



Process Expert - General Purpose Library Classic

Process Templates Reference Manual

EIO0000000987.17
03/2023



Legal Information

The Schneider Electric brand and any trademarks of Schneider Electric SE and its subsidiaries referred to in this guide are the property of Schneider Electric SE or its subsidiaries. All other brands may be trademarks of their respective owners.

This guide and its content are protected under applicable copyright laws and furnished for informational use only. No part of this guide may be reproduced or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), for any purpose, without the prior written permission of Schneider Electric.

Schneider Electric does not grant any right or license for commercial use of the guide or its content, except for a non-exclusive and personal license to consult it on an "as is" basis. Schneider Electric products and equipment should be installed, operated, serviced, and maintained only by qualified personnel.

As standards, specifications, and designs change from time to time, information contained in this guide may be subject to change without notice.

To the extent permitted by applicable law, no responsibility or liability is assumed by Schneider Electric and its subsidiaries for any errors or omissions in the informational content of this material or consequences arising out of or resulting from the use of the information contained herein.

As part of a group of responsible, inclusive companies, we are updating our communications that contain non-inclusive terminology. Until we complete this process, however, our content may still contain standardized industry terms that may be deemed inappropriate by our customers.

© 2023 – Schneider Electric. All rights reserved.

Table of Contents

Safety Information.....	5
Qualification of Personnel	5
Proper Use.....	6
Before You Begin.....	6
Start-up and Test.....	7
Operation and Adjustments	7
About the Book.....	9
Overview	12
Process Templates	12
Composition Strategy for Application Templates.....	12
Naming Convention	15
List of Process Templates.....	19
Common Services	21
Accessing General Purpose Library User Guides and Technical Documents using RTNS Feature	21
Hyperlink Services.....	22
Signal Conditioning and Processing	23
Signal Processing.....	23
Analog Input Signals - Use Case Description	23
\$AnalogInput - Analog Input Conditioning.....	26
\$AnalogInput1 - Analog Input with Configurable Range	28
\$AnalogLinear - Linear Interpolation	30
\$AnalogOutput - Analog Output Conditioning	31
\$AnalogCalc - Analog Calculation	33
\$DigitalCalc - Digital Calculation	35
\$DigitalInput - Digital Input Conditioning	36
\$DigitalOutput - Digital Output.....	38
\$MAnalogInput1 - Multiple Analog Input with Configurable Range.....	39
\$Range - Control Expert Analog Range.....	41
\$TOTAL - Totalizer	41
On/Off Device Control	45
\$HandValve - Manual Valve.....	45
\$Motor - 1 Speed/1 Rotation Direction Motor.....	47
\$Motor2 - 2 Speed/2 Rotation Directions Motor	53
\$MValve - Motorized Valve Without Positioner	57
\$Valve - On/Off Valve.....	61
\$Valve2 - On/Off Valve With Two Outputs	64
Analog Device Control	68
\$ControlValve - Control Valve.....	68
\$MotorVS - Motor with Variable Speed Drive	72
\$MValveWithPos - Motorized Valve with Positioner.....	76
Process Control.....	82
\$IMC - Internal Model Controller	82
\$LeadLag - Lead-Lag Controller	84
\$PID - PID Controller.....	85
\$PIDLoop - PID Closed Loop Circuit	87

\$PWMController - Pulse Width Modulated Controller	88
\$Ramp - Ramp.....	90
\$RatioCtl - Ratio Controller	91
\$SplitRange - Split-Range Controller	93
\$Step3Ctl - Three-Step Controller/Positioner	94
Sequential Control	97
\$SequenceDFB - Sequence Advanced Control	97
Using the \$SequenceDFB Template	102
Batch Phase Manager.....	107
\$IBPhase - Individual Batch Phase Manager.....	107
Using the \$IBPhase Template	112
Auxiliary Functions.....	115
\$AlarmSummary - Alarm Summary.....	115
\$AnalogSelect - Analog Signal Selector	117
\$AnalogSelect1 - Analog Selector with Monitoring Interface	118
\$MessageBox - Messages to the Operator	119
\$SPBool	120
\$SPReal	122
\$SPInt	123
\$SPDuration	125
Index	129

Safety Information

Important Information

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, service, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.



The addition of this symbol to a "Danger" or "Warning" safety label indicates that an electrical hazard exists which will result in personal injury if the instructions are not followed.



This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

DANGER

DANGER indicates a hazardous situation which, if not avoided, **will result in** death or serious injury.

WARNING

WARNING indicates a hazardous situation which, if not avoided, **could result in** death or serious injury.

CAUTION

CAUTION indicates a hazardous situation which, if not avoided, **could result in** minor or moderate injury.

NOTICE

NOTICE is used to address practices not related to physical injury.

Please Note

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and its installation, and has received safety training to recognize and avoid the hazards involved.

Qualification of Personnel

A qualified person is one who has the following qualifications:

- Skills and knowledge related to the construction and operation of electrical equipment and the installation.
- Knowledge and experience in industrial control programming.
- Received safety-related training to recognize and avoid the hazards involved.

The qualified person must be able to detect possible hazards that may arise from parameterization, modifying parameter values and generally from mechanical,

electrical, or electronic equipment. The qualified person must be familiar with the standards, provisions, and regulations for the prevention of industrial accidents, which they must observe when designing and implementing the system.

Proper Use

This product is a library to be used together with the automation control systems and is intended solely for the purposes described in the present documentation as applied in the industrial sector.

Always observe the applicable safety-related instructions, the specified conditions, and the technical data.

Perform a risk evaluation concerning the specific use before using the product. Take protective measures according to the result.

Since the product is used as a part of an overall system, you must ensure the safety of the personnel by means of the concept of this overall system (for example, machine concept).

Any other use is not intended and may be hazardous.

Before You Begin

Do not use this product on machinery lacking effective point-of-operation guarding. Lack of effective point-of-operation guarding on a machine can result in serious injury to the operator of that machine.

WARNING

UNGUARDED EQUIPMENT

- Do not use this software and related automation equipment on equipment which does not have point-of-operation protection.
- Do not reach into machinery during operation.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

This automation equipment and related software is used to control a variety of industrial processes. The type or model of automation equipment suitable for each application will vary depending on factors such as the control function required, degree of protection required, production methods, unusual conditions, government regulations, etc. In some applications, more than one processor may be required, as when backup redundancy is needed.

Only you, the user, machine builder or system integrator can be aware of all the conditions and factors present during setup, operation, and maintenance of the machine and, therefore, can determine the automation equipment and the related safeties and interlocks which can be properly used. When selecting automation and control equipment and related software for a particular application, you should refer to the applicable local and national standards and regulations. The National Safety Council's Accident Prevention Manual (nationally recognized in the United States of America) also provides much useful information.

In some applications, such as packaging machinery, additional operator protection such as point-of-operation guarding must be provided. This is necessary if the operator's hands and other parts of the body are free to enter the pinch points or other hazardous areas and serious injury can occur. Software products alone cannot protect an operator from injury. For this reason the software cannot be substituted for or take the place of point-of-operation protection.

Ensure that appropriate safeties and mechanical/electrical interlocks related to point-of-operation protection have been installed and are operational before

placing the equipment into service. All interlocks and safeties related to point-of-operation protection must be coordinated with the related automation equipment and software programming.

NOTE: Coordination of safeties and mechanical/electrical interlocks for point-of-operation protection is outside the scope of the Function Block Library, System User Guide, or other implementation referenced in this documentation.

Start-up and Test

Before using electrical control and automation equipment for regular operation after installation, the system should be given a start-up test by qualified personnel to verify correct operation of the equipment. It is important that arrangements for such a check are made and that enough time is allowed to perform complete and satisfactory testing.

⚠ WARNING

EQUIPMENT OPERATION HAZARD

- Verify that all installation and set up procedures have been completed.
- Before operational tests are performed, remove all blocks or other temporary holding means used for shipment from all component devices.
- Remove tools, meters, and debris from equipment.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Follow all start-up tests recommended in the equipment documentation. Store all equipment documentation for future references.

Software testing must be done in both simulated and real environments.

Verify that the completed system is free from all short circuits and temporary grounds that are not installed according to local regulations (according to the National Electrical Code in the U.S.A, for instance). If high-potential voltage testing is necessary, follow recommendations in equipment documentation to prevent accidental equipment damage.

Before energizing equipment:

- Remove tools, meters, and debris from equipment.
- Close the equipment enclosure door.
- Remove all temporary grounds from incoming power lines.
- Perform all start-up tests recommended by the manufacturer.

Operation and Adjustments

The following precautions are from the NEMA Standards Publication ICS 7.1-1995:

(In case of divergence or contradiction between any translation and the English original, the original text in the English language will prevail.)

- Regardless of the care exercised in the design and manufacture of equipment or in the selection and ratings of components, there are hazards that can be encountered if such equipment is improperly operated.

- It is sometimes possible to misadjust the equipment and thus produce unsatisfactory or unsafe operation. Always use the manufacturer's instructions as a guide for functional adjustments. Personnel who have access to these adjustments should be familiar with the equipment manufacturer's instructions and the machinery used with the electrical equipment.
- Only those operational adjustments required by the operator should be accessible to the operator. Access to other controls should be restricted to prevent unauthorized changes in operating characteristics.

About the Book

Document Scope

This document describes the process templates, and the Control and Supervision services that they provide.

For a more detailed description of their associated Control and Supervision services and configuration parameters, refer to the user guides mentioned in this document.

To use process templates, you need to have knowledge of EcoStruxure Process Expert, and of the Control and Supervision Participants.

Validity Note

This document has been updated for the release of EcoStruxure™ Process Expert 2023.

Related Documents

The characteristics that are described in the present document, as well as those described in the documents included in the Related Documents section below, can be found online. To access the information online, go to the Schneider Electric home page www.se.com/ww/en/download/.

The characteristics that are described in the present document should be the same as those characteristics that appear online. In line with our policy of constant improvement, we may revise content over time to improve clarity and accuracy. If you see a difference between the document and online information, use the online information as your reference.

Title of Documentation	Reference Number
EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual	EIO0000000988
EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual	EIO0000000989
EcoStruxure™ Process Expert - General Purpose Library Classic Device Templates Reference Manual	EIO0000001308
EcoStruxure™ Process Expert - General Purpose Library Classic Communication Templates Reference Manual	EIO0000001311
EcoStruxure Process Expert User Guide	EIO0000001114
EcoStruxure Process Expert Global Templates Reference Manual	EIO0000001986 (eng)
EcoStruxure Process Expert Runtime Navigation Services User Guide	EIO0000001574 (eng)

Technical Support

Visit <https://www.se.com/myschneider/> for support, software updates, and latest information.

Product Related Information

⚠ WARNING

LOSS OF CONTROL

- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.¹
- Test each implementation of a system for proper operation before placing it into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

¹ For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

Examples described in this manual are provided for information only.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Adapt examples that are given in this manual to the specific functions and requirements of your industrial application before you implement them.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Terminology Derived from Standards

The technical terms, terminology, symbols and the corresponding descriptions in this manual, or that appear in or on the products themselves, are generally derived from the terms or definitions of international standards.

In the area of functional safety systems, drives and general automation, this may include, but is not limited to, terms such as safety, safety function, safe state, fault, fault reset, malfunction, failure, error, error message, dangerous, etc.

Among others, these standards include:

Standard	Description
IEC 61131-2:2007	Programmable controllers, part 2: Equipment requirements and tests.
ISO 13849-1:2015	Safety of machinery: Safety related parts of control systems. General principles for design.
EN 61496-1:2013	Safety of machinery: Electro-sensitive protective equipment.

Standard	Description
	Part 1: General requirements and tests.
ISO 12100:2010	Safety of machinery - General principles for design - Risk assessment and risk reduction.
EN 60204-1:2006	Safety of machinery - Electrical equipment of machines - Part 1: General requirements.
ISO 14119:2013	Safety of machinery - Interlocking devices associated with guards - Principles for design and selection.
ISO 13850:2015	Safety of machinery - Emergency stop - Principles for design.
IEC 62061:2015	Safety of machinery - Functional safety of safety-related electrical, electronic, and electronic programmable control systems.
IEC 61508-1:2010	Functional safety of electrical/electronic/programmable electronic safety-related systems: General requirements.
IEC 61508-2:2010	Functional safety of electrical/electronic/programmable electronic safety-related systems: Requirements for electrical/electronic/programmable electronic safety-related systems.
IEC 61508-3:2010	Functional safety of electrical/electronic/programmable electronic safety-related systems: Software requirements.
IEC 61784-3:2016	Industrial communication networks - Profiles - Part 3: Functional safety fieldbuses - General rules and profile definitions.
2006/42/EC	Machinery Directive
2014/30/EU	Electromagnetic Compatibility Directive
2014/35/EU	Low Voltage Directive

In addition, terms used in the present document may tangentially be used as they are derived from other standards such as:

Standard	Description
IEC 60034 series	Rotating electrical machines
IEC 61800 series	Adjustable speed electrical power drive systems
IEC 61158 series	Digital data communications for measurement and control – Fieldbus for use in industrial control systems

Finally, the term zone of operation may be used in conjunction with the description of specific hazards, and is defined as it is for a hazard zone or danger zone in the Machinery Directive (2006/42/EC) and ISO 12100:2010.

