS_raP_Opr_Area Configuration Considerations

Operand	Type	Description
PlantPAx® control	raP_Opr_Area	Instance of data structure (backing tag) required for proper operation of instruction
Bus	raP_UDT_Opr_Bus	Bus component for organization control 0 if not using organization Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.

Extended Alarms Routine

The raP_Opr_ExtddAlm (Extended Alarm Block) Add-On Instruction is used to provide notification to operators of abnormal conditions or events for up to 32 additional items external to a parent object. For more information, see the Rockwell Automation Library of Process Objects, publication PROCES-RM200.

This instruction handles the connections of the commands from the parent object:

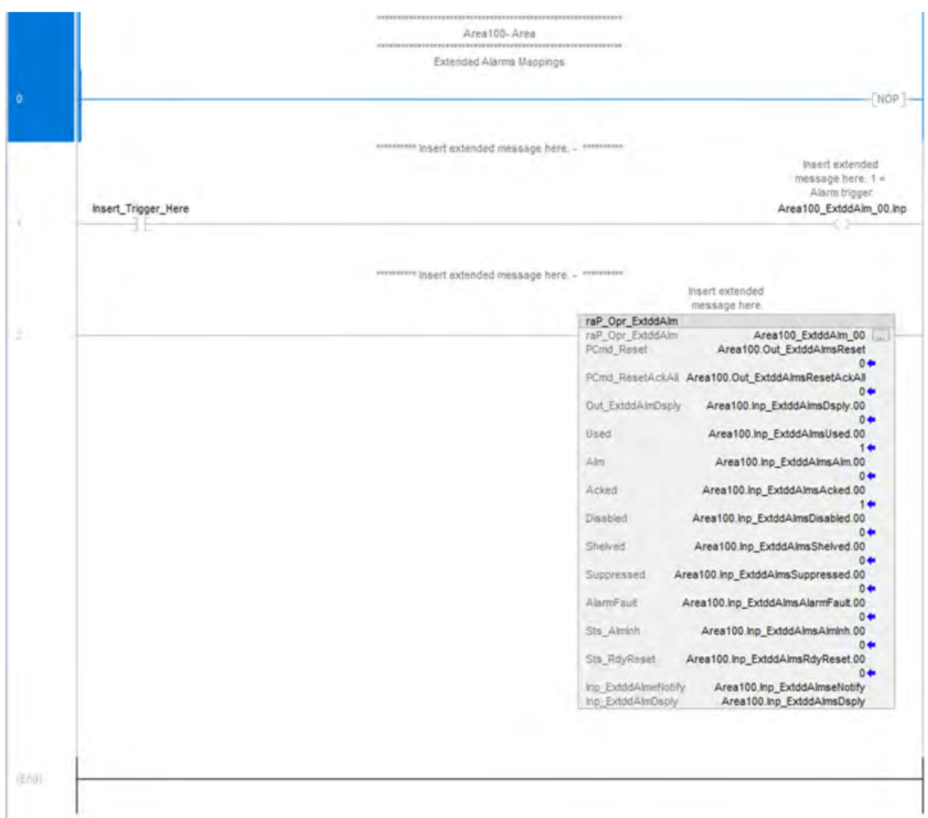
- Acknowledge
- Reset
- Enabling/Disabling
- Suppress/Unsuppress
- UnShelve

This instruction handles the connections of the status from the raP_Opr_ExtddAlm:

- Used
- Alarm
- Acknowledged
- Disabled
- Suppressed
- Shelved
- Alarm Fault
- Ready for Reset
- Notify value

raP_Opr_ExtddAlm Parameters

Parameter	Description
PCmd_Reset	Program command to reset alarm request.
PCmd_ResetAckAll	Program command to reset and acknowledges all alarms. The instruction clears this operand automatically.
Out_ExtddAlmDsply	1 = Extended alarm severity value is greatest of all extended alarms active. Each bit represents an individual alarm (0…31).
Used	1 = Used.
Alm	1 = Alarm is active.
Acked	1 = In alarm acknowledged.
Disabled	1 = Alarm disabled.
Shelved	1 = Alarm shelved.
Suppressed	1 = Alarm suppressed.
AlarmFault	1 = Alarm fault.
Sts_AlmInh	1 = One or more alarms shelved, disabled, or suppressed.
Sts_Rdy_Reset	1 = A latched alarm condition is ready to be reset.
Inp_ExtddAlmeNotify	Extended alarms status enumerate values: 0 = Not in alarm 1 = Not in alarm unacknowledged or reset requires 2 = Low severity alarm acknowledged 3 = Low severity alarm unacknowledged 4 = Medium severity alarm acknowledged 5 = Medium severity alarm unacknowledged 6 = High severity alarm acknowledged 7 = High severity alarm unacknowledged 8 = Urgent severity alarm acknowledged 9 = Urgent severity alarm unacknowledged
Inp_ExtddAlmDsply	1 = Extended alarm severity value is greatest of all extended alarms active. Each bit represents an individual alarm (0…31).



Notes:

Chapter 40

Generic Equipment Module (EMGEN) Control Strategy

An equipment module is a functional group of equipment that can carry out a finite number of specific minor processing activities. An equipment module is typically centered around a piece of process equipment (a weigh tank, a process heater, a scrubber, etc.). This term applies to both the physical equipment and the equipment entity.

The CS_raP_Opr_EMGen control strategy controls an Equipment Module in a variety of modes and monitors for fault conditions.

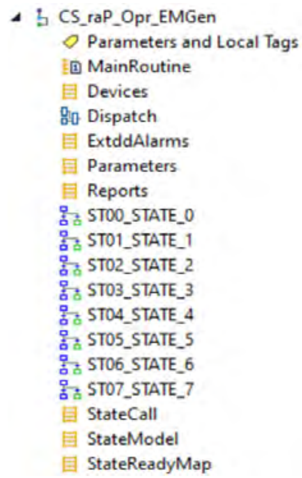
Use when:

- You want to group equipment, and you want to apply a custom state model.
- You want to provide the following for a group of equipment:
  - Apply a mode model to the equipment group.
  - Definable Commands and states.
  - Apply interlocks and/or permissives to the group of equipment.
  - Parameters that define the behavior of the group of equipment.
  - Report resultant data from the group of equipment.
  - A faceplate that allows monitoring and control of the equipment grouping.
  - Alarm if any device fails.
  - Monitor step (description), and allow forcing of steps in maintenance mode.
  - Allow configurable alarms for certain process / equipment failure conditions.

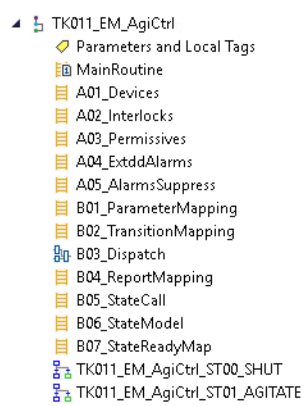
Do NOT use when:

- You need to sequence / coordinate a device, and do not require any of the above.
- You want to apply an ISA 88.01 state model to the equipment, use the CS_raP_Opr_EPGen_PHASE control strategy instead.
- You want to apply the PackML state model.

The EMGEN control strategy is available as a program in the process library:



Import the appropriate control strategy as a **program** in your controller project.

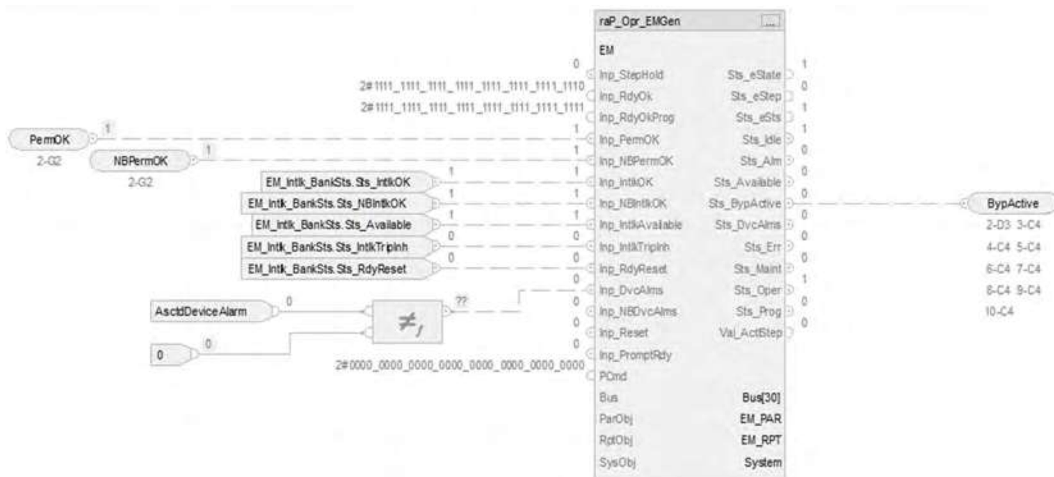
Routine	Description
Devices	Status of devices. Add logic appropriate to your application.
Dispatch	Contains raP_Opr_EMGen Add-On Instruction.
ExtddAlarms	Contains instances of external alarms and trigger logic.
Parameters	Contains raP_Opr_EMGen parameter mapping to and from Parameter blocks [_ParRpt (Enum, Integer, Real, String)] to raP_Opr_EMGen instance.
Reports	Contains raP_Opr_EMGen report mapping to and from Parameter blocks [_ParRpt (Enum, Integer, Real, String))] to raP_Opr_EMGen instance.
ST00_STATE_0…31	32 available user-defined routines that contain logic which sequences and coordinates devices (implement states as required). You can rename these routines appropriately for your project. 
StateCall	Calls the routine for the associated state when that state is active.
StateModel	Sets which state is active based upon the state request for that state and any other required conditions.
StateReadyMap	Equipment Module StateReadyMap Routine - Defines when each Equipment Module State is Available for both selection by the HMI and selection by Controller Logic. For each state number, 0 to 31: • EM.Inp_RdyOk.0 to 31 needs to be true for that state to be available to select from the HMI • EM.Inp_RdyOkProg.0 to 31 needs to be true for that state to be available to enter via Program Commands