NOTE: The aforementioned standards may or may not apply to the specific products cited in the present documentation. For more information concerning the individual standards applicable to the products described herein, see the characteristics tables for those product references.

Overview

Process Templates

Description

The EcoStruxure Process Expert General Purpose library provides resources:

- That have been pre-configured and tested by Schneider Electric.
- That are designed for automating a large variety of processes.

Process templates model generic functionalities of process equipment but also hardwired signals from the field, and variables. They represent the application of the system.

They are used to implement the Control and Supervision services that are provided by the software Participants, streamlining the engineering of systems.

The resources providing these services are encapsulated in dedicated facet references, which are then organized in composite references.

At the highest level, the process template represents the control module.

You can easily customize the services that are provided by process templates to fulfill your specific requirements. You can also link process templates with compatible device and communication templates, using interfaces.

Process Template Services

Facet templates referenced by process control module templates provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed. Function blocks and variables are the resources that are encapsulated in these facet references to provide such services.
Supervision	These services complement the Control services. Supervision services are optional and those corresponding to selected Control are activated by default. Tags, alarms, and genies are the resources that are encapsulated in these facet references to provide such services. Data is provided by the associated Control resources.

Also, for both Control and Supervision services, you can configure parameters during instantiation to meet the requirements of your system.

Composition Strategy for Application Templates

General

The following general rules are applied when designing application library templates:

- Decoupling between participants
- Coupling into the participant
- Composition approach
- Exposed interfaces

Decoupling Between Participants

Maximum decoupling between facets from different participants is expected. One participant does not need to know the internal implementation of the other one to be able to collaborate.

For instance, the Control Expert logic facets expose the OPC items to be required for the Citect SCADA data facets in order that the Citect SCADA data facets are not altered as a result of any change in the Control Expert logic facet data structures (DDT field names). So, the interfaces between participants contain so many items as required to respect this rule.

Coupling into the Participant

To minimize the data to be exchanged between facets from the same participant, the library designer can assume the details of the data structure that is shared between the facets by means of the interfaces.

For instance, the Citect SCADA Genie facets can assume which is the naming convention being implemented for Citect SCADA tags into the Citect SCADA data facets. That means that, by sharing the name of the object being implemented through the interface, the genie can assume the name of tags involved.

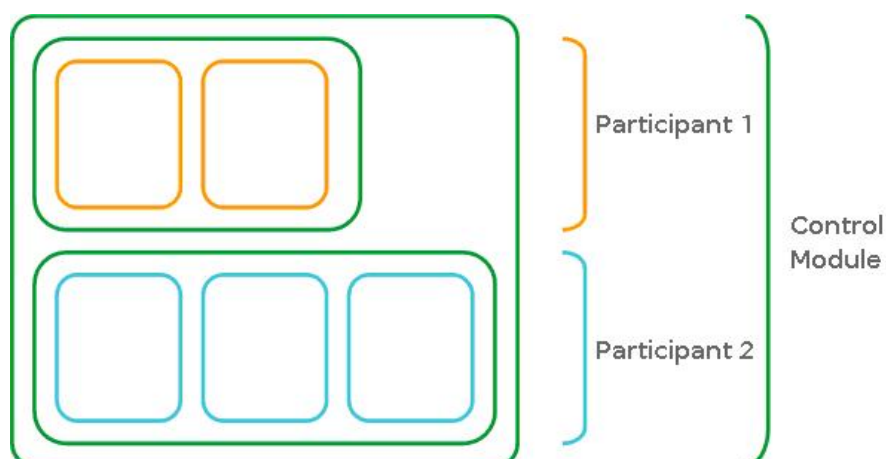
But, if the templates being designed could be used directly from the system application and linked by means of interface links, you need to provide the right connectivity between them (the user making the system application can only make interface links for establishing the relationship between different instances; whereas, the library designer has the opportunity to access the elements into the interfaces).

Composition Approach

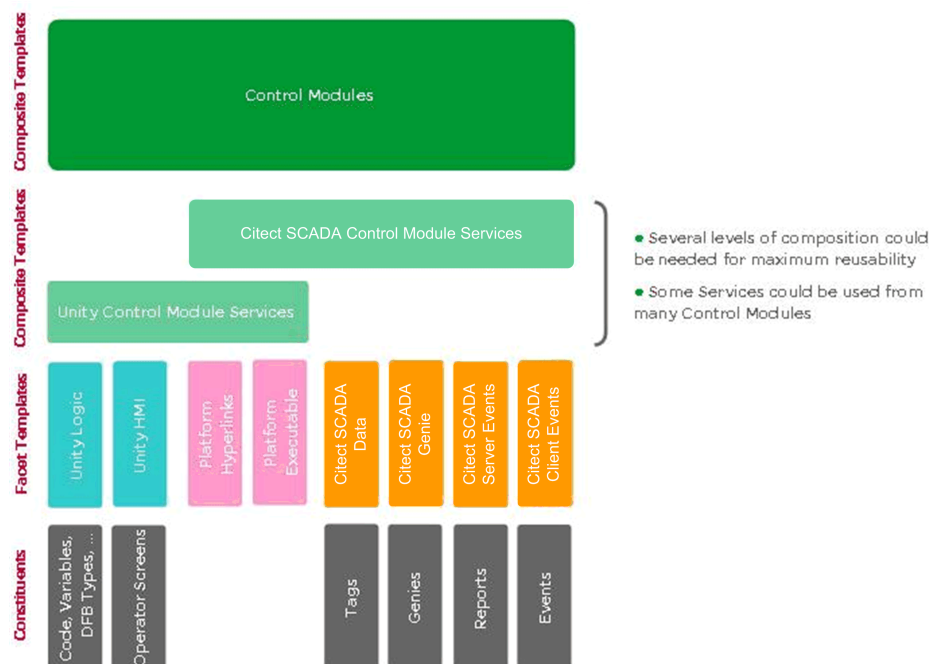
The objective is to provide composite templates which bring the features required for a control module (as defined in ISA-S88) that involves one or more participants when needed. Avoid the use of design composite templates with too many nested levels to avoid big changes propagation and to facilitate their maintainability. The libraries provide facet and composite templates which could be used to create instances or as part of other composite templates. This modular design allows the user to instantiate such templates in a consistent way for creating the application, independent of the participants involved and the required services (optional elements in the composite templates).

The first level of encapsulation is by participant. So, a horizontal approach is implemented. Composite templates offer the functionality from a concrete participant that can be combined for providing the complete functionality expected for a type of control module (that is, Motor 1 speed 1 direction).

The following figure shows general composition approach.



The following figure shows resources for application templates.



Another aspect to be considered is the perimeter for the control module templates.

The following capabilities are encapsulated by means of independent control module templates that the user can link in instantiation time:

- Foundation templates: Hardware Projection, Hardware Diagnosis, Peer to Peer, and so on.
- Communication templates.
- Device templates.
- Process templates: Optionally the process templates include the projection of their related hardwired signals to facilitate the process of creation of hardwired signals with a minimum effort.

Exposed Interfaces

The templates expose interfaces that allows the user to make the following types of connections:

- To topological instances (mapping interfaces): When the application template requires some data provided by the topological instances (that is, I/O addresses), they are exposed through the mapping interfaces. Such types of interfaces can be used only from facet templates (not from composite templates) as the mapping process is executed from facets assigned to projects.
- To other application instances (application interfaces): The templates offer at least the capability to make the more usual connections between them (that is, analog input -> PID -> analog output) at the system level when such connections are complex (several data to be shared). This means that the user could create some logical connections between instances created directly at the system application.

The standard library does not offer low level services which enable the user to do the following:

- Make any connection between application instances.
- Replace the services offered by the participants themselves (that is, FBD capabilities to define control logics for implementing Interlocks).

The interface names help the user to understand the concrete meaning of the interfaces in their specific scope.

So, even when the interface definition name and the chosen role are determining the data to be exchanged, the name of the interface allows the user to do the following:

- to determine its specific purpose.
- to distinguish between interfaces exposed by the same template which were created from the same interface definition.

Naming Convention

General

A key point for harmonization of the templates is to use a standard naming convention.

The general rules are applied depending on the following type of template being defined:

- Any types of templates
- Any interface models
- Mapping interface models
- Application interface models
- Application facet templates
- Application composite templates

Any Types of Templates

The following rules are applied to any types of templates:

- **Prefix:** The character \$ is used as a prefix for templates made by Schneider Electric (Standard templates including interface models). This is avoiding any overlap with templates created by other users.
- **Aliases:** The objects model is bringing the possibility to use aliases to hide the origin of the deferred selections, parameters and interfaces. In the standard library, we are not using such capability. So the user creating references (Definition time) or instances (Instantiation time) will see the real name of the related selections, parameters, and interfaces.
- **Documentation:** Templates need to include references for enabling the access to their related engineering documentation (that is, user manuals). Such documentation needs to be attached to the related function (that is, `DEVCTL` explanation as part of the composite where the Control Expert Logic and the Control Expert HMI facets are encapsulated).

Even when such facets are not explicitly in some of the drawings included in this document, they are considered implicitly as included.

- **Constituent names:** To allow using multiple instances of the same template into the same participant project, it is commonly required to generate unique identifiers. The strategy to be applied is to add suffixes as in EcoStruxure Process Expert (for instance, `M1001_DEVCTL`, `M1001_DEVCTL_ST`, `M1001_CONDSUM`, and so on). That is also confirming the compatibility with previously created resources (that is, Smart Genies for representing the motor implemented through the Control function `DEVCTL`).

Verify that the templates are establishing similar suffixes for avoiding the names conflicts. The designed templates force the user to use the same naming convention when creating new composite templates. The designed templates are created by recombining the ones coming from the standard libraries. This process increases their reusability even for the standard templates.

So, the constituents names are calculated by means of bindings by the following rules:

- The root of the constituent names is generated by using the name of the instance entered by the user.
- The suffixes need to be added only when the fact of not using them is creating a name conflict at the template level that is being defined. This means that establishing suffixes need to be delayed as much as possible to the top composition levels.
- The suffixes should follow the naming convention applied in EcoStruxure Process Expert and need to be useful for understanding the role of the constituents in the concrete scenario.

For instance:

- The user creates an instance of the composite template `$Motor` and enters the name `M1001` for it. This is reducing (but not avoiding) names conflicts (in fact, they can be only avoided by the participant in generation time regardless the chosen naming convention).
- This name is propagated to encapsulated templates through bindings without any suffices up to the level where that is causing a conflict (see the example in the next point).
- When combining the Control functionality for controlling the motor (`DEVCTL`) and the detected failures management (`CONDSUM`), it is required to concatenate the related suffixes (`M1001_DEVCTL` and `M1001_CONDSUM`) to avoid creation of 2 DFBs with the same name, which is not allowed in Control projects.
- When the facet defines the required constituents for implementing the control part of the object (`DEVCTL`), it is required to concatenate additional suffixes (`M1001_DEVCTL_ST` for the status data, `M1001_DEVCTL_CFG` for the configuration data) for avoiding to create duplicate names.
- In such case, if the user decides to instantiate directly the last mentioned facet template, it provides that no generated objects have duplicate names (into the scope of the instance). The template does not force the user to use the complete naming convention. For instance, `M1001` (`DEVCTL`), `M1001_ST` and `M1001_CFG`; rather than `M1001_DEVCTL`, `M1001_DEVCTL_ST` and `M1001_DEVCTL_CFG` generated when instantiating the template `$Motor`.

Any Interface Models

The following rules are applied:

- **Roles representation:**
 - Unidirectional interfaces: Female representation is used for the role in charge of producing the values for elements whereas Male is used for the other role.

Mapping Interface Models

The following rules are applied:

- Role names:
 - CO: Is used from the application templates representing communications objects (CO) that receive the data provided from the system configuration. This role is represented graphically as male.
 - DO: Is used from the application templates representing device objects (DO) that receive the data provided from the system configuration. This role is represented graphically as male.
 - HO: Is used from the topological template (also called Hardware Objects (HO)) to expose the information provided by the hardware (for example, topological addresses, slave number, and so on) to the project facets. This role is represented graphically as female.
 - SO: Is used from the application templates representing the signal objects (SO) or I/O module objects that receive the data provided from the system configuration. This role is represented graphically as male.