Dispatch Routine

The Dispatch routine contains these Function Block sheets:

Sheet	Description
EMGEN100	Equipment Module Add-On Instruction
Permissives	Process Permissives instruction The Process Permissives (PPERM) instruction collects, or sums up, the permissive conditions that let a piece of equipment energize. In most cases, permissive conditions must be true to energize equipment. Once the equipment is energized, permissives are ignored.
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The instruction monitors by passable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder

Dispatch EPGEN100 Sheet



raP_Opr_EMGen Input References

Parameter	Description
PermOK	Input connection from Permissives sheet 0 (State 1) 1 = On permissives OK, device can turn On
NBPermOK	Input connection from Permissives sheet 0 (State 1) 1 = Non-bypassable On permissives OK, device can turn On
EM_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
EM_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
EM_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
EM_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
EM_Intlk_BankSts.Sts_RdyReset	Interlock bank status 1 = A latched interlock (returned to OK) is ready to be reset
AsctdDeviceAlarm	Associated Device Alarm Active if any Bits are Logic 1

raP_Opr_EMGen Output References

Parameter	Description
BypActive	Output connection to permissives and interlock bank sheets

raP_Opr_EMGen Configuration Considerations

Operand	Type	Description
PlantPAx control	raP_Opr_EMGen	Instance of data structure (backing tag) required for proper operation of instruction
Bus	raP_UDT_Opr_Bus	Bus component
ParObj	raP_UDT_Opr_ParRpt_Intfc	Optional parameter object interface. Link to routine
RptObj	raP_UDT_Opr_ParRpt_Intfc	Optional report object interface. Link to routine
SysObj	raP_UDT_Opr_System	System component.

Dispatch Permissive Sheet

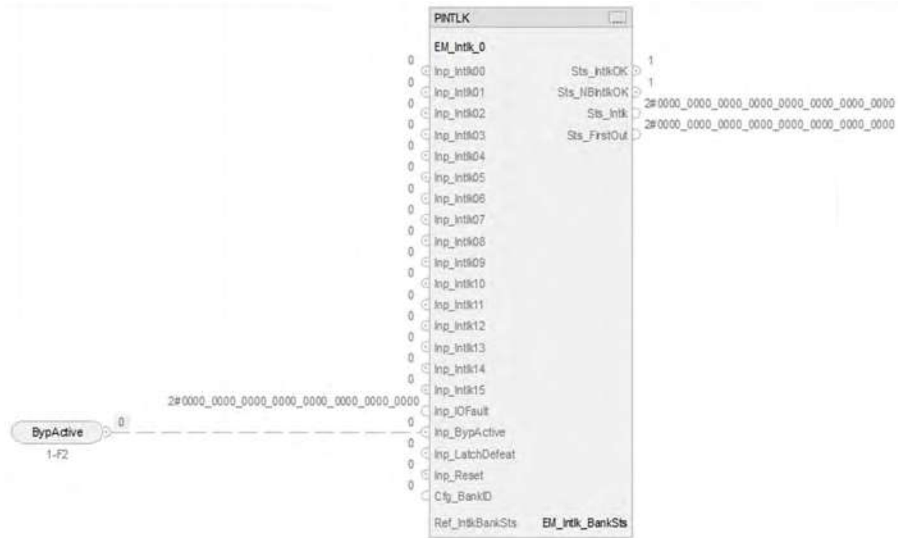
PPERM Input References

Parameter	Description
BypActive	Input connection from the EPGEN100 sheet

PPERM Output References

Parameter	Description
PermOK	Overall permissive status (1 = OK to energize)
NBPermOK	Non-bypassable permissive status (1 = all non-bypassable permissives OK to energize)

Dispatch Interlock Bank Sheet



PINTLK Input Reference

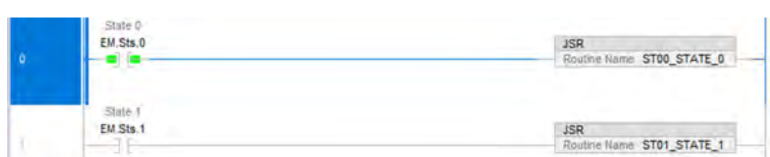
Parameter	Description
BypActive	Input connection from the CS_D4SD sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

State Routine Example

StateCall Routine Example



StateModel Routine Example

StateReadyMap Routine Example

Extended Alarms Routine

The raP_Opr_ExtddAlm (Extended Alarm Block) Add-On Instruction is used to provide notification to operators of abnormal conditions or events for up to 32 additional items external to a parent object. For more information, see the Rockwell Automation Library of Process Objects, publication PROCES-RM200.

This instruction handles the connections of the commands from the parent object:

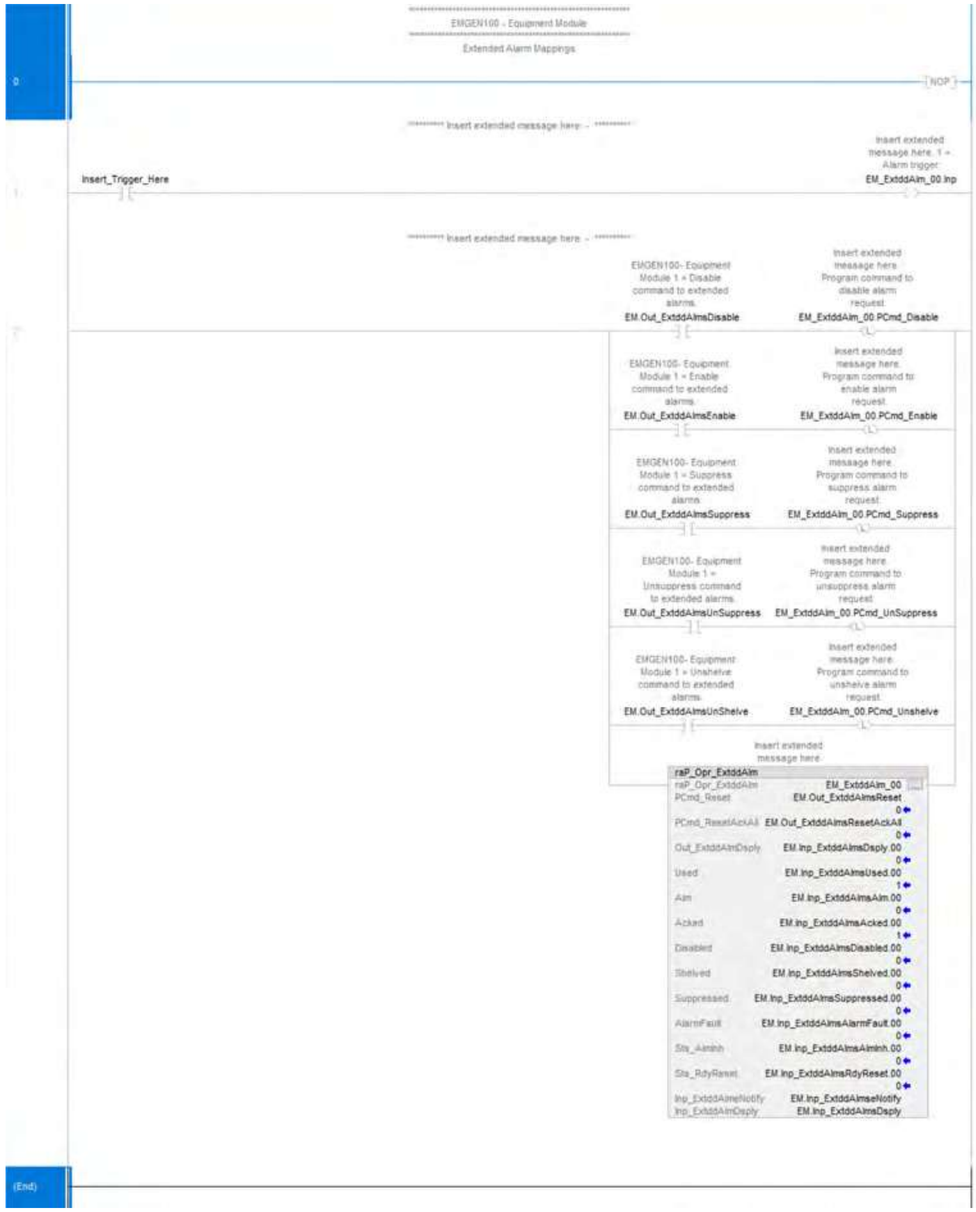
- Acknowledge
- Reset
- Enabling/Disabling
- Suppress/Unsuppress
- UnShelve

This instruction handles the connections of the status from the raP_Opr_ExtddAlm:

- Used
- Alarm
- Acknowledged
- Disabled
- Suppressed
- Shelved
- Alarm Fault
- Ready for Reset
- Notify value

raP_Opr_ExtddAlm Parameters

Parameter	Description
PCmd_Reset	Program command to reset alarm request.
PCmd_ResetAckAll	Program command to reset and acknowledges all alarms. The instruction clears this operand automatically.
Out_ExtddAlmDsply	1 = Extended alarm severity value is greatest of all extended alarms active. Each bit represents an individual alarm (0…31).
Used	1 = Used.
Alm	1 = Alarm is active.
Acked	1 = In alarm acknowledged.
Disabled	1 = Alarm disabled.
Shelved	1 = Alarm shelved.
Suppressed	1 = Alarm suppressed.
AlarmFault	1 = Alarm fault.
Sts_AlmInh	1 = One or more alarms shelved, disabled, or suppressed.
Sts_Rdy_Reset	1 = A latched alarm condition is ready to be reset.
Inp_ExtddAlmeNotify	Extended alarms status enumerate values: 0 = Not in alarm 1 = Not in alarm unacknowledged or reset requires 2 = Low severity alarm acknowledged 3 = Low severity alarm unacknowledged 4 = Medium severity alarm acknowledged 5 = Medium severity alarm unacknowledged 6 = High severity alarm acknowledged 7 = High severity alarm unacknowledged 8 = Urgent severity alarm acknowledged 9 = Urgent severity alarm unacknowledged
Inp_ExtddAlmDsply	1 = Extended alarm severity value is greatest of all extended alarms active. Each bit represents an individual alarm (0…31).