Application Interface Models

As the same interface models can be used for several purposes and the objective of this key mechanism is to implement the collaboration between participants, their names need to be as generic as possible.

The naming convention is conditioned by the nature of the interface models.

The following types of interface models are considered for the standard library:

- Elementary interface models: The more elementary interface models are for exchanging just one element of an elementary type (not a nested Interface). These interfaces are for a generic usage as the interface name (the unique identifier) determines the type of the transported data.
- Single element interface models: Interfaces that contain just one element for a specific usage are named as the constituent type from the participant.
- Multiple elements interface models: Interfaces that contain several elements with a common objective. Instead of having several elementary interface models, one new interface model is created with many elements as needed.

The following rules are applied depending on the types:

- Any types of interface models
- Elementary interface models
- Single element interface models
- Multiple element interface models

Any Types of Interface Models

The following general rules are applied to any interface models:

- Interface model name: It is recommended to use upper and lowercase for better readability. The exception is when we use names that are directly used by the participant. In such case, the exact name is used.

- Role names:

Definition and references: For example, a template is used for defining some constituents (that is, definition of Control variables) that could be used by one or more similar participants which need to access to such constituents (that is, references to Control variables).

In such case, the role names to be used are:

- Definition: Is for the role in charge of defining the constituents. This role is represented graphically as female.
- Reference: Is for the role in charge of accessing (it does not matter for which kind of access: read, write, read/write, and so on) to such constituents. This role is represented graphically as male.

Elementary Interface Models

- Interface model name: The name of the real-time data type being exchanged is directly used for identifying the interface model (that is, \$Bool for exchanging the name of a boolean constituent). It is required to limit the usage of the data being exchanged.

For instance, to a concrete participant, they could be added some suffixes for avoiding. Such interface models could be used outside of its expected scope.

- Element name: The element is identified as name.

Single Element Interface Models

- Interface model name: The name needs to be identical to the type of the data being exchanged as defined in the participant (that is, \$DEVCTL for exchanging the name of a Control variable of type \$DEVCTL).
- Element name: The element is identified as name.

Multiple Element Interface Models

- Interface definition name: The name should reflect the meaning of the set of elements (that is, \$DevCtlStatus for exchanging the names of the OPC items in the Control Expert Logic which are needed for accessing from Supervision).

Application Facet Templates

As much as possible, the name of the facet templates needs to be closer to the name of the encapsulated constituents or elements. As there could be several facet templates addressing the same functionality but from different points of view (for instance, different facet types) of the participant, consider the possibility to add some suffixes for distinguishing them.

- Suffixes: The following suffixes are used for identifying the templates:
 - _UL for Control Expert logic facet templates.
 - _UH for Control Expert HMI facet templates.
 - _CD for Citect SCADA data facet templates.
 - _CG for Citect SCADA genie facet templates.
 - _CR for Citect SCADA server event facet templates.
 - _CC for Citect SCADA client event facet templates.
 - _PH for platform Hyper link facet templates.

Application Composite Templates

As the composite templates are used to combine functionality provided by other facet or composite templates, it should reflect the complete functionality being encapsulated.

Basically, there are 2 use cases:

- **Control module:** Typically the composite template combines functionality from several participants for representing the complete functionality required for a type of control module (that is, motor 1 speed 1 direction). In such case, the type of control module is abbreviated to determine the name of the template. It is recommended to use upper and lowercase for maximizing readability (that is, \$Motor).
- **Control module services:** The contained functionality is provided by the same participant for covering some services required for implementing one or more types of control modules. In such case, a suffix is added for expressing their scope (that is, the suffix _UC for the template \$Motor_UC is expressing that it encapsulates functionality required from the perspective of the Control control for implementing a motor 1 speed 1 direction).

Suffixes: The following suffixes are used for identifying the templates:

- _UC for Control Expert logic composite templates.
- _UH for Control Expert HMI composite templates.
- _CS for Citect SCADA data composite templates.
- _CG for Citect SCADA genie composite templates.
- _CR for Citect SCADA server event composite templates.
- _CC for Citect SCADA client event composite templates.
- _CS for Citect SCADA (data and genie) composite templates.

List of Process Templates

List of Families

The process templates described in this document are grouped by family.

The table lists the control module templates of each family:

Family name	Process templates	Description
Signal processing	\$AnalogCalc, page 33	Analog calculations
	\$AnalogInput, page 26	Analog input conditioning
	\$AnalogInput1, page 28	Analog input conditioning with configurable range
	\$AnalogLinear, page 30	Linearization
	\$AnalogOutput, page 31	Analog output conditioning
	\$DigitalCalc, page 35	Digital calculations
	\$DigitalInput, page 36	Digital input conditioning
	\$DigitalOutput, page 38	Digital output conditioning
	\$MAnalogInput1, page 39	Multiple analog inputs with configurable range
	\$Range, page 41	Control Expert analog range
		16-channel BMXERT

Family name	Process templates	Description
	\$TOTAL, page 41	Totalizer
On/Off device control	\$HandValve, page 45	Hand valve
	\$Motor, page 47	1-speed/1-rotation direction motor
	\$Motor2, page 53	2-speed/2-rotation direction motor
	\$MValve, page 57	Motorized valve without positioner
	\$Valve, page 61	On/Off valve
	\$Valve2, page 64	On/Off valve (2 outputs)
Analog device control	\$ControlValve, page 68	Control valve
	\$MotorVS, page 72	Motor with variable speed drive
	\$MValveWithPos, page 76	Motorized valve with positioner
Process control	\$IMC, page 82	Internal model controller
	\$LeadLag, page 84	Lead-lag controller
	\$PID, page 85	PID controller
	\$PIDLoop, page 87	Closed loop control combining the AnalogInput1, PID, and AnalogOutput process templates.
	\$PWMController, page 88	Pulse width modulated controller
	\$Ramp, page 90	Ramp
	\$RatioCtl, page 91	Ratio controller
	\$SplitRange, page 93	Split-range controller
	\$Step3Ctl, page 94	3-Step controller/positioner
Sequential control	\$SequenceDFB, page 97	Sequence advanced control
Batch Phase Manager	\$IBPhase, page 107	Individual Batch phase manager
Auxiliary functions	\$AlarmSummary, page 115	Alarm summary
	\$AnalogSelect, page 117	Analog signal selector
	\$AnalogSelect1, page 118	Analog signal selector with monitoring variables
	\$MessageBox, page 119	Messages to the operator
	\$SPBool, page 120	BOOL type setpoint
	\$SPReal, page 122	REAL type setpoint
	\$SPInt, page 123	INT type setpoint
	\$SPDuration, page 125	TIME type setpoint

Common Services

Overview

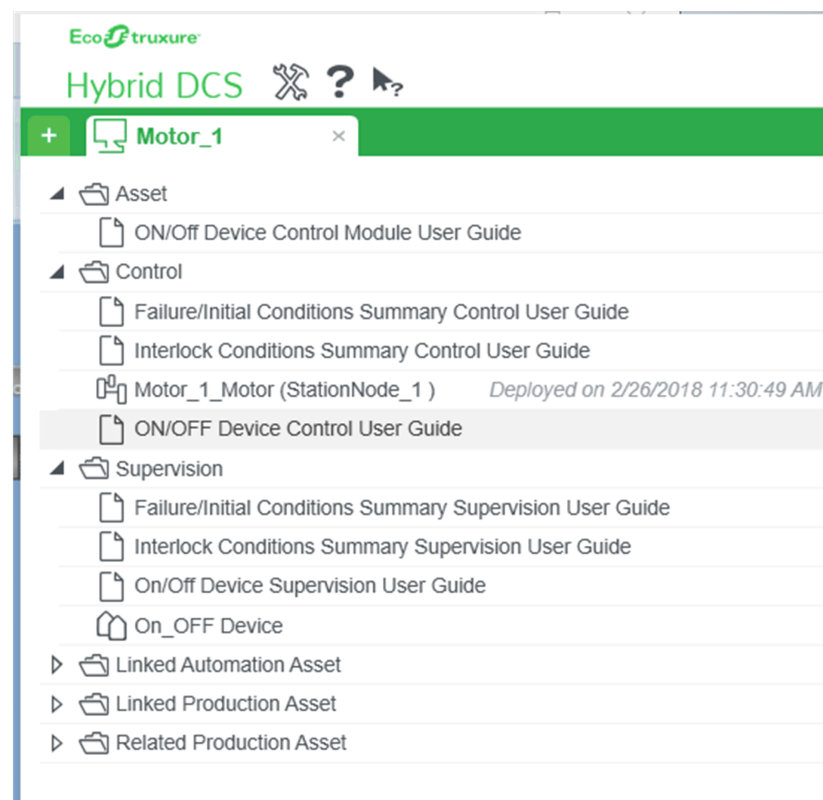
This chapter describes the different services that are common to the control modules.

Accessing General Purpose Library User Guides and Technical Documents using RTNS Feature

General Description

The General Purpose Library user guides and technical documents can be accessed using the Runtime Navigation Services (see EcoStruxure™ Process Expert, Runtime Navigation Services , User Guide) provided by the EcoStruxure Process Expert.

In the Operation Client, the user guides are segregated as shown in the image below.

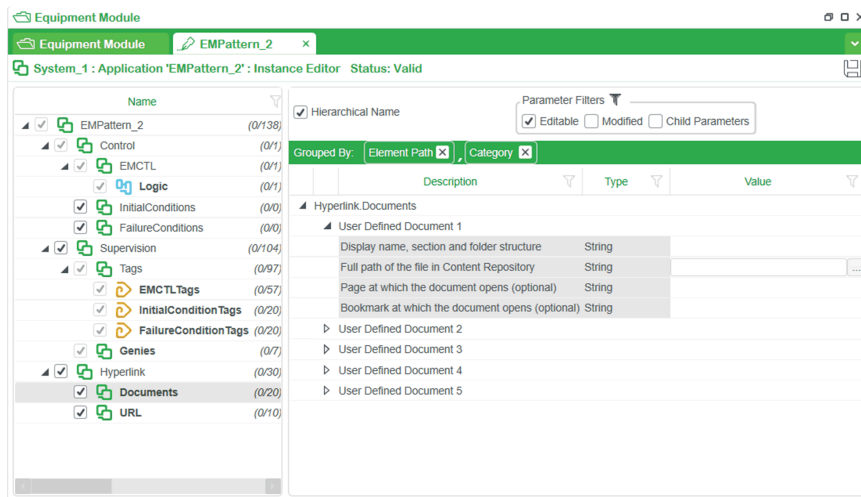


NOTE: The user guides of the optional Services (for example, *Failure Conditions*, *Interlock Conditions*) will be available in their respective sections (**Asset**, **Control** or **Supervision**) if you have enabled the optional services in the EcoStruxure Process Expert application.

Hyperlink Services

General Description

The General Purpose Library control modules allows you to link the documents and URL to its instances. The hyperlink service is disabled by default, you have to enable the hyperlink service and configure the parameters as shown in the below screen shot. For more details refer to the topic describing how to use the hyperlink service (see EcoStruxure™ Process Expert, Runtime Navigation Services , User Guide).



NOTE: A maximum of five documents and five URLs can be added to each instance.

Signal Conditioning and Processing

Overview

This chapter describes the functionality of process templates for signal conditioning and processing, and their composition.

These templates do not reflect any specific installation.

⚠ WARNING

LOSS OF CONTROL

- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.¹
- Test each implementation of a system for proper operation before placing it into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

¹ For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

Signal Processing

Overview

This section describes the functionality of the signal processing templates and their composition.

Analog Input Signals - Use Case Description

Overview

This topic describes the different use cases to read an analog signal from the channel of an analog input module.

It includes the use case where the analog signal is read from an STBAHI8321 HART interface module that received signals from a Krohne field instrument or other HART device.

The signal acquisition can be done by either or a combination of the following templates:

- \$AnalogInput, page 26
- \$AnalogInput1, page 28
- \$AHISignal_UL (see EcoStruxure™ Process Expert , Foundation Application Templates, User Guide)

For each use case, it provides information on how to implement it and on the result that you can see when refining the logical Control Participant project. For implementation details, refer to the *EcoStruxure Process Expert User Guide*.

Prerequisites

To perform the steps described in the following tables, you need to create the necessary topological entities, which are, depending on the use case:

- A controller with an analog input module
- An Advantys island with an STBAHI8321 HART interface module
- A topological Krohne device entity, matching the application template

You also need to create the corresponding Control project and assign the Control facets of the described instances.

Analog Input Signal from I/O Card

You are reading a signal from the channel of an I/O module of the controller:

Instantiated templates and element selection	Instance configuration	Hardware mapping	Logic generated in Control Participant project
\$AnalogInput or \$AnalogInput1 AnalogInputSignal element: Selected	<ul style="list-style-type: none"> • Include also Signal Quality: Selected • Data type: Int 	AIChannel/SO to AIChannel/HO	AISignalCond1 EFB along with variables (SIGNALVALUE, ERROR, COMSTATUS, MODULESTATUS, and NOTMODULESTATUS) to provide signal quality information and AINPUT or AINPUT1 DFBs
	<ul style="list-style-type: none"> • Include also Signal Quality: Unselected Data type: Int 		AISignalCond1 EFB without quality signal variables only SIGNALVALUE variable at CHANNELVALUE pin and AINPUT or AINPUT1 DFBs

Analog Input Signal from STBAHI8321 HART Module

You are using a Krohne instrumentation library template in the application, which references the \$AHISignal_UL facet template:

Instantiated templates and element selection	Hardware mapping	Logic generated in Control Participant project
Krohne template AHISignal element: Selected	AHI8321ChData/SO to AHI8321ChData/HO	Corresponding Krohne device and AHISIGNAL DFBs

You are using a HART device that is not part of the Krohne instrumentation library and you are mapping its signals to the STBAHI8321 HART module:

Instantiated templates and element selection	Instance configuration	Instance links	Hardware mapping	Logic generated in Control Participant project
Preferred method				
\$AnalogInput or \$AnalogInput1 AnalogInputSignal element: Selected	<ul style="list-style-type: none"> • Include also Signal Quality: Selected by default • Data type: Real 	–	AIChannelReal/SO to AIChannelReal/HO for each channel.	AISignalCondReal1 EFB along with variables (SIGNALVALUE, ERROR, COMSTATUS, MODULESTATUS, and NOTMODULESTATUS) to provide signal quality information and AINPUT or AINPUT1 DFBs
Allows reading the following variables: <i>PV, SV, TV, QV, CurrentVal</i> .				
Instantiated templates and element selection	Instance configuration	Instance links	Hardware mapping	Logic generated in Control Participant project
Alternative method, using the \$AHISignal_UL facet template for Krohne-like custom made templates				
\$AHISignal_UL \$AnalogInput or \$AnalogInput1 AnalogInputSignal element: Selected NOTE: You can instantiate up to four \$AnalogInput or \$AnalogInput1 templates depending on the number of channels you want to map.	<ul style="list-style-type: none"> • Include also Signal Quality: Selected by default • Data type: Real 	Link the signal interface of the \$AHISignal_UL instance to the RealVar interface of the \$AnalogInput or \$AnalogInput1 instance. NOTE: For the output interface, for each channel, you can select between two, with or without signal quality information.	AHI8321ChData/SO to AHI8321ChData/HO	AHISIGNAL and AINPUT or AINPUT1 DFBs
Allows reading the <i>DiagData</i> variable per channel.				

NOTE: Instantiate as many \$AnalogInput or \$AnalogInput1 templates as there are channels to be read (up to four).

Analog Input Signal from an External Variable

You are providing the signal and the signal quality through an EDT template or a link with another process instance:

Instantiated templates and element selection	Instance configuration	Instance links	Logic generated in Control Participant project
Template providing the variable of type Int \$AnalogInput or \$AnalogInput1 AnalogInputSignal element: Selected	<ul style="list-style-type: none"> • Include also Signal Quality: Selected • Data type: Int 	Link the interface providing the variable value and quality signal to the AINPUTSignalExt interface of the \$AnalogInput or \$AnalogInput1 instance.	AINPUT or AINPUT1 DFB with variables at CHIN and CHFAILURE pins
	Include also Signal Quality: Unselected Data type: Int	Link the interface providing the variable value to the IntVar interface of the \$AnalogInput or \$AnalogInput1 instance.	AINPUT or AINPUT1 DFB with variable at CHIN pin
Template providing the variable of type Real \$AnalogInput or \$AnalogInput1 AnalogInputSignal element: Selected	Include also Signal Quality: Selected by default Data type: Real	Link the interface providing the variable value to the RealVar interface of the \$AnalogInput or \$AnalogInput1 instance.	

\$AnalogInput - Analog Input Conditioning

Description

The \$AnalogInput process control module template is designed to read signals, and optionally, signal quality data from analog input modules. Alternatively, it can receive signals from an external variable of type Int or Real.

You can also use the template to read signals and signal quality of one channel of an STBAHI8321 HART interface module, page 23.

The template provides linear scaling, cut-off, diagnosis, and simulation functions that you can configure in the **Properties** window of the instance (see EcoStruxure™ Process Expert, User Guide).

During operation, genies and the corresponding faceplate allow you to display the measured value, the set point, ranges, alarms, and so on.

NOTE: For more information on how to use the template, refer to the description of possible use cases, page 23.

Common Links

You can link the \$AnalogInput template with \$AnalogCalc, page 33 process template to perform calculations on the signal value by using the **Links Editor** (see EcoStruxure™ Process Expert, User Guide).

Parameters

Configuration

The table describes the **Configuration** parameters of the \$AnalogInput template that you can configure:

Element Name	Name	Type	Default value	Description
Logic	BadPV	Float	0.0	Default value in engineering units to be used for the output signal when the channel is not operational.
	HysteresisSP	Float	0.0	Hysteresis setpoint for disconnecting the detected alarms.
	DWATCH Selection	Enum	True Value	Allows enabling or disabling the monitoring of the corresponding alarm. You can configure the input in one of the following ways: <ul style="list-style-type: none"> True Value: Enables monitoring of the alarm by setting the value of the corresponding pin to TRUE. Interface: Enables the corresponding interface, allowing you to provide the value through an interface of type \$Bool. Refinement: Allows you to set the value during refinement. The default value that is generated for the corresponding pin is NULL.
	HHWATCH Selection			
	HWATCH Selection			
	LLWATCH Selection			
	LWATCH Selection			

Range

The table describes the **Range** parameters of the \$AnalogInput template that you can configure:

Element Name	Name	Type	Default value	Description
Logic	HighRaw	Short	2047	High range of the input signal coming from the channel.
	LowRaw	Short	0	Low range of the input signal coming from the channel.
	PVRangeHigh	Float	100.0	High range of the output measurement (engineering units), corresponding to the maximum value of the channel input (configured for the HighRaw parameter).
	PVRangeLow	Float	0.0	Low range of the output measurement (engineering units), corresponding to the minimum value of the channel input (configured for the LowRaw parameter).
	PVEngineeringUnit	String	%	Engineering units that are used to display the process value.
	PVFormat	String	####.#EU	Format that is used to display the process value.

Time

The table describes the **Time** parameters of `$AnalogAlarms` that you can configure:

Element Name	Name	Type	Default value	Description
Logic	TONSP	Duration	00:00:00	Timing setpoint for connecting the active alarms.
	TOFFSP	Duration		Timing setpoint for disconnecting the active alarms.

For a description of the configurable parameters of the optional **AnalogInputSignal** element, refer to the corresponding Control facet template that is indicated in the composition of the `$AnalogInput` template.

Composition

The `$AnalogInput` template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$AnalogInput` control module and the corresponding facet, which implements the service:

Control services (\$AnalogInput_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$AnalogInput_CS)*	Corresponding facet template	Supervision service description
Core services						
AnalogInput	Logic	\$AINPUT_UL	Refer to the AINPUT DFB**.	AnalogInput- Tags	\$AINPUT_CD	Refer to Analog Input Management***.
	—	—	—	Analog input genies	\$AIIPVA_CG	
					\$AIIPVSPA_CG	
					\$AIPVA_CG	
					\$AIPVSPA_CG	
Optional I/O signal						

Control services (\$AnalogInput_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$AnalogInput_CS) *	Corresponding facet template	Supervision service description
AnalogInputSignal		\$AISignal_UL (see EcoStruxure™ Process Expert , Foundation Application Templates, User Guide)	Refer to the <i>AI/SignalCondReal1</i> and <i>AI/SignalCond1</i> EFB (see EcoStruxure™ Process Expert, Foundation Control Services User Guide).	–	–	–
Optional services						
AnalogAlarms	Logic	\$AALARM_UL	Refer to the AALARM DFB**.	AlarmTags *	\$AALARM_CD	Refer to <i>Analog Input Management</i> .
NOTE: * The service is activated by default. ** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual *** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The \$AnalogInput template exposes the following interfaces:

Interface identifier	Type/role	Description
PV	\$Real/Def	Provides the process value in engineering units.
PVQ	\$ASignalwithQuality/Def	Links to a process object
PVRanged	\$RangedValue/Def	Provides the following data as configured in the Properties window of the instance: <ul style="list-style-type: none"> • High Range (raw) • Low Range (raw) • High Range (eng. units) • Low Range (eng. units) • Engineering Units • Format Type
IntVar	\$Int/Ref	To acquire the analog signal from an external variable of type Int
AINPUTSignalExt	\$AINPUTSignal/HO	To acquire the analog signal value and signal quality from an external variable of type Int
RealVar	\$Real/Ref	To acquire the analog signal value and signal quality from an external variable of type Real
AInputName	\$AINPUT_IF/Def	Instance name concatenated with control block name

\$AnalogInput1 - Analog Input with Configurable Range

Description

The \$AnalogInput1 process control module template is designed to read signals, and optionally signal quality data from analog input modules. Alternatively, it can receive signals from an external variable of data type Int or Real.

You can also use the template to read 1 signal value and signal quality of 1 channel of an STBAHI8321 HART interface module, page 23.

The template encapsulates Control logic, which provides linear scaling, cut-off, diagnosis, alarm, and simulation functions that you can configure in the **Properties** window of the instance (see EcoStruxure™ Process Expert, User Guide).

During operation, genies and the corresponding faceplate allow you to display the measured value, the set point, ranges, alarms, and so on.

In addition to the features provided by the `$AnalogInput` template, this template allows you to configure the signal range from the Supervision runtime.

NOTE: For more information on how to use the template, refer to the description of possible use cases, page 23.

Common Links

You can link the `$AnalogInput1` template with `$AnalogCalc`, page 33 process template to perform calculations on the scaled signal value by using the **Links Editor** (see EcoStruxure™ Process Expert, User Guide).

Parameters

The parameters of this template that you can configure are the same as for the `$AnalogInput` template, page 26

Composition

The `$AnalogInput1` template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$AnalogInput1` control module and the corresponding facet, which implements the service:

Control services (\$AnalogInput1_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$AnalogInput1_CS)	Corresponding facet template	Supervision service description
Core services						
AnalogInput1	Logic	\$AINPUT1_UL	Refer to the AINPUT1 DFB**.	Analog Input1Tags	\$AINPUT1_CD	Refer to Analog Input with configurable range Management***.
	—	—	—	Analog input1 genies	\$AI1LPVSPA_CG	
					\$AI1LPVA_CG	
					\$AI1RPVA_CG	
					\$AI1RPVSPA_CG	
Optional I/O signal						
AnalogInputSignal		\$AISignal_UL	Refer to the AISignalCondReal1 and AISignalCond1 EFB (see EcoStruxure™ Process Expert,	—	—	—

Control services (\$AnalogInput1_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$Ana- logI- nput1_CS)	Corresponding facet template	Supervision service description
			Foundation Control Services User Guide).			
Optional services						
AnalogAlarms	Logic	\$AALARM_UL	Refer to the AALARM DFB**.	AlarmTags *	\$AALARM_CD	Refer to <i>Analog Input with configurable range Management.</i>
<p>* The service is activated by default.</p> <p>** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual</p> <p>*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual</p>						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The \$AnalogInput1 template exposes the following interfaces:

Interface identifier	Type/role	Description
PV	\$Real/Def	Provides the process value in engineering units.
PVQ	\$ASignalwithQuality/ Def	Links to a process object.
PVRanged	\$RangedValue/Def	Provides the following data as configured in the Properties window of the instance: <ul style="list-style-type: none"> • High Range (raw) • Low Range (raw) • High Range (eng. units) • Low Range (eng. units) • Engineering Units • Format Type
IntVar	\$Int/Ref	To acquire the analog signal from an external variable of type Int
AINPUTSignalExt	\$AINPUTSignal/HO	To acquire the analog signal value and signal quality from an external variable of type Int
RealVar	\$Real/Ref	To acquire the analog signal value and signal quality from an external variable of type Real
ExtPVQ	\$PVQ/PO	External PV with quality. Links to a device object.
AInput1Name	\$AINPUT1_IF/Def	Instance name concatenated with control block name

\$AnalogLinear - Linear Interpolation

General Description

The \$AnalogLinear process control module template is used to transform a signal by using a linear characterization function. The characterization function is defined with the use of points (up to five).

Composition

The `$AnalogLinear` template references composite and facet templates, which provide the core Control services.

The table describes the services that are available from the `$AnalogLinear` control module and the corresponding facet, which implements the service:

Control services	Composite template	Corresponding facet template	Control service description
Logic	<code>\$AnalogLinear_UC</code>	<code>\$ALINEAR_UL</code>	Refer to the <code>ALINEAR</code> DFB (see EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual).