EMGEN100 - Equipment Module
Extended Alarm Mappings
NOP
Insert extended message here
Insert_Trigger_Here
EM_ExtddAlm_00.Inp
EM.Out_ExtddAlmsDisable
EM_ExtddAlm_00.PCmd_Disable
EM.Out_ExtddAlmsEnable
EM_ExtddAlm_00.PCmd_Enable
EM.Out_ExtddAlmsSuppress
EM_ExtddAlm_00.PCmd_Suppress
EM.Out_ExtddAlmsUnSuppress
EM_ExtddAlm_00.PCmd_UnSuppress
EM.Out_ExtddAlmsUnShelve
EM_ExtddAlm_00.PCmd_Unshelve
raP_Opr_ExtddAlm
EM_ExtddAlm_00
EM.Out_ExtddAlmsReset
EM.Out_ExtddAlmsResetAckAll
EM.Inp_ExtddAlmsDsply.00
EM.Inp_ExtddAlmsUsed.00
EM.Inp_ExtddAlmsAlm.00
EM.Inp_ExtddAlmsAcked.00
EM.Inp_ExtddAlmsDisabled.00
EM.Inp_ExtddAlmsShelved.00
EM.Inp_ExtddAlmsSuppressed.00
EM.Inp_ExtddAlmsAlarmFault.00
EM.Inp_ExtddAlmsAlmInh.00
EM.Inp_ExtddAlmsRdyReset.00
EM.Inp_ExtddAlmseNotify
EM.Inp_ExtddAlmsDsply
(End)

Parameters and Reports Routines

The raP_Tec_ParRpt Add-On Instruction is used to implement parameter data items. Use when:

- You need the ability to view or modify a parameter from either the HMI or from logic.
- You need to arbitrate parameter input based on mode.
- You need the ability to limit the value of a parameter, from either the HMI or logic.
- You need the ability to capture an initial parameter value (based on a trigger), and provide an indication if the parameter was adjusted from the initial value.
- You need to limit the adjustment of a parameter within a deadband relative to an initial value.
- You need to apply command confirmation (i.e. Electronic Signature) to parameter entry from the HMI.
- Your parameter is read only or read/write.
- Your need a Parameter (recipe) or Report (resultant) parameter.
- Your parameter is of data type: Integer, Real, String, or is an Enumeration.

For more information, see the Rockwell Automation Library of Process Objects, publication PROCES-RM200.

raP_Tec_ParRpt References

Parameter	Description
PSet_E	Program issued setting of enumeration parameter value.
PSet_I	Program issued setting of integer parameter value.
PSet_R	Program issued setting of real parameter value.
PSet_S	Program issued setting of string parameter value.

raP_Tec_ParRpt Configuration Considerations

Operand	Type	Description
ParObj	raP_UDT_Opr_ParRpt_INTfC	Parameter object link to equipment
RptObj	raP_UDT_Opr_ParRpt_INTfC	Report object link to equipment

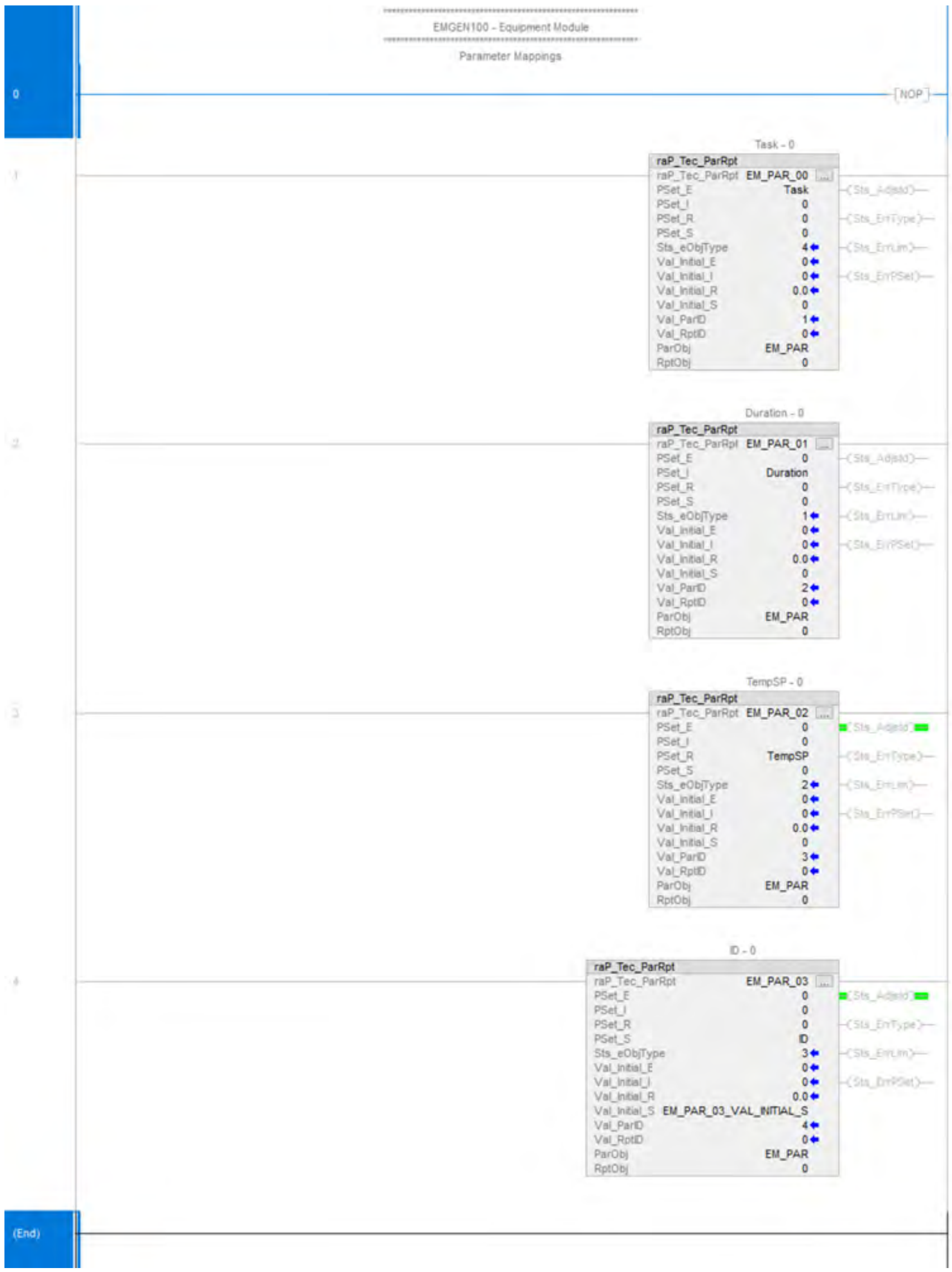
IMPORTANT You cannot set both ParObj and RptObj in the same Add-On Instruction.
- If you set ParObj, then RptObj must be zero.
- If you set RptObj, then ParObj must be zero.

CS_raP_Opr_EMGen Parameters Routine

Maps Parameters from tags (input) to the standard EM_GEN parameter structure. The steps required to create this mapping logic are as follows:

1. First determine the parameters needed for your Equipment Module, and ensure the associated tags (input) are defined within your program.
2. Open the CS_raP_Opr_EMGen Parameters Routine.
3. Starting at parameter zero (EM_Par_00), determine the type of parameter block required (Enumeration, Integer, Real, or String).
4. Modify the rest of the reports routine as needed.

Typically the PSet variable in the instruction would represent the parameter tag (as defined in the program tags). For each additional Parameter required, increment the EM_Par_## number. Up to a maximum of 48 reports can be included per EM_GEN.



CS_raP_Opr_EMGen Reports Routine

Maps Resultant/Report data to output tags from the standard EM_GEN report structure. The steps required to create this mapping logic are as follows:

1. First determine the report data items needed for your Equipment Module, and make sure the associated tags (output) are defined within your program.
2. Open the CS_raP_Opr_EMGen Reports Routine.
3. Starting at report zero (EM_RPT_00), determine the type of parameter block required (Enumeration, Integer, Real, or String).
4. Modify the rest of the reports routine as needed.

Typically the PSet variable in the instruction would represent the parameter tag (as defined in the program tags). For each additional Parameter required increment the EM_RPT_0## number. Up to a maximum of 48 reports can be included per EM_GEN.

Notes:

Generic Equipment Phase (EPGEN) Control Strategy

An equipment phase is a functional group of equipment that can conduct a finite number of specific minor processing activities when directed by a phase (recipe).

The CS_raP_Opr_EPGen_PHASE control strategy controls an Equipment Phase in various modes and monitors for fault conditions.

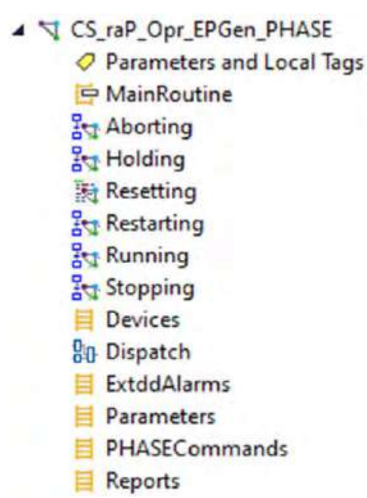
Use when:

- You want to group equipment, and you want to apply the ISA 88.01 state model using PhaseManager™.
- You want to provide the following for a group of equipment.
  - Apply a mode model to the equipment group.
  - Apply interlocks and/or permissives to the group of equipment.
  - Parameters that define the behavior of the group of equipment.
  - Report resultant data from the group of equipment.
  - A faceplate that allows monitoring and control of the equipment grouping.
  - Monitor step (description), and allow forcing of steps in maintenance mode.
  - Allow alarms to be defined for certain process / equipment failure conditions.
  - Alarming function, including alarms based on device failure.

Do NOT use when:

- You must sequence or coordinate a device, and do not require any of the above.
- You want to apply a custom state model to the equipment, use the CS_raP_Opr_EMGen control strategy instead.
- You want to apply the PackML state model.

The EPGEN control strategy is available as a program in the process library:



Import the appropriate control strategy as a **program** in your controller project.

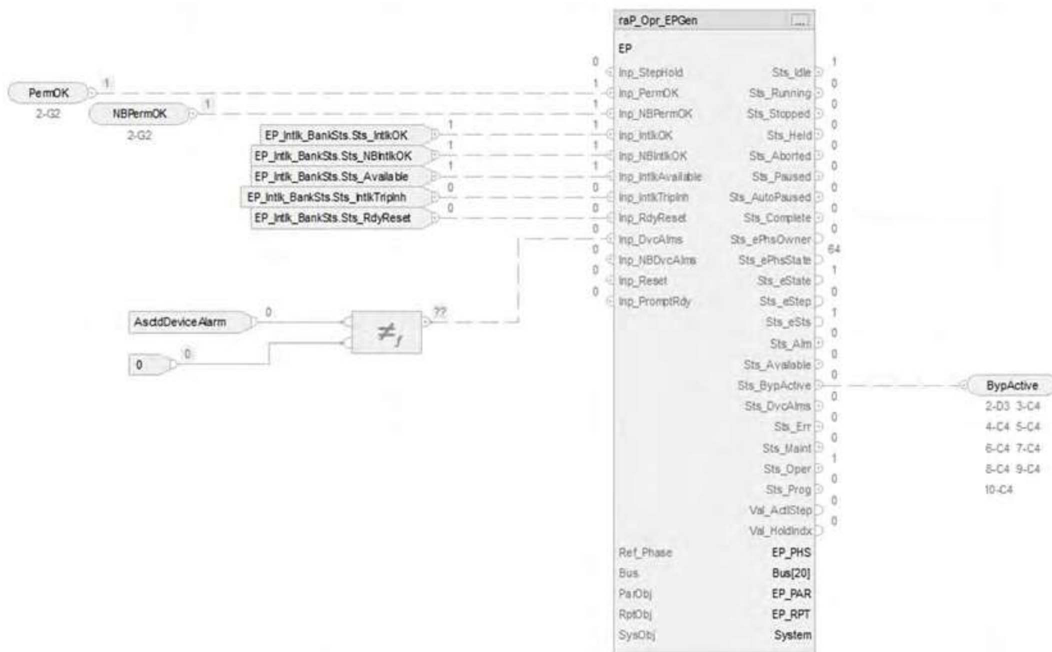
Routine	Description
Aborting	Used for shutting down equipment in an emergency situation. If you have implemented Stopping, you would at a minimum duplicate the stopping logic within Aborting. In some In some cases, the sequence in an emergency situation (Aborting) differs from the orderly shutdown of equipment (Stopping). Add logic appropriate to your application.
Holding	Used if equipment or a subset of equipment must be shut down when the phase enters the hold state. It can also be advantageous to release owned equipment if maintaining ownership while held constrains production by maintaining ownership of shared equipment. Add logic appropriate to your application.
Resetting	Used to perform "clean-up" activities such as release owned equipment. Add logic appropriate to your application.
Restarting	Generally implemented if Holding is implemented. Used to bring equipment from the state that it is in at the end of the Holding state back to the state it was in prior Holding. Add logic appropriate to your application.
Running	Use to start up equipment, and acquire ownership of equipment (if necessary). Add logic appropriate to your application.
Stopping	Use if equipment must be shut down in a given sequence.
Devices	Status of devices. Add logic appropriate to your application.
Dispatch	Contains the raP_Opr_EPGen Add-On Instruction.
ExtddAlarms	Contains instances of external alarms and trigger logic.
Parameters	Equipment Phase Parameters Routine - EP parameter mapping and logic
PHASECommands	Maps commands from EPGEN to PhaseManager commands
Report	Equipment Phase Reports Routine - EP Report mapping and logic

Dispatch Routine

The Dispatch routine contains these Function Block sheets:

Sheet	Description
EPGEN100	Equipment Phase Add-On Instruction
Permissives	Process Permissives instruction The Process Permissives (PPERM) instruction collects, or sums up, the permissive conditions that let a piece of equipment energize. In most cases, permissive conditions must be true to energize equipment. Once the equipment is energized, permissives are ignored.
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The instruction monitors by passable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder

Dispatch EPGEN100 Sheet



raP_Opr_EPGen Input References

Parameter	Description
PermOK	Input connection from Permissives sheet 0 (State 1) 1 = On permissives OK, device can turn On
NBPermOK	Input connection from Permissives sheet 0 (State 1) 1 = Non-bypassable On permissives OK, device can turn On
EP_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
EP_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
EP_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
EP_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
EP_Intlk_BankSts.Sts_RdyReset	Interlock bank status 1 = A latched interlock (returned to OK) is ready to be reset
AsctdDeviceAlarm	Associated Device Alarm Active if any Bits are Logic 1.

raP_Opr_EPGen Output References

Parameter	Description
BypActive	Output connection to permissives and interlock bank sheets

raP_Opr_EPGen Configuration Considerations

Operand	Type	Description
PlantPAx® control	raP_Opr_EPGen	Instance of data structure (backing tag) required for proper operation of instruction
Ref_Phase	PHASE	Referenced phase.
Bus	raP_UDT_Opr_Bus	Bus component
ParObj	raP_UDT_Opr_ParRpt_Intfc	Optional parameter object interface
RptObj	raP_UDT_Opr_ParRpt_Intfc	Optional report object interface
SysObj	raP_UDT_Opr_System	System component.

Dispatch Permissive Sheet

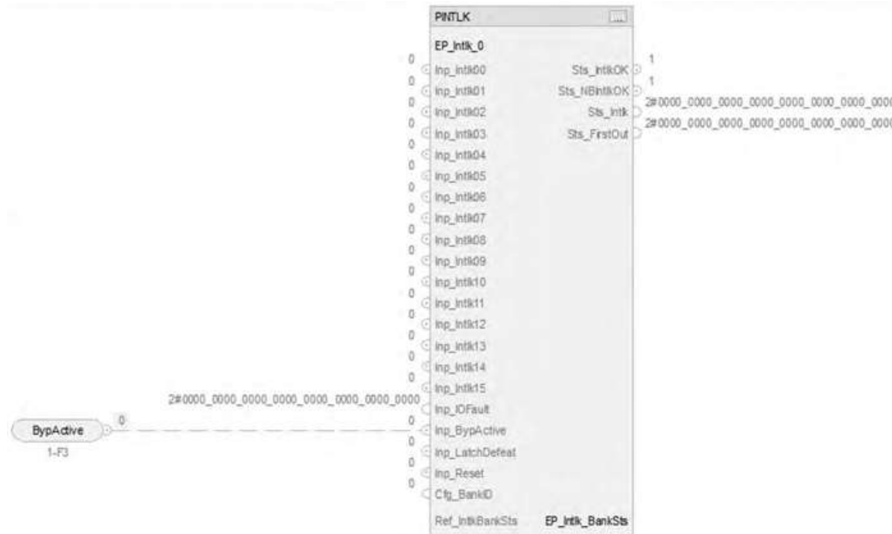
PPERM Input References

Parameter	Description
BypActive	Input connection from the EPGEN100 sheet

PPERM Output References

Parameter	Description
PermOK	Overall permissive status (1 = OK to energize)
NBPermOK	Non-bypassable permissive status (1 = all non-bypassable permissives OK to energize)

Dispatch Interlock Bank Sheet



PINTLK Input Reference

Parameter	Description
BypActive	Input connection from the CS_D4SD sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

PHASECommands Routine

Maps commands from the EP_GEN instance to PhaseManager commands.