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
Error	<code>\$Bool/Def</code>	EDT interface
X	<code>\$Real/Ref</code>	EDT interface
Y	<code>\$Real/Def</code>	EDT interface

`$AnalogOutput` - Analog Output Conditioning

General Description

The `$AnalogOutput` process control module template is used to condition an analog signal usually associated with a physical output.

The process template provides linear scaling, interlocking, diagnosis, and owner management functions that you can configure in the **Properties** window of the instance (see EcoStruxure™ Process Expert, User Guide).

Parameters

Configuration

The table describes the **Configuration** parameters of the `$AnalogOutput` template that you can configure:

Element Name	Name	Type	Default value	Description
Logic	DeltaSP	Float	1.0	Absolute increase/decrease value in SP when button is pressed.
	DeltaFastSP	Float	2.0	Absolute increase/decrease value in SP when fast button is pressed.

Time

The table describes the **Time** parameters of the `$AnalogOutput` template that you can configure:

Element Name	Name	Type	Default value	Description
Logic	TRepeat	Duration	00:00:00.00	High range of the output signal corresponding to the channel.

Range

The table describes the **Range** parameters of the `$AnalogOutput` template that you can configure:

Element Name	Name	Type	Default value	Description
Logic	HighRaw	Short	2047	High range of the output signal corresponding to the channel.
	LowRaw	Short	0	Low range of the output signal corresponding to the channel.

For a description of the configurable parameters of the other elements of **LocalPanel** element, refer to the corresponding Control facet template that is indicated in the composition of the `$AnalogOutput` template.

Composition

The `$AnalogOutput` template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$AnalogOutput` control module and the corresponding facet, which implements the service:

Control services (\$AnalogOutput_UC)		Corresponding facet template	Control service description	Associated Supervi- sion services (\$Ana- logOut- put_CS) *	Corresponding facet template	Supervision service description
Core services						
AnalogOutput	Logic	\$AOUTPUT_UL	Refer to the AOUTPUT DFB**.	AnalogOut- put Tags	\$AOUTPUT_CD	Refer to Analog Output Manage- ment***.
	-	-	-	Analog output genies	\$harightdown- valve3_10_CG	
					\$havalve2_10_ CG	
					\$vaupright- valve3_10_CG	
					\$vavalve2_10_ CG	
Optional I/O signal						
AOSignal		\$AOSignal_UL	Refer to the AOSIGNALCOND FCB**.	-	-	-
Optional services						
Interlocks *		\$CONDSUM1_UL	Refer to the CONDSUM1 DFB**.	Interloc- kTags *	\$CONDSUM1_CD	Refer to Analog Output Manage- ment.
LocalPanel	Logic	\$AOUTPUTLP_UL	Refer to the AOUTPUTLP DFB**.	LocalPanel- Tags	\$AOUTPUTLP_CD	

Control services (\$AnalogOutput_UC)		Corresponding facet template	Control service description	Associated Supervi- sion services (\$Ana- logOut- put_CS) *	Corresponding facet template	Supervision service description
	LPMoDe- DISignal	\$DISignal_UL	Refer to the LPSignal pin of AOUTPUTLP DFB**.	—	—	
	ZeroDISig- nal	\$DISignal_UL	Refer to the ZEROSignal pin of AOUTPUTLP DFB**.	—	—	
	CSDISig- nal	\$DISignal_UL	Refer to the CSSignal pin of AOUTPUTLP DFB**.	—	—	
	OpenDI- Signal	\$DISignal_UL	Refer to the OpenSignal pin of AOUTPUTLP DFB**.	—	—	
	CloseDI- Signal	\$DISignal_UL	Refer to the CloseSignal pin of AOUTPUTLP DFB**.	—	—	
	PBLight- OPSignal	\$DOSignal_UL	Refer to the PBLightSignal pin of AOUTPUTLP DFB**.	—	—	
* The service is activated by default.						
** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual						
*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
ILCK	\$Bool/Ref	EDT interface
RSP	\$Real/Ref	EDT interface
SPrange	\$Range/Ref	EDT interface
RSPRanged	\$RangedValue/Ref	EDT interface
RSPRangedwTrack	\$RangedValueWithTracking/Ref	EDT interface
CHOUT	\$Int/Def	EDT interface
OP	\$Real/Def	EDT interface
OPQ	\$ASignalwithQuality/Def	Links to process object
AORange	\$Range/Def	EDT interface
IntVar	\$Int/Def	EDT interface
AOutputName	\$AOUTPUT_IF/Def	Instance name concatenated with control block name

\$AnalogCalc - Analog Calculation

General Description

The \$AnalogCalc process control module template is used to perform calculations on an analog signal.

The process template can perform the following calculations:

- Derivative rate of change (ROC)
- Totalizing (integral with the trapezoidal rule)
- Average

- Minimum
- Maximum

The template stores up to 20 samples of the measurement (present value *PV*), and calculations are performed on them.

Common Links

You can link this template with the `$AnalogInput` process template, page 26 to perform calculations on previously scaled analog inputs.

In turn, the template can provide calculated data to `$AnalogInput` process templates (configured with an external *PV* input), providing Supervision interfaces during operation. It allows representing these calculations graphically and using alarm functions.

Parameters

Configuration

The table describes the **Configuration** parameters of the `$AnalogCalc` template that you can configure:

Element Name	Name	Type	Default value	Description
Logic	ScanRate	Duration	00:00:01	Allows you to configure the sample acquisition time. The DFB takes a sample each time a time equal to or greater than the one configured for this input has elapsed. For calculation purposes, the time elapsed between one DFB execution and another is taken into account.
	NumberOfSamples	UnsignedShort	20	Maximum number of samples that need to be stored and that are used in the calculations.
	ROCCoefficient	Float	1.0	Conversion factor applied to the derivative calculation output. Allows the frequency to be calculated in the desired engineering units.
	TotalCoefficient	Float	1.0	Conversion factor applied to the totalizing calculation output. Allows the total to be calculated in the desired engineering units.

Composition

The `$AnalogCalc` template references composite and facet templates, which provide the core Control services.

The table describes the services that are available from the `$AnalogCalc` control module and the corresponding facet, which implements the service:

Control services	Composite template	Corresponding facet template	Control service description
Logic	<code>\$ACALC_UC</code>	<code>\$ACALC_UL</code>	Refer to the <code>ACALC</code> DFB (see <i>EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual</i>).

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
PV	\$Real/Ref	EDT interface
AcalcName	\$ACALC_IF/Def	Instance name concatenated with control block name

\$DigitalCalc - Digital Calculation

General Description

The \$DigitalCalc process control module template is used to perform calculations based on a digital signal. The calculations performed are frequency and totalizing calculations.

The template stores up to 20 samples with rising edges in the measurement (present value **PV**) based on which the calculations are performed.

Parameters

Configuration

The table describes the **Configuration** parameters of the \$DigitalCalc template that you can configure:

Element Name	Name	Type	Default value	Description
Logic	NumberOfSamples	UnsignedShort	20	Maximum number of samples that need to be stored and that are used in the calculations.
	FrequencyCoefficient	Float	1.0	Conversion factor applied to the frequency calculation. Allows the frequency to be calculated in the desired engineering units (for example, RPM).
	TotalCoefficient	Float	1.0	Conversion factor applied to the totalizing calculation output. Allows the total to be calculated in the desired engineering units.

Composition

The \$DigitalCalc template references composite and facet templates, which provide the core Control services.

The table describes the services that are available from the \$DigitalCalc control module and the corresponding facet, which implements the service:

Control services	Composite template	Corresponding facet template	Control service description
Logic	\$DCALC_UC	\$DCALC_UL	Refer to the DCALC DFB (see EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual).

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
PV	\$Bool/Ref	EDT interface
DCalcName	\$DCALC_IF/Def	Instance name concatenated with control block name

\$DigitalInput - Digital Input Conditioning

General

The \$DigitalInput process control module template is used to condition a digital signal usually coming from a physical input.

The process template provides timing functions for the connection and/or disconnection, simulation, and can consider the signal status as an alarm function.

Parameters

Configuration

The table describes the **Configuration** parameters of the \$DigitalInput template that you can configure:

Element Name	Name	Type	Default value	Description
Logic	BadPV	Boolean	1	1 = Sets the output signal (<i>PV</i>) to 1 when the channel is not operational.
	Watch	Enum	Refinement	Allows enabling (1) or disabling (0) the detected alarm monitoring depending on the dynamic conditions of the process. You can configure the input in 1 of the following ways: <ul style="list-style-type: none"> • Interface: Allows you to connect an interface of type \$Bool to the WatchDI interface of the instance. • Refinement: You can enable/disable the alarm monitoring function during refinement. • Never: The alarm monitoring function is disabled. • Always: The alarm monitoring function is enabled.
	AlarmST	Enum	True Value	Defines if <i>PV</i> value needs to be considered as a detected alarm. You can configure the input in 1 of the following ways: <ul style="list-style-type: none"> • Interface: Allows you to connect an interface of type \$Bool to the WatchSTDI interface of the instance. • Refinement: You can set the value during refinement. • False Value: The function is disabled (0). • True Value: The function is enabled (1).

Time

The table describes the **Time** parameters of the `$DigitalInput` template that you can configure:

Element Name	Name	Type	Default value	Description
Logic	TONSP	Duration	00:00:00	Timing set-point for digital input connection <i>CHIN</i>) in msec (0 to deactivate timing).
	TOFFSP	Duration	00:00:00	Timing set-point for digital input disconnection <i>CHIN</i>) in msec (0 to deactivate timing).

For a description of the configurable parameters of the optional **DISignal** element, refer to `DISignal_UL` (see *EcoStruxure™ Process Expert*, Foundation Application Templates, User Guide).

Composition

The `$DigitalInput` template references composite and facet templates, which provide the following services:

Control	Core service, plus additional optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$DigitalInput` control module and the corresponding facet, which implements the service:

Control services (\$DigitalInput_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$DigitalInput_CS)*	Corresponding facet template	Supervision service description
Core service						
DInput	Logic	\$DINPUT_UL	Refer to the DINPUT DFB**.	Tags	\$DINPUT_CD	Refer to Digital Input Management***.
	—	—	—	Digital input genies	\$ARROW_CG	
					\$HHVALVE2_CG	
					\$VHVALVE2_CG	
					\$TEMP_CG	
					\$HHVALVE3_CG	
					\$VHVALVE3_CG	
					\$PRES_CG	
					\$XS_CG	
					\$SWITH_CG	
					\$PE_CG	
\$XS2_CG						
Optional I/O signal						
DISignal		\$DISignal_UL	Refer to the <i>DISignalCond1</i> EFB (see EcoStruxure™ Process Expert, Foundation Control Services User Guide).	—	—	
Optional service						

Control services (\$DigitalInput_UC)	Corresponding facet template	Control service description	Associated Supervision services (\$DigitalIn- put_CS) *	Corresponding facet template	Supervision service description
Maintenance	\$DEVMNT_UL	Refer to the DEVMNT DFB**.	Maintenance Tags	\$DEVMNT_CD	
<p>* The service is activated by default.</p> <p>** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual</p> <p>*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual</p>					

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
PV	\$Bool/Def	EDT interface
PVQ	\$DSignalwithQuality/Def	Links to a process object
BoolVar	\$Bool/Ref	EDT interface
DINPUTSignalExt	\$DINPUTSignal/PO	Links to a device object
DInputName	\$DINPUT_IF/Def	Instance name concatenated with control block name

\$DigitalOutput - Digital Output

General Description

The \$DigitalOutput process control module template is used to condition a digital signal usually associated with a physical output.

The process template provides interlocking, diagnosis, and owner management functions that you can configure in the **Properties** window of the instance (see EcoStruxure™ Process Expert, User Guide).

Parameters

For a description of the configurable parameters of the **DOSignal** element, refer to the corresponding Control facet template that is indicated in the composition of the \$DigitalOutput template.