Steps required to map EP_GEN Phase Commands in Ladder:

1. Open the CS_raP_Opr_EPGen PhaseCommands Routine.
2. Modify the selected phase requests as required. EP_GEN Phase request interface points are as follows:

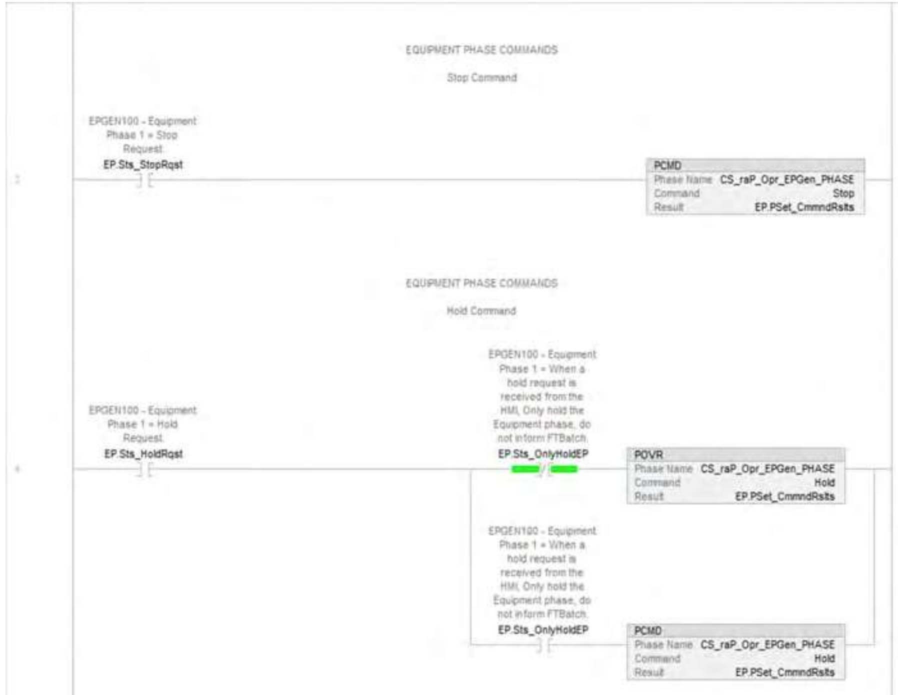
- Sts_StartRqst
- Sts_HoldRqst
- Sts_RestartRqst
- Sts_StopRqst
- Sts_AbortRqst
- Sts_ResetRqst
- Sts_PauseRqst
- Sts_ResumeRqst
- Sts_StateCmpltRqst

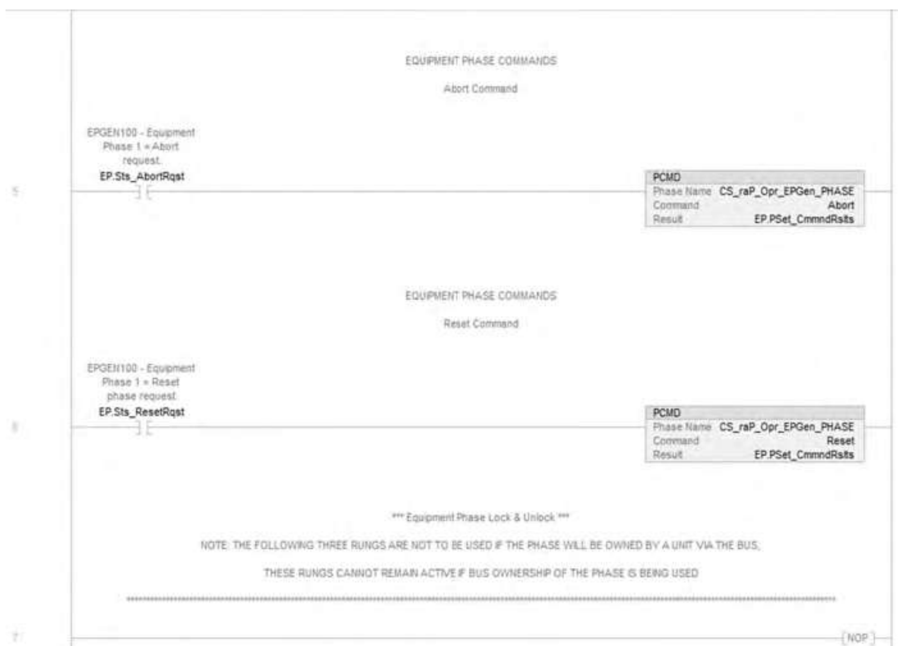
3. EP_GEN Request interface points are defined so that a 1 = Requested.

- Typically a command rung would contain an XIC that represents the EP_GEN phase request (.Sts_<state>Rqst), and a PCMD or POVR instruction to issue the corresponding command to the PhaseManager Phase
- Phase Commands (PCMD) exist for Start, ReStart, Reset, Pause, and Resume states.
- Phase Override Commands (POVR) exist for Stop, Hold, and ABORT states.
- A Phase Command required for each Phase State routine you have defined within your PhaseManager Phase.

4. The PCMD and POVR require definition of several reference tags:

- Phase Name <tag>_Phase
- Command <phase command>
- Result <tag>_PSet_CmmndRslts


EQUIPMENT PHASE COMMANDS
Stop Command
EPGEN100 - Equipment Phase 1 = Stop Request
EP.Sts_StopRqst
PCMD
Phase Name CS_raP_Opr_EPGen_PHASE
Command Stop
Result EP.PSet_CmmndRslts
EQUIPMENT PHASE COMMANDS
Hold Command
EPGEN100 - Equipment Phase 1 = Hold Request
EP.Sts_HoldRqst
EPGEN100 - Equipment Phase 1 = When a hold request is received from the HMI, Only hold the Equipment phase, do not inform FTBatch.
EP.Sts_OnlyHoldEP
POVR
Phase Name CS_raP_Opr_EPGen_PHASE
Command Hold
Result EP.PSet_CmmndRslts
EPGEN100 - Equipment Phase 1 = When a hold request is received from the HMI, Only hold the Equipment phase, do not inform FTBatch.
EP.Sts_OnlyHoldEP
PCMD
Phase Name CS_raP_Opr_EPGen_PHASE
Command Hold
Result EP.PSet_CmmndRslts


EQUIPMENT PHASE COMMANDS
Abort Command
EPGEN100 - Equipment Phase 1 = Abort request.
EP.Sts_AbortRqst
PCMD
Phase Name CS_raP_Opr_EPGen_PHASE
Command Abort
Result EP.PSet_CmmndRslts
EQUIPMENT PHASE COMMANDS
Reset Command
EPGEN100 - Equipment Phase 1 = Reset phase request.
EP.Sts_ResetRqst
PCMD
Phase Name CS_raP_Opr_EPGen_PHASE
Command Reset
Result EP.PSet_CmmndRslts
*** Equipment Phase Lock & Unlock ***
NOTE: THE FOLLOWING THREE RUNGS ARE NOT TO BE USED IF THE PHASE WILL BE OWNED BY A UNIT VIA THE BUS.
THESE RUNGS CANNOT REMAIN ACTIVE IF BUS OWNERSHIP OF THE PHASE IS BEING USED
NOP


*** Equipment Phase Lock & Unlock ***
When in Operator Control mode an Equipment Phase can be manually controlled on the HMI by an operator, an operator can also Lock an Equipment Phase in Operator Control mode
When in Program Control mode an Equipment Phase can be Owned by an External Sequencer (Batch), an Internal Sequencer or the Phase Monitor Logix
1. When the Equipment Phase is Not Locked in Program/Operator Control mode and is Owned, then Lock the Equipment Phase in Program Control mode
2. When the Equipment Phase is Locked in Program Control mode and is no longer Owned, then Unlock the Equipment Phase
3. After Unlocking the Equipment Phase, set the control mode to the configured 'Normal' control mode once
*** 1. Command to Lock ***
EP.Sts_ProgOperLock
EP.Sts_PhsOwnerExtlSeq
EP.Sts_PhsOwnerIntlSeq
EP.Sts_PhsOwnerLogix
EP.PCmd_Lock
***2. Command to Unlock ***
EP.Sts_ProgOperLock
EP.Sts_Prog
EP.Sts_PhsOwnerExtlSeq
EP.Sts_PhsOwnerIntlSeq
EP.Sts_PhsOwnerLogix
EP.PCmd_Unlock
EP_SetNormal
*** 3. Command to select configured 'Normal' control mode ***
EP.Sts_ProgOperLock
EP_SetNormal
EP.Cfg_ProgNormal
EP.PCmd_Oper
EP_SetNormal
EP.Cfg_ProgNormal
EP.PCmd_Prog
(End)

Extended Alarms Routine

The raP_Opr_ExtddAlm (Extended Alarm Block) Add-On Instruction is used to provide notification to operators of abnormal conditions or events for up to 32 additional items external to a parent object. For more information, see the Rockwell Automation Library of Process Objects, publication PROCES-RM200.