Composition

The \$DigitalOutput template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the \$DigitalOutput control module and the corresponding facet, which implements the service:

Control services (\$DigitalOutput_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$Digital- Output_CS)	Corre- sponding facet template	Supervision service description
Core services						
DOutput	Logic	\$DOUTPUT_UL	Refer to the DOUTPUT DFB**.	DOutputTags	\$DOUT- PUT_CD	Refer to Digital Output Manage- ment**.
	—	—	—	Digital output genie	\$DOut- put_CG	
Optional I/O signal						
DOSignal		\$DOSignal_UL	Refer to the DOSIGNAL DFB**.	—	—	—
Optional services						
Interlocks		\$CONDSUM1_UL	Refer to the CONDSUM1 DFB**.	InterlockTags	\$CON- DSUM1_CD	
Maintenance		\$DEVMNT_UL	Refer to the DEVMNT DFB**.	MainteTags *	\$DEVMNT_ CD	
* The service is activated by default.						
** See EcoStruxure™ Process Expert - General Purpose Library Classic Equipment Module Control Services Reference Manual						
*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
ILCK	\$Bool/Ref	EDT interface
SP	\$Real/Ref	EDT interface
SPQ	\$DSignalwithQuality/Ref	Links to a process object
CHOUTwQuality	\$DOUTPUTSignal/PO	Links to a device object
RSP	\$Bool/Ref	EDT interface
CHOUT	\$Bool/Def	EDT interface
DOutputSignal	\$DOUTPUTSignal/DO	Links to a device object
DOUTPUTSignalExt	\$DOUTPUTSignal/DO	Links to a device object
DOutputName	\$DOUTPUT_IF/Def	Instance name concatenated with control block name

\$MAnalogInput1 - Multiple Analog Input with Configurable Range

General Description

The \$MAnalogInput1 process control module template is used to condition up to 4 analog signals normally coming from physical inputs and to select one of them based on the chosen selection criterion. The range can be configured from the Supervision runtime.

The process template provides linear scaling, cut-off, diagnosis, and simulation functions that you can configure in the **Properties** window of the instance (see EcoStruxure™ Process Expert, User Guide).

Common Links

Using the **Links Editor** (see EcoStruxure™ Process Expert, User Guide), you can link this template with:

- The `$AALARM_UC` process facet template. This allows you to incorporate functions for the evaluation of detected alarms associated with the measurement.
- The `$AnalogCalc` process template, page 33. This allows you to incorporate calculations on the analog input.

Composition

The `$MAnalogInput1` template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$MAnalogInput1` control module and the corresponding facet, which implements the service:

Control services (\$MultipleAInput1_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$Multi- pleAIn- put1_CS) *	Corre- sponding facet template	Supervi- sion service description
Core services						
MultipleAnalogInput1	Logic	\$MAINPUT1_UL	Refer to the MAINPUT1 DFB**.	MultipleAln- put1Tags	\$MAIN- PUT1_CD	Refer to Multiple Analog Input Manage- ment***.
	–	–	–	Multiple input1 genies	\$MAI1RP- VA_CG	
Optional I/O signals						
AnalogInputSignal1		\$AISignal_UL	Refer to the <i>AI/SignalCondReal1</i> and <i>AI/SignalCond1</i> EFB (see EcoStruxure™ Process Expert, Foundation Control Services User Guide).	–	–	–
AnalogInputSignal2		\$AISignal_UL		–	–	–
AnalogInputSignal3		\$AISignal_UL		–	–	–
AnalogInputSignal4		\$AISignal_UL		–	–	–
* The service is activated by default.						
** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual						
*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
PV	<code>\$Real/Def</code>	EDT interface
PVRanged	<code>\$RangedValue/Def</code>	EDT interface
IntVar (AnalogInputSignal1)	<code>\$Int/Ref</code>	EDT interface
IntVar (AnalogInputSignal2)	<code>\$Int/Ref</code>	EDT interface
IntVar (AnalogInputSignal3)	<code>\$Int/Ref</code>	EDT interface
IntVar (AnalogInputSignal4)	<code>\$Int/Ref</code>	EDT interface

Interface identifier	Type/role	Description
AINPUTSignalExt (AnalogInputSignal1)	\$AINPUTSignal/PO	Links to a device object
AINPUTSignalExt (AnalogInputSignal2)	\$AINPUTSignal/PO	Links to a device object
AINPUTSignalExt (AnalogInputSignal3)	\$AINPUTSignal/PO	Links to a device object
AINPUTSignalExt (AnalogInputSignal4)	\$AINPUTSignal/PO	Links to a device object
MAInput1Name	\$MAINPUT1_IF/Def	Instance name concatenated with control block name

\$Range - Control Expert Analog Range

General Description

The \$Range process control module template is used to configure range values for variables such as PV, SP, and so on.

This process template helps to configure parameters such as high value, low value, engineering unit, format, and memory address.

Composition

The \$Range template references composite and facet templates, which provide the core Control services.

The table describes the services that are available from the \$Range control module and the corresponding facet, which implements the service:

Control services	Composite template	Corresponding facet template	Control service description
Logic	—	\$Range_UL	Range DDT gets generated in the Control Participant in the section for variables.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
AnalogRange	\$Range/Def	EDT interface

\$TOTAL - Totalizer

General Description

The \$TOTAL process control module template is used to perform totalization calculation based on any one of the three inputs (analog signal, digital pulse and counter).

Common Links

In **Links Editor**, user can link this template to perform totalizer calculation based on the following inputs:

- *PVFLOW* using the templates that support \$Real interface

- *PVFLOWQ* using the templates that support *\$ASignalwithQuality* interface.
- *PVPULSE* using the templates that support *\$Bool* interface.
- *PVPULSEQ* using the templates that support *\$DSignalwithQuality* interface.
- *CounterSignalUDINT* using the templates that support *\$UDInt* interface.
- *CounterSignalUDINText* using the templates that support *\$CNTRSignalwQuality* interface.
- *CounterSignalUINT* using the templates that support *\$UInt* interface.
- *CounterSignalUINText* using the templates that support *\$CNTRSignalwQuality* interface.

NOTE:

- If any one of the four inputs are linked with templates that support the respective interface, then the other three inputs along with signal object interface of *\$TOTAL* template gets disabled.
- If none of the inputs are linked then, signal object interface of *\$TOTAL* template will be available for hardware mapping.

Parameter Configuration

The table describes the configuration parameters of the *\$TOTAL* template that user can configure.

Element Name	Category Name	Identifier Name	Type	Default value	Description
Logic	Configuration	K Coefficient	Float	1.0	Totalized conversion coefficient applied to totalizing calculations. This is one time configuration.
		AutoStart	Boolean	False	Whether the operation to be started automatically once reaching <i>IDLE</i> state.
		AutoReset	Boolean	False	Whether the operation to be started automatically once reaching <i>STOPPED</i> state.
	Configuration flow	Scanrate	Duration	00:00:01.000	Time interval at which sample is considered for totalization.
	Configuration counter	Roll Over Maximum Count	Unsigned-Long	0	Enter the maximum roll over value of counter module. NOTE: If a value is entered other than 0, only then the variable will be created in the <i>ROLLOVERMAX</i> input pin of the <i>TOTAL</i> DFB.
Supervision	Configuration	LASTTOTAL_Deadband	Float	0.0	Deadband of last total
		TOTAL_Deadband	Float	0.0	Deadband of total
		SP_Deadband	Float	0.0	Deadband of setpoint
		TOTAL_EngUnits	String	EU	Engineering unit of total
		TOTAL_Format	String	####.#EU	Format of total
		SP_EngUnits	String	EU	Engineering unit of setpoint
		SP_Format	String	####.#EU	Format of setpoint
	Historize	Historize_HELDFORError	Boolean	False	Enable Historize held for detected error alarm
		Historize_SP	Boolean	False	Enable Historize setpoint
		Historize_Total	Boolean	False	Enable Historize total
		Historize_LASTTOTAL	Boolean	False	Enable Historize last total

Element Name	Category Name	Identifier Name	Type	Default value	Description
		SP_T_Historize	Boolean	True	Enable Historize setpoint value trend
		Total_T_Historize	Boolean	True	Enable Historize total value trend
Genies	Configuration	TOTAL_HOLD	Boolean	False	Hide Hold command button
		TOTAL_RESET	Boolean	False	Hide Reset command button
		TOTAL_RESTART	Boolean	False	Hide Restart command button
		TOTAL_STARTT	Boolean	False	Hide Start command button
		TOTAL_STOP	Boolean	False	Hide Stop command button

Composition

The \$TOTAL template references composite and facet templates, which provide the following services:

Control	Core services.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the \$TOTAL control module and the corresponding facet, which implements the service:

Control \$Totalizer_UC		Corresponding facet template	Control Service description	Associated Supervision services (\$TOTAL_CS) *	Corre- sponding facet template	Supervision Service description
Core services						
TOTAL	LOGIC	\$TOTAL_UC	Refer to the TOTAL DFB*.	TOTAL-Tags	\$TOTAL_CD	Refer to TOTAL**.
	—	—	—	TOTAL-Genies	\$TOTAL_10_CG \$Total-WithBut- tons_10_CG \$SPTotal_10_CG \$SPTotal-WithBut- tons_10_CG	Refer to TOTAL**.
Optional services						
Failures	LOGIC	\$CONDSUM_UL	Refer to the CONDSUM DFB*.	Failure tags	\$CONDSUM_CD	Refer to TOTAL**.
I/O Signals						
Fail	LOGIC	DISignal_UL	Refer to the FAIL pin of TOTAL DFB*.	Failure tags	\$CONDSUM_CD	Refer to TOTAL**.
* See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual						
** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
PVFLOW	\$Real/Ref	EDT Interface
PVFLOWQ	\$ASignalwithQuality/Ref	Nested interface consisting of \$Real and \$Bool
PVPULSE	\$Bool/Ref	EDT Interface
PVPULSEQ	\$DSignalwithQuality/Ref	Nested interface consisting of \$Bool and \$Bool
CounterSignalUDINT	\$UDInt/Ref	EDT Interface
CounterSignalUDINExt	\$CNTRSignalwQuality/Ref	DDT Interface
CounterSignalUINT	\$UInt/Ref	EDT Interface
CounterSignalUINExt	\$CNTRSignalwQuality/Ref	DDT Interface
RSP	\$Real/Ref	EDT Interface
FAIL	\$Bool/Ref	EDT Interface
RSTART	\$Bool/Ref	EDT Interface
RHOLD	\$Bool/Ref	EDT Interface
RRESTART	\$Bool/Ref	EDT Interface
RSTOP	\$Bool/Ref	EDT Interface
RRESET	\$Bool/Ref	EDT Interface
TOTAL	\$Real/Def	EDT Interface
LASTTOTAL	\$Real/Def	EDT Interface
SETPOINT	\$Real/Def	EDT Interface
TOTALName	\$TOTAL_IF/Def	Instance name concatenated with control block name

On/Off Device Control

Overview

This chapter describes the functionality of the on/off device control process templates and their composition.

These templates do not reflect any specific installation.

⚠ WARNING

LOSS OF CONTROL

- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.¹
- Test each implementation of a system for proper operation before placing it into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

¹ For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

\$HandValve - Manual Valve

General Description

The main objective of the \$HandValve process control module template is to manage valves without actuators and with one or two limit switches or any other type of positioner with two final positions.

Common Links

You can link this process template with \$DigitalInput process template, page 36 by using the **Links Editor** (see EcoStruxure™ Process Expert, User Guide). It allows you to incorporate conditioning functions for the digital signals of the limit switches.

Parameter

Configuration

The table describes the **Configuration** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default Value	Description
-	Open	Negate	Boolean	False	True = Negates the input or output signal
	Close				
	Open	DISignal-Name	String	Null	Name used for the generated EFB and variable.
	Close				

The `$HandValve` template features no configurable parameters other than the general `$System` parameters (`$Name`, `$Description`, and `$Area`).

Composition

The `$HandValve` template references composite and facet templates, which provide the following services:

Control	Core services.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$HandValve` control module and the corresponding facet, which implements the service:

Control services (\$HandValve_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$Hand- Valve_CS) *	Corre- sponding facet template	Supervi- sion service descrip- tion
HandValve	Logic	\$HValve_UL	Refer to the HVALVE DFB**.	HandValve- Tags	\$HValve_ CD	Refer to Hand Valve Manage- ment***.
	—	—	—	Hand valve genies	\$HVal- veHH- VALVE2_ CG	
					\$HVal- veVH- VALVE2_ CG	
Optional I/O signals						
OpenSignal *		\$DISignal_UL	Refer to the ZSH pin of HVALVE DFB**.	—	—	—
CloseSignal *		\$DISignal_UL	Refer to the ZSL pin of HVALVE DFB**.	—	—	—
* The service is activated by default.						
** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual						
*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
OpenPositionFB	\$Bool/Ref	EDT interface
ClosePositionFB	\$Bool/Ref	EDT interface
HValveName	\$HVALVName/Def	Instance name concatenated with control block name

\$Motor - 1 Speed/1 Rotation Direction Motor

General Description

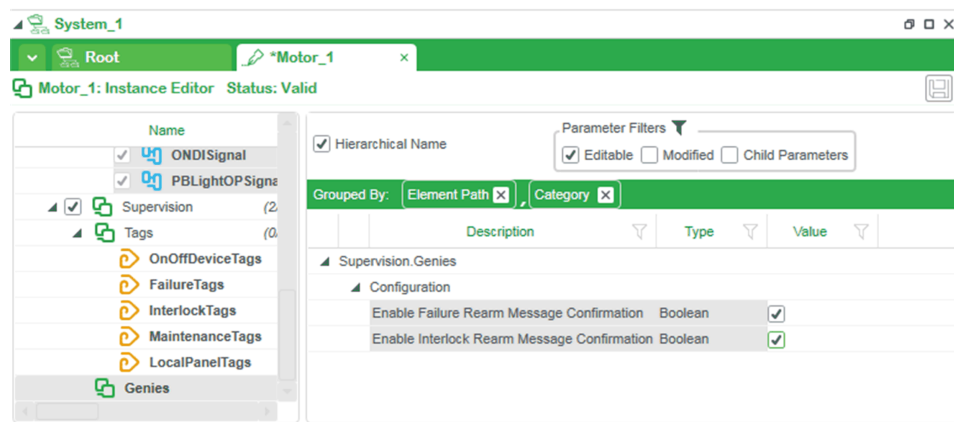
The \$Motor process control module template is used to manage on/off devices such as on/off actuated valves and discrete motors (without variable speed).