This instruction handles the connections of the commands from the parent object:

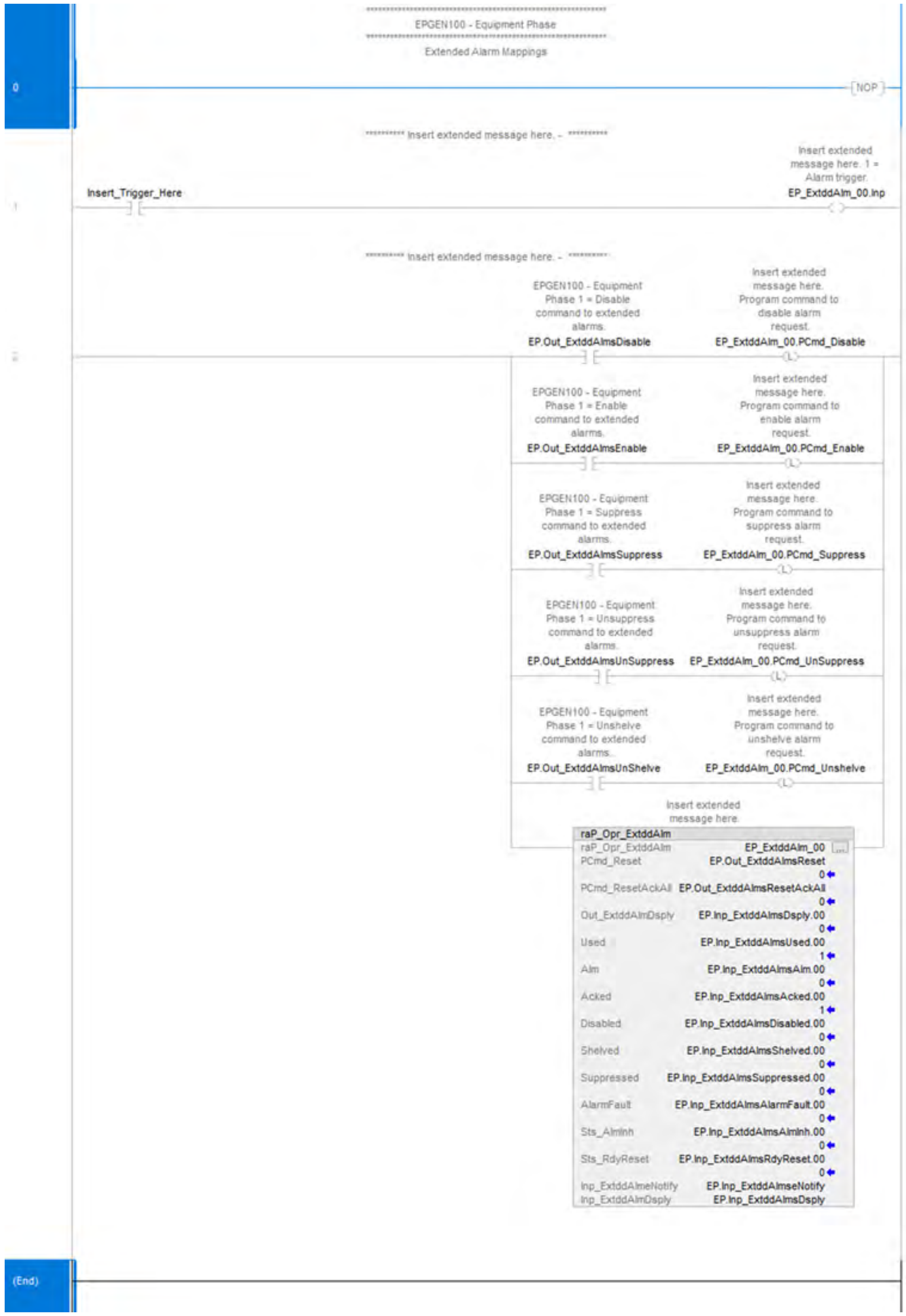
- Acknowledge
- Reset
- Enabling/Disabling
- Suppress/Unsuppress
- UnShelve

This instruction handles the connections of the status from the raP_Opr_ExtddAlm:

- Used
- Alarm
- Acknowledged
- Disabled
- Suppressed
- Shelved
- Alarm Fault
- Ready for Reset
- Notify value

raP_Opr_ExtddAlm Parameters

Parameter	Description
PCmd_Reset	Program command to reset alarm request.
PCmd_ResetAckAll	Program command to reset and acknowledges all alarms. The instruction clears this operand automatically.
Out_ExtddAlmDsply	1 = Extended alarm severity value is greatest of all extended alarms active. Each bit represents an individual alarm (0…31).
Used	1 = Used.
Alm	1 = Alarm is active.
Acked	1 = In alarm acknowledged.
Disabled	1 = Alarm disabled.
Shelved	1 = Alarm shelved.
Suppressed	1 = Alarm suppressed.
AlarmFault	1 = Alarm fault.
Sts_AlmInh	1 = One or more alarms shelved, disabled, or suppressed.
Sts_Rdy_Reset	1 = A latched alarm condition is ready to be reset.
Inp_ExtddAlmeNotify	Extended alarms status enumerate values: 0 = Not in alarm 1 = Not in alarm unacknowledged or reset requires 2 = Low severity alarm acknowledged 3 = Low severity alarm unacknowledged 4 = Medium severity alarm acknowledged 5 = Medium severity alarm unacknowledged 6 = High severity alarm acknowledged 7 = High severity alarm unacknowledged 8 = Urgent severity alarm acknowledged 9 = Urgent severity alarm unacknowledged
Inp_ExtddAlmDsply	1 = Extended alarm severity value is greatest of all extended alarms active. Each bit represents an individual alarm (0…31).


EPGEN100 - Equipment Phase
Extended Alarm Mappings
NOP
Insert extended message here.
Insert_Trigger_Here
Insert extended message here. 1 = Alarm trigger.
EP_ExtddAlm_00.Inp
EPGEN100 - Equipment Phase 1 = Disable command to extended alarms.
EP.Out_ExtddAlmsDisable
Insert extended message here. Program command to disable alarm request.
EP_ExtddAlm_00.PCmd_Disable
EPGEN100 - Equipment Phase 1 = Enable command to extended alarms.
EP.Out_ExtddAlmsEnable
Insert extended message here. Program command to enable alarm request.
EP_ExtddAlm_00.PCmd_Enable
EPGEN100 - Equipment Phase 1 = Suppress command to extended alarms.
EP.Out_ExtddAlmsSuppress
Insert extended message here. Program command to suppress alarm request.
EP_ExtddAlm_00.PCmd_Suppress
EPGEN100 - Equipment Phase 1 = Unsuppress command to extended alarms.
EP.Out_ExtddAlmsUnSuppress
Insert extended message here. Program command to unsuppress alarm request.
EP_ExtddAlm_00.PCmd_UnSuppress
EPGEN100 - Equipment Phase 1 = Unshelve command to extended alarms.
EP.Out_ExtddAlmsUnShelve
Insert extended message here. Program command to unshelve alarm request.
EP_ExtddAlm_00.PCmd_Unshelve
Insert extended message here.
raP_Opr_ExtddAlm
raP_Opr_ExtddAlm EP_ExtddAlm_00
PCmd_Reset EP.Out_ExtddAlmsReset
PCmd_ResetAckAll EP.Out_ExtddAlmsResetAckAll
Out_ExtddAlmDsply EP.Inp_ExtddAlmsDsply.00
Used EP.Inp_ExtddAlmsUsed.00
Alm EP.Inp_ExtddAlmsAlm.00
Acked EP.Inp_ExtddAlmsAcked.00
Disabled EP.Inp_ExtddAlmsDisabled.00
Shelved EP.Inp_ExtddAlmsShelved.00
Suppressed EP.Inp_ExtddAlmsSuppressed.00
AlarmFault EP.Inp_ExtddAlmsAlarmFault.00
Sts_AlmInh EP.Inp_ExtddAlmsAlmInh.00
Sts_RdyReset EP.Inp_ExtddAlmsRdyReset.00
Inp_ExtddAlmeNotify EP.Inp_ExtddAlmseNotify
Inp_ExtddAlmDsply EP.Inp_ExtddAlmsDsply
(End)

Parameters and Reports Routines

The raP_Tec_ParRpt Add-On Instruction is used to implement parameter data items. Use when:

- You need the ability to view or modify a parameter from either the HMI or from logic.
- You must arbitrate parameter input based on mode.
- You need the ability to limit the value of a parameter, from either the HMI or logic.
- You need the ability to capture an initial parameter value (based on a trigger), and provide an indication if the parameter was adjusted from the initial value.
- You must limit the adjustment of a parameter within a deadband relative to an initial value.
- You must apply command confirmation (that is, Electronic Signature) to parameter entry from the HMI.
- Your parameter is read-only or read/write.
- Your need a Parameter (recipe) or Report (resultant) parameter.
- Your parameter is of data type: Integer, Real, String, or is an Enumeration.

For more information, see the Rockwell Automation Library of Process Objects, publication PROCES-RM200.

raP_Tec_ParRpt References

Parameter	Description
PSet_E	Program issued setting of enumeration parameter value.
PSet_I	Program issued setting of integer parameter value.
PSet_R	Program issued setting of real parameter value.
PSet_S	Program issued setting of string parameter value.

raP_Tec_ParRpt Configuration Considerations

Operand	Type	Description
ParObj	raP_UDT_Opr_ParRpt_INTfC	Parameter object link to equipment
RptObj	raP_UDT_Opr_ParRpt_INTfC	Report object link to equipment

IMPORTANT You cannot set both ParObj and RptObj in the same Add-On Instruction.
- If you set ParObj, then RptObj must be zero.
- If you set RptObj, then ParObj must be zero.

CS_raP_Opr_EPGen Parameters Routine

Maps Parameters from Phase tags (input) to the standard EP_GEN parameter structure. The steps required to create this mapping logic are as follows:

1. First determine the parameters needed for your Equipment Phase, and confirm the associated tags (input) are defined within your PhaseManager program.
2. Open the CS_raP_Opr_EPGen Parameters Routine.
3. Start at parameter zero (EP_Par_00), and determine the type of parameter block required (Enumeration, Integer, Real, or String).
4. Modify the rest of the reports routine as needed.

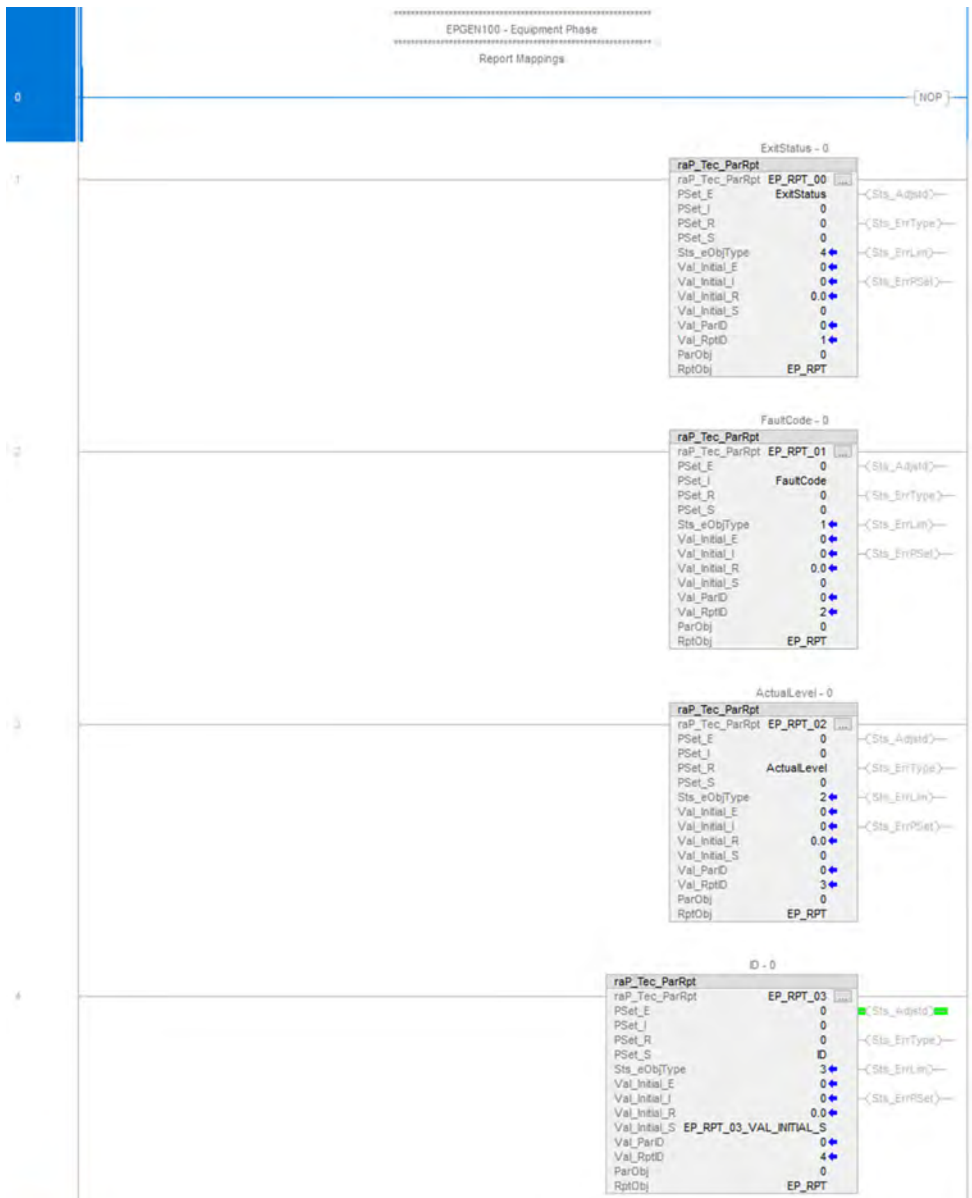
Typically the PSet variable in the instruction would represent the parameter tag (as defined in the program tags). For each additional Parameter required, increment the EP_Par_## number. Up to a maximum of 48 reports can be included per EP_GEN.

CS_raP_Opr_EPGen Reports Routine

Maps Resultant/Report data to Phase tags (output) from the standard EP_GEN report structure. The steps required to create this mapping logic are as follows:

1. First determine the report data items needed for your Equipment Phase, and confirm the associated tags (output) are defined within your PhaseManager program.
2. Open the CS_raP_Opr_EPGen Reports Routine.
3. Start at report zero (EP_RPT_00), and determine the type of parameter block required (Enumeration, Integer, Real, or String).
4. Modify the rest of the reports routine as needed.

Typically the PSet variable in the instruction would represent the parameter tag (as defined in the program tags). For each additional Parameter required increment the EP_RPT_0## number. Up to a maximum of 48 reports can be included per EP_GEN.



Notes:

Process Unit Control Strategy

The Process Unit control strategy groups equipment together, and provides a propagation mechanism for aggregating status from equipment, and broadcasting commands to equipment. For example, each vessel, tank, mixer, machine within the control system would be considered a Unit. You can consolidate the status from groups of equipment, and display the consolidated status on an HMI.

These status items include:

- Alarm Status
- Alarm Severity
- Mode
- Configuration Errors
- Prompt Status

You can also manage any of the following functions for a group of equipment with a global set of commands:

- Mode
- Alarm Acknowledge
- Alarm Reset

The CS_raP_Opr_Unit control strategy is available as two routines in the process library. Import the appropriate control strategy as a **routine** in your controller project.

Routine	Description
UNIT100	Function Block control strategy routine
ExtddAlarms	Contains instances of external alarms and trigger logic.

The Unit100 routine contains these Function Block sheets:

Sheet	Description
raP_Opr_Unit	Process Unit Add-On Instruction
Permissives	Process Permissives instruction The Process Permissives (PPERM) instruction collects, or sums up, the permissive conditions that let a piece of equipment energize. In most cases, permissive conditions must be true to energize equipment. Once the equipment is energized, permissives are ignored.
Interlock Bank 0 Interlock Bank 1 Interlock Bank 2 Interlock Bank 3 Interlock Bank 4 Interlock Bank 5 Interlock Bank 6 Interlock Bank 7	The instruction monitors bypassable and non-bypassable Interlocks that force the analog output to a specific configured (safe) value or to maintain the current value (configurable). There are 8 interlock bank sheets; each sheet exposes 16 of the available 32 interlocks per bank by default. Use those sheets and interlocks that you need; delete the remainder.

CS_raOpr_Unit Sheet

The Unit100 routine contains the CS_raP_Opr_Unit sheet.

raP_Opr_Unit Input References

Parameter	Description
Unit100_Inp_EStop	1 = Emergency stop input ok.
Unit100_Inp_SStop	1 = Software stop input ok.
PermOK	Input connection from Permissives sheet 0 (State 1) 1 = On permissives OK, device can turn On
NBPermOK	Input connection from Permissives sheet 0 (State 1) 1 = Non-bypassable On permissives OK, device can turn On
Unit100_Intlk_BankSts.Sts_IntlkOK	Interlock bank status 1 = OK to run 0 = Stop
Unit100_Intlk_BankSts.Sts_NBIntlkOK	Interlock bank status 1 = All non-bypassable interlocks OK to run
Unit100_Intlk_BankSts.Sts_Available	Interlock bank status 1 = Available
Unit100_Intlk_BankSts.Sts_IntlkTripInh	Interlock bank status 1 = Interlock trip inhibit - stops equipment but does not trip
Unit100_Intlk_BankSts.Sts_RdyReset	Interlock bank status 1 = A latched interlock (returned to OK) is ready to be reset

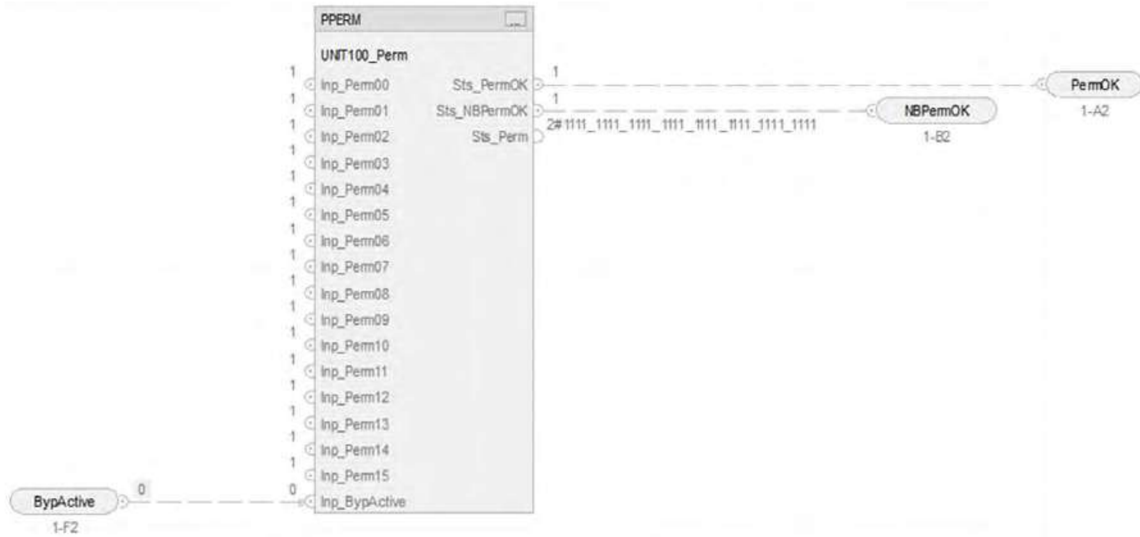
raP_Opr_Unit Output References

Parameter	Description
BypActive	Output connection to permissives and interlock bank sheets

raP_Opr_Unit Configuration Considerations

Operand	Type	Description
PlantPAx® control	raP_Opr_Unit	Instance of data structure (backing tag) required for proper operation of instruction
Bus	raP_UDT_Opr_Bus	Bus component for organization control. 0 if not using organization Bus[x].Obj when using organization See the Rockwell Automation Library of Process Objects Reference Manual, publication PROCES-RM200.
BatchIntfc	raP_UDT_Opr_PUnitFTBatch_Intfc	FactoryTalk Batch Interface
ParObj	raP_UDT_Opr_ParRpt_Intfc	Parameter object interface
RptObj	raP_UDT_Opr_ParRpt_Intfc	Report object interface