The template allows managing the associated devices from the sequential control, the continuous control, and/or the Supervision system. It depends on the configuration of the associated devices and the system requirements.

Optional Rearm Confirmation

The template allows you to display a dialog box during operation when you reset the device after interlock s or abnormal conditions have been cleared. The dialog box requires that you confirm the reset of the device.

To display the dialog box in runtime, select the [parameter](#), [page 47](#) of the **Genies** element (Supervision services) in the **Properties** window of the instance.



Parameters

Configuration

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- The *CONFREARMEN* pin has to be set to 1 for the *OP* pin to become 0, under non-confirmed operations.
- The *FAILREARMEN* pin has to be set to 1 for the *OP* pin to become 0, under detected fail conditions.
- Changes to the configuration of these parameters have to be performed by competent personnel only.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

The table describes the **Configuration** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
Logic		ConfRear-mEN	Boolean	1	<ul style="list-style-type: none"> • 1 = Withdraws the output <i>OP</i> when a non-confirmed operation is detected (<i>ALARM</i> = 1) and requires a mandatory manual resetting, once the <i>ALARM</i> condition disappears. • 0 = The output <i>OP</i> remains in its previous state when a non-confirmed operation is detected (<i>ALARM</i> = 1) and does not need manual resetting, once the <i>ALARM</i> condition disappears.
Logic		FailRear-mEN	Boolean	1	<ul style="list-style-type: none"> • 1 = Withdraws the output <i>OP</i> upon a detected failure condition (<i>FAILD</i> = 1) and requires a mandatory manual resetting, once the <i>FAIL</i> condition disappears. • 0 = The output <i>OP</i> remains in its previous state upon a detected failure condition (<i>FAILD</i> = 1) and does not need manual resetting, once the <i>FAIL</i> condition disappears.
Running		Negate	Boolean	False	True = Negates the input or output signal.
Fail					
ExternalControl					
OPDOSignal					
LocalPanel	LPModeSignal				
	ZERODISignal				
	CSDISignal				
	OFFDISignal				
	ONDISignal				
	PBLightOPSignal				
Running		xSignal-Name	String	Null	Name used for the generated EFB/DFB and variables. x represents the element/child element name.
Fail					
ExternalControl					
LocalPanel	LPModeSignal				
	ZERODISignal				
	CSDISignal				
	OFFDISignal				
	ONDISignal				
	PBLightOPSignal				

Element Name	Child Element Name	Name	Type	Default value	Description
OPDOSignal		OPSignal-Name	String	Null	Name used for the generated DFB and variables.
Genies		Failure Rearm Confirmation	Boolean	0	0 = The reset command acts on the device with no confirmation being required. 1 = Displays a dialog box in runtime, which requires you to confirm the reset of the device.
<p>NOTE: Values of boolean parameters are set by using check boxes:</p> <ul style="list-style-type: none"> Selected = True Cleared = False 					

Basic Alarm Configuration

The table describes the **Basic Alarm Configuration** parameters that you can configure:

Element Name	Name	Type	Default value	Description
OnOffDeviceTags	Name_X	String	Null	Name_X : Name of the physical device associated with the alarm. Corresponds to the <i>Alarm Name</i> property of advanced alarms of the Supervision Participant.
MaintenanceTags				
LocalPanelTags				
OnOffDeviceTags	Help_X	String	Null	Help_X : Name of the graphics page that opens when the help function is called. Corresponds to the <i>Help</i> property of advanced alarms of the Supervision Participant.
MaintenanceTags				
LocalPanelTags				
OnOffDeviceTags	Delay_X	Duration	00:00:00	Delay_X : Alarm delay period. Corresponds to the <i>Delay</i> property of advanced alarms of the Supervision Participant.
MaintenanceTags				
LocalPanelTags				
OnOffDeviceTags	Category_X	String	Label with the syntax: <i>SGC_CAT_<type of component>_<type of alarm></i>	Alarm category label or number. Corresponds to the <i>Category</i> property of advanced alarms of the Supervision Participant.
MaintenanceTags				
LocalPanelTags				
NOTE: For advanced alarms that are process events <i><type of component></i> and <i><type of alarm></i> are replaced by EVENT , for example SGC_CAT_EVENT .				

X	<p>Suffix of the variable tag name or advanced alarm tag that is managed by the corresponding Supervision components. For example, Name_ALARM, Paging_ALARM, and Historize_ALARM correspond to the <i>CM Name_DEVCTL_ALARM</i> advanced alarm tag managed by the <i>DEVCTL</i> Control resource.</p> <p>Refer to the description of tags of the Supervision component.</p>
---	---

Advanced Alarm Configuration

The table describes the **Advanced Alarm Configuration** parameters that you can configure:

Element Name	Name	Type	Default value	Description		
OnOffDeviceTags	Privilege_X	String	Label with the syntax: <i>SGC_PRIV_<type of component>_<type of action></i>	Privilege level required to acknowledge or disable the alarm. Corresponds to the <i>Privilege</i> property of advanced alarms of the Supervision Participant.		
MaintenanceTags						
LocalPanelTags				For example, for Privilege_ALARM , the label is SGC_PRIV_D_ALARM . Refer to Privilege Levels and Labels (see EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual). NOTE: For advanced alarms that are process events <i><type of component></i> and <i><type of action></i> are replaced by EVENT , for example SGC_PRIV_EVENT .		
OnOffDeviceTags	Custom1_X to Custom8_X	String	Null	Custom1_X to Custom8_X: String to filter active alarms (64 characters maximum). Corresponds to the <i>Custom 1 to Custom 8</i> property of advanced alarms of the Supervision Participant.		
MaintenanceTags				Refer to <i>Advanced Alarm Properties</i> in the help of the Supervision Participant for details.		
LocalPanelTags						
OnOffDeviceTags	Paging_X	Boolean	0	Paging_X: Read/write property, which indicates the paging status of the alarm. Corresponds to the <i>Paging</i> property of advanced alarms of the Supervision Participant.		
MaintenanceTags				1 = The alarm is paged.		
LocalPanelTags						
OnOffDeviceTags	PagingG_X	String	Null	PagingG_X: A read-only text string that indicates the paging group to which the alarm belongs (80 characters maximum). Corresponds to the <i>Paging Group</i> property of advanced alarms of the Supervision Participant.		
MaintenanceTags						
LocalPanelTags						
<table><tr><td>X</td><td>Suffix of the variable tag name or advanced alarm tag that is managed by the corresponding Supervision components. For example, Name_ALARM, Paging_ALARM, and Historize_ALARM correspond to the <i>CM Name_DEVCTL_ALARM</i> advanced alarm tag managed by the <i>DEVCTL</i> Control resource. Refer to the description of tags of the Supervision component.</td></tr></table>					X	Suffix of the variable tag name or advanced alarm tag that is managed by the corresponding Supervision components. For example, Name_ALARM , Paging_ALARM , and Historize_ALARM correspond to the <i>CM Name_DEVCTL_ALARM</i> advanced alarm tag managed by the <i>DEVCTL</i> Control resource. Refer to the description of tags of the Supervision component.
X	Suffix of the variable tag name or advanced alarm tag that is managed by the corresponding Supervision components. For example, Name_ALARM , Paging_ALARM , and Historize_ALARM correspond to the <i>CM Name_DEVCTL_ALARM</i> advanced alarm tag managed by the <i>DEVCTL</i> Control resource. Refer to the description of tags of the Supervision component.					

Historize

The table describes the **Historize** parameters that you can configure:

Element Name	Name	Type	Default value	Description
OnOffDevice-Tags	Historize_X	Boolean	Refer to the properties of the template instance for the default value of these parameters.	Corresponds to the <i>Historize</i> property of advanced alarms of the Supervision Participant.
FailureTags				1 = Sets the <i>Historize</i> property of the corresponding tag to <i>True</i> .
InterlockTags				Refer to <i>Advanced Alarm Properties</i> in the help of the Supervision Participant for details.
Maintenance-Tags				
LocalPanelTags				
X		Suffix of the variable tag name or advanced alarm tag that is managed by the corresponding Supervision components. For example, Name_ALARM , Paging_ALARM , and Historize_ALARM correspond to the <i>CM Name_DEVCTL_ALARM</i> advanced alarm tag managed by the <i>DEVCTL</i> Control resource.		
		Refer to the description of tags of the Supervision component.		

Message

The table describes the **Message** parameters that you can configure:

Element Name	Name	Type	Default value	Description
FailureTags	CondYY_Message	String	Null	Description that appears on the diagnostic information or interlock condition tab of the faceplate when the corresponding detected failure or interlock condition arises. YY corresponds to the number of the condition that is detected: <ul style="list-style-type: none">01 to 15 for diagnostic information01 to 07 for interlock conditions
InterlockTags	ILCKCondYY_Message (interlocks)			

X	Suffix of the variable tag name or advanced alarm tag that is managed by the corresponding Supervision components. For example, Historize_CONDW correspond to the <i>CM Name_FC_CONDSUM_CONDW</i> advanced alarm tag managed by the <i>CONDSUM</i> Control resource. Refer to the description of tags of the Supervision component.
---	---

Variable Tag Disable

The table describes the **Variable Tag Disable** parameters that you can configure:

Element Name	Name	Type	Default value	Description
FailureTags	BYPASSWordSelection	Boolean	0	<ul style="list-style-type: none">For FailureTags: 1 = Disables the possibility to bypass each detected failure condition.For InterlockTags: 1 = Disables the possibility to bypass each interlock condition.
InterlockTags				
FailureTags	REARMREQWordSelection	Boolean	0	<ul style="list-style-type: none">For FailureTags: 1 = Disables the requirement to reset the device manually after an individual detected failure condition is cleared.For InterlockTags: 1 = Disables the requirement to reset the device manually after an individual interlock condition is cleared.
InterlockTags				
X	Suffix of the variable tag name or advanced alarm tag that is managed by the corresponding Supervision components. For example, Historize_CONDW correspond to the <i>CM Name_FC_CONDSUM_CONDW</i> advanced alarm tag managed by the <i>CONDSUM</i> Control resource. Refer to the description of tags of the Supervision component.			

Composition

The `$Motor` template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$Motor` control module and the corresponding facet, which implements the service:

Control services (\$Motor_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$Motor_CS) *	Correspond- ing facet template	Supervision service description
Core services						
Motor	Logic	\$DEVCTL_UL	Refer to the DEVCTL DFB**.	OnOffDeviceTags *	\$DEVCTL_CD	Refer to On/Off Motor Management***.
	—	—	—	Motor genres	\$HLMOTOR_CG	
					\$HRMOTOR_CG	
					\$VMOTOR_CG	
					\$VUMOTOR_CG	
	—	—	—	Pump genres	\$LCPUMP_CG	
					\$RCPUMP_CG	
Optional I/O signals						
ExternalControl		\$DISignal_UL	Refer to the EXTCTLD pin of DEVCTL DFB**.	—	—	—
Running		\$DISignal_UL	Refer to the ZSH pin of DEVCTL DFB**.	—	—	—
OPDO Signal		\$DOSignal_UL	Refer to the OP pin of DEVCTL DFB**.	—	—	—
Fail		\$DISignal_UL	Refer to the FAIL pin of DEVCTL DFB**.	—	—	—
Optional services						
Interlocks *		\$CONDSUM1_UL	Refer to the CONDSUM1 DFB**.	InterlockTags *	\$CONDSUM1_CD	Refer to On/Off Motor Management.
Failures *		\$CONDSUM_UL	Refer to the CONDSUM DFB**.	FailureTags *	\$CONDSUM_CD	
Maintenance		\$DEVMNT_UL	Refer to the DEVMNT DFB**.	MaintenanceTags	\$DEVMNT_CD	
Local-Panel	Logic	\$DEVLP_UL	Refer to the DEVLP DFB**.	LocalPanelTags	\$DEVLP_CD	
	LPMo- deDI- Signal	\$DISignal_UL	Refer to the LPSignal pin of DEVLP DFB**.	—	—	
	ZeroDI- Signal	\$DISignal_UL	Refer to the ZEROSignal pin of DEVLP DFB**.	—	—	
	CSDI- Signal	\$DISignal_UL	Refer to the CSSignal pin of DEVLP DFB**.	—	—	
	OFFDI- Signal	\$DISignal_UL	Refer to the OFFSignal pin of DEVLP DFB**.	—	—	
	ONDI- Signal	\$DISignal_UL	Refer to the ONSignal pin of DEVLP DFB**.	—	—	
	PBLight- OPSig- nal	\$DOSignal_UL	Refer to the PBLightSignal pin of DEVLP DFB**.	—	—	
* The service is activated by default.						
** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual						
*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
ILCK	\$Bool/Ref	EDT interface
MotorOP	\$Bool/Def	EDT interface
Dev1S1D	\$DEV1S1D/PO	Links to a process object
RemoteSetpoint	\$Bool/Ref	EDT interface
ZSHPOS	\$Bool/Def	High limit position reached.
ZSLPOS	\$Bool/Def	Low limit position reached.
DEVCTLName	\$DEVCTL_IF/Def	Instance name concatenated with control block name

NOTE: The **MotorOP**, **ZSHPOS** and **ZSLPOS** interfaces can be used to achieve dual output functionality using \$DUALOP_UC template.

\$Motor2 - 2 Speed/2 Rotation Directions Motor

General Description

The \$Motor2 process control module template is used to control on/off motors with two rotation directions (forward/reverse) or with two speeds (for example, slow speed and fast speed).

The process template allows the managing associated motors from the sequential control, the continuous control, and/or the Supervision system depending on their configuration and the system needs.

Optional Rearm Confirmation

The template allows you to display a dialog box upon rearming interlock conditions and the detection of abnormal conditions during operation.

To display the dialog box in runtime, select the **Enable Interlock Rearm Message Confirmation** / **Enable Failure Rearm Message Confirmation** parameter of the **Genies** element (Supervision services) in the **Properties** window of the instance.

Parameters

Configuration

⚠ WARNING
<p>UNINTENDED EQUIPMENT OPERATION</p> <ul style="list-style-type: none"> The <i>CONFREARMEN</i> pin has to be set to 1 for the <i>OPx</i> (x = 1 or 2) pin to become 0, under non-confirmed operations. The <i>FAILREARMEN</i> pin has to be set to 1 for the <i>OPx</i> (x = 1 or 2) pin to become 0, under detected fail conditions. Changes to the configuration of these parameters have to be performed by competent personnel only. <p>Failure to follow these instructions can result in death, serious injury, or equipment damage.</p>

The table describes the **Configuration** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
Logic		ConfRearmEN	Boolean	1	<ul style="list-style-type: none"> 1 = Withdraws the output OP_x ($x = 1$ or 2) when a non-confirmed operation is detected ($ALARM = 1$) and requires a mandatory manual resetting, once the $ALARM$ condition disappears. 0 = The output OP_x ($x = 1$ or 2) remains in its previous state when a non-confirmed operation is detected ($ALARM = 1$) and does not need manual resetting, once the $ALARM$ condition disappears.
Logic		FailRearmEN	Boolean	1	<ul style="list-style-type: none"> 1 = Withdraws the output OP_x ($x = 1$ or 2) upon a detected failure condition ($FAILD = 1$) and requires a mandatory manual resetting, once the $FAIL$ condition disappears. 0 = The output OP_x ($x = 1$ or 2) remains in its previous state upon a detected failure condition ($FAILD = 1$) and does not need manual resetting, once the $FAIL$ condition disappears.
ForwardRunningSignal	LocalPanel	Negate	Boolean	False	True = Negates the input or output signal.
ReverseRunningSignal					
ForwardFailSignal					
ReverseFailSignal					
ExternalControl					
OP1DOSignal					
OP2DOSignal					
LPMoDeSignal					
ZERODISignal					
CSDISignal					
OFFDISignal					
ONDISignal					
DirectS1DISignal					
ReverseS2DISignal					
PBLightSignal					
ForwardRunningSignal	LocalPanel	xSignalName	String	Null	Name used for the generated EFB/DFB and variables. x represents the element name.
ReverseRunningSignal					
ForwardFailSignal					
ReverseFailSignal					
ExternalControl					
LPMoDeSignal					
ZERODISignal					
CSDISignal					
OFFDISignal					
ONDISignal					
DirectS1DISignal					
ReverseS2DISignal					
PBLightSignal					

Element Name	Child Element Name	Name	Type	Default value	Description
OP1DOSignal		OPSignalForw-Name	String	Null	Name used for the generated DFB and variables.
OP2DOSignal		OPSignalBack-Name	String	Null	Name used for the generated DFB and variables.
NOTE: Values of boolean parameters are set by using check boxes: <ul style="list-style-type: none"> Selected = True Cleared = False 					

Time

The table describes the **Time** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
Logic		Timeout	Duration	00:00:03	Confirmation for timeout
		ScanTime		00:00:05	Minimum duration taken by an active alarm.

Composition

The \$Motor2 template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the \$Motor2 control module and the corresponding facet, which implements the service:

Control services (\$MOTOR2OnOffDevice_UC)	Corresponding facet template	Control service description	Associated Supervision services (\$MOTOR2OnOffDevice_CS) *	Corresponding facet template	Supervision service description
Core services					
Motor2	Logic	\$MOTOR2_UL	Refer to the MOTOR2 DFB**.	Motor2Tags *	Refer to 2-Speed/2-Rotation Directions Motor Management***.
	—	—	—	Motor genies	
				\$MOTOR2_CD	
				\$HLMOTOR2D_CG	
				\$HLMOTOR2S_CG	
				\$HRMOTOR2D_CG	
				\$HRMOTOR2S_CG	
				\$V MOTOR2D_CG	
				\$V MOTOR2S_CG	
				\$VUMOTOR2D_CG	
				\$VUMOTOR2S_CG	

Control services (\$MOTOR2OnOffDevice_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$MOTOR2OnOffDe- vice_CS) *	Corre- sponding facet template	Supervision service description
Optional I/O signals						
ForwardRunningSignal		\$DISignal_UL	Refer to the ZSH pin of MOTOR2 DFB**.	—	—	—
ReverseRunningSignal		\$DISignal_UL	Refer to the ZSL pin of MOTOR2 DFB**.	—	—	—
ForwardFailSignal		\$DISignal_UL	Refer to the FAIL1 pin of MOTOR2 DFB**.	—	—	—
ReverseFailSignal		\$DISignal_UL	Refer to the FAIL2 pin of MOTOR2 DFB**.	—	—	—
ExternalControl		\$DISignal_UL	Refer to the EXTCTLD pin of MOTOR2 DFB**.	—	—	—
OP1DOSignal		\$DOSignal_UL	Refer to the OP1 pin of MOTOR2 DFB**.	—	—	—
OP2DOSignal		\$DOSignal_UL	Refer to the OP2 pin of MOTOR2 DFB**.	—	—	—
Optional services						
Interlocks*		\$CONDSUM1_UL	Refer to the CONDSUM1 DFB**.	InterlockTags	\$CON- DSUM1_CD	Refer to 2- Speed/2- Rotation Directions Motor Management.
ForwardFailures*		\$CONDSUM_UL	Refer to the CONDSUM DFB**.	FailureTags *	\$CON- DSUM_CD	
ReverseFailures*		\$CONDSUM_UL	Refer to the CONDSUM DFB**.	FailureTags *	\$CON- DSUM_CD	
Maintenance		\$DEVMNT_UL	Refer to the DEVMNT DFB**.	Maintenan- ceTags	\$DEVMNT_ CD	
LocalPanel	Logic	\$MOTOR2LP_UL	Refer to the MOTOR2LP DFB**.	LocalPanel- Tags	\$MO- TOR2LP_ CD	Refer to 2- Speed/2- Rotation Directions Motor Management.
	LPMo- deDISig- nal	\$DISignal_UL	Refer to the LPSignal pin of MOTOR2LP DFB**.	—	—	
	ZERO- DISignal	\$DISignal_UL	Refer to the ZEROSignal pin of MOTOR2LP DFB**.	—	—	
	CSDI- Signal	\$DISignal_UL	Refer to the CSSignal pin of MOTOR2LP DFB**.	—	—	
	OFFDI- Signal	\$DISignal_UL	Refer to the OFFSignal pin of MOTOR2LP DFB**.	—	—	
	Di- rec- tS1DI- Signal	\$DISignal_UL	Refer to the DirectS1Signal pin of MOTOR2LP DFB**.	—	—	
	Rever- seS2DI- Signal	\$DISignal_UL	Refer to the ReverseS1Signal pin of MOTOR2LP DFB**.	—	—	
	PBLight- OPSig- nal	\$DOSignal_UL	Refer to the PBLightSignal pin of MOTOR2LP DFB**.	—	—	
* The service is activated by default.						
** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual						
*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
ILCK	\$Bool/Ref	EDT interface
OP1	\$Bool/Def	EDT interface
OP2	\$Bool/Def	EDT interface
DEV1S2D	\$DEV1S2D/PO	Links to a process object
DEV2S1D	\$DEV2S1D/PO	Links to a process object
RSPSEL	\$Bool/Ref	EDT interface
RSP	\$Bool/Ref	EDT interface
ILCKSEL	\$Bool/Ref	EDT interface
MOTOR2Name	\$MOTOR2_IF/Def	Instance name concatenated with control block name

\$MValve - Motorized Valve Without Positioner

General Description

The \$MValve process control module template is designed to control a motorized valve or a gate, with two limit switches (open-valve and closed-valve) and two-rotational-directions-motor based control.

Optional Rearm Confirmation

The template allows you to display a dialog box upon rearming interlock conditions and the detection of abnormal conditions during operation.

To display the dialog box in runtime, select the **Enable Interlock Rearm Message Confirmation / Enable Failure Rearm Message Confirmation** parameter of the **Genies** element (Supervision services) in the **Properties** window of the instance.

Parameters

Configuration

⚠ WARNING
UNINTENDED EQUIPMENT OPERATION <ul style="list-style-type: none"> The <i>CONFREARMEN</i> pin has to be set to 1 for the <i>SP_MOTOR2</i> pin to become 0, under non-confirmed operations. Changes to the configuration of these parameters have to be performed by competent personnel only. <p>Failure to follow these instructions can result in death, serious injury, or equipment damage.</p>

The table describes the **Configuration** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
Valve	Logic	ConfRearmEN	Boolean	1	<ul style="list-style-type: none">1 = Withdraws the output <i>SP_MOTOR2</i> when a non-confirmed operation is detected (<i>ALARM</i> = 1) and requires a mandatory manual resetting, once the <i>ALARM</i> condition disappears.0 = The output <i>SP_MOTOR2</i> remains in its previous state when a non-confirmed operation is detected (<i>ALARM</i> = 1) and does not need manual resetting, once the <i>ALARM</i> condition disappears.
Motor2					
Valve	Logic	FailRearmEN	Boolean	1	<ul style="list-style-type: none">1 = Withdraws the output <i>SP_MOTOR2</i> upon a detected failure condition (<i>FAILD</i> = 1) and requires a mandatory manual resetting, once the <i>FAIL</i> condition disappears.0 = The output <i>SP_MOTOR2</i> remains in its previous state upon a detected failure condition (<i>FAILD</i> = 1) and does not need manual resetting, once the <i>FAIL</i> condition disappears.
Motor2					
HighLimitSwitch	Logic	BadPV	Boolean	1	1 = Sets the output signal (<i>PV</i>) to 1 when the channel is not operational.
LowLimitSwitch	Logic				
HighLimitSwitch	Logic	AlarmST	Enum	Null	PV state to be considered as alarm.
LowLimitSwitch	Logic				
OP1DOSignal		Negate	Boolean	False	True = Negates the input or output signal.
OP2DOSignal					
HighLimitSwitch	DISignal				
LowLimitSwitch	DISignal				
ForwardRunningSignal					
ReverseRunningSignal					
ForwardFailSignal					
ReverseFailSignal					
ExternalControlSignal					
LocalPanel	LPMoDeDISignal				
	ZERODISignal				
	CSDISignal				
	OffDISignal				
	OpenDISignal				
	CloseDISignal				
	PBLightDOSignal				
ForwardRunningSignal		xSignalName	String	Null	Name used for the generated EFB/DFB and variables. x represents the child element name.
ReverseRunningSignal					
ForwardFailSignal