Permissive Sheet



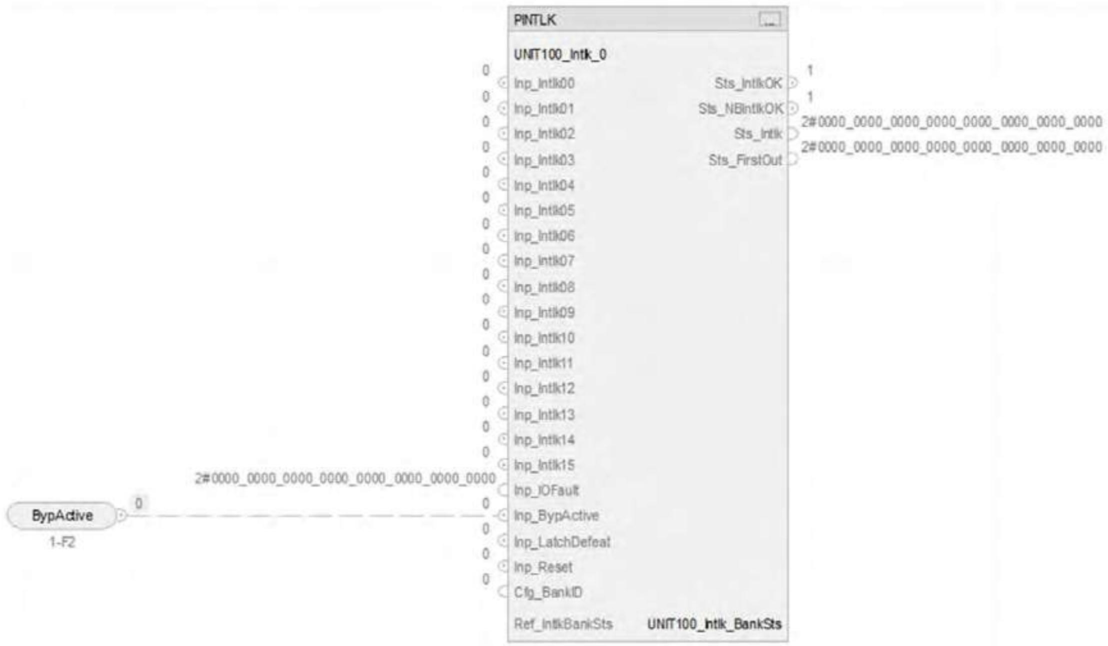
PPERM Input References

Parameter	Description
BypActive	Input connection from the raP_Opr_Unit sheet

PPERM Output References

Parameter	Description
PermOK	Overall permissive status (1 = OK to energize)
NBPermOK	Non-bypassable permissive status (1 = all non-bypassable permissives OK to energize)

Interlock Bank Sheet



PINTLK Input Reference

Parameter	Description
BypActive	Input connection from the raP_Opr_Unit sheet

PINTLK Configuration Considerations

Operand	Type	Description
PlantPAx® control	P_INTERLOCK	Instance of data structure (backing tag) required for proper operation of instruction DS4D100 in this example corresponds to a state device
Ref_IntlkBankSts	P_INTERLOCK_BANK_STATUS	Reference interlock bank status

Extended Alarms Routine

The raP_Opr_ExtddAlm (Extended Alarm Block) Add-On Instruction is used to provide notification to operators of abnormal conditions or events for up to 32 additional items external to a parent object. For more information, see the Rockwell Automation Library of Process Objects, publication PROCES-RM200.

This instruction handles the connections of the commands from the parent object:
- Acknowledge
- Reset
- Enabling/Disabling
- Suppress/Unsuppress
- UnShelve

This instruction handles the connections of the status from the raP_Opr_ExtddAlm:
- Used
- Alarm
- Acknowledged
- Disabled
- Suppressed
- Shelved
- Alarm Fault
- Ready for Reset
- Notify value

raP_Opr_ExtddAlm Parameters

Parameter	Description
PCmd_Reset	Program command to reset alarm request.
PCmd_ResetAckAll	Program command to reset and acknowledges all alarms. The instruction clears this operand automatically.
Out_ExtddAlmDsply	1 = Extended alarm severity value is greatest of all extended alarms active. Each bit represents an individual alarm (0…31).
Used	1 = Used.
Alm	1 = Alarm is active.
Acked	1 = In alarm acknowledged.
Disabled	1 = Alarm disabled.
Shelved	1 = Alarm shelved.
Suppressed	1 = Alarm suppressed.
AlarmFault	1 = Alarm fault.
Sts_AlmInh	1 = One or more alarms shelved, disabled, or suppressed.
Sts_Rdy_Reset	1 = A latched alarm condition is ready to be reset.
Inp_ExtddAlmeNotify	Extended alarms status enumerate values: 0 = Not in alarm 1 = Not in alarm unacknowledged or reset requires 2 = Low severity alarm acknowledged 3 = Low severity alarm unacknowledged 4 = Medium severity alarm acknowledged 5 = Medium severity alarm unacknowledged 6 = High severity alarm acknowledged 7 = High severity alarm unacknowledged 8 = Urgent severity alarm acknowledged 9 = Urgent severity alarm unacknowledged
Inp_ExtddAlmDsply	1 = Extended alarm severity value is greatest of all extended alarms active. Each bit represents an individual alarm (0…31).


UNIT100 - Unit
Extended Alarms Mappings
NOP
Insert extended message here
Insert extended message here 1 = Alarm trigger
Insert_Trigger_Here
UNIT100_ExtddAlm_00.Inp
Insert extended message here
Insert extended message here
raP_Opr_ExtddAlm
raP_Opr_ExtddAlm UNIT100_ExtddAlm_00
PCmd_Reset UNIT100.Out_ExtddAlmsReset 0
PCmd_ResetAckAll UNIT100.Out_ExtddAlmsResetAckAll 0
Out_ExtddAlmDsply UNIT100.Inp_ExtddAlmsDsply.00 0
Used UNIT100.Inp_ExtddAlmsUsed.00 1
Alm UNIT100.Inp_ExtddAlmsAlm.00 0
Acked UNIT100.Inp_ExtddAlmsAcked.00 1
Disabled UNIT100.Inp_ExtddAlmsDisabled.00 0
Shelved UNIT100.Inp_ExtddAlmsShelved.00 0
Suppressed UNIT100.Inp_ExtddAlmsSuppressed.00 0
AlarmFault UNIT100.Inp_ExtddAlmsAlarmFault.00 0
Sts_AlmInh UNIT100.Inp_ExtddAlmsAlmInh.00 0
Sts_RdyReset UNIT100.Inp_ExtddAlmsRdyReset.00 0
Inp_ExtddAlmeNotify UNIT100.Inp_ExtddAlmseNotify
Inp_ExtddAlmDsply UNIT100.Inp_ExtddAlmsDsply
(End)

Rockwell Automation Support

Use these resources to access support information.

Technical Support Center	Find help with how-to videos, FAQs, chat, user forums, Knowledgebase, and product notification updates.	rok.auto/support
Local Technical Support Phone Numbers	Locate the telephone number for your country.	rok.auto/phonesupport
Technical Documentation Center	Quickly access and download technical specifications, installation instructions, and user manuals.	rok.auto/techdocs
Literature Library	Find installation instructions, manuals, brochures, and technical data publications.	rok.auto/literature
Product Compatibility and Download Center (PCDC)	Download firmware, associated files (such as AOP, EDS, and DTM), and access product release notes.	rok.auto/pcdc

Documentation Feedback

Your comments help us serve your documentation needs better. If you have any suggestions on how to improve our content, complete the form at rok.auto/docfeedback.

Waste Electrical and Electronic Equipment (WEEE)

At the end of life, this equipment should be collected separately from any unsorted municipal waste.

Rockwell Automation maintains current product environmental compliance information on its website at rok.auto/pec.

Rockwell Otomasyon Ticaret A.Ş. Kar Plaza İş Merkezi E Blok Kat:6 34752, İçerenköy, İstanbul, Tel: +90 (216) 5698400 EEE Yönetmeliğine Uygundur

Connect with us.

rockwellautomation.com expanding **human possibility**®

AMERICAS: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
EUROPE/MIDDLE EAST/AFRICA: Rockwell Automation NV, Pegasus Park, De Kleetlaan 12a, 1831 Diegem, Belgium, Tel: (32) 2663 0600, Fax: (32) 2 663 0640
ASIA PACIFIC: Rockwell Automation SEA Pte Ltd, 2 Corporation Road, #04-05, Main Lobby, Corporation Place, Singapore 618494, Tel: (65) 6510 6608, FAX: (65) 6510 6699
UNITED KINGDOM: Rockwell Automation Ltd., Pitfield, Kiln Farm, Milton Keynes, MK11 3DR, United Kingdom, Tel: (44)(1908) 838-800, Fax: (44)(1908) 261-917

Publication PROCES-RM201A-EN-P - January 2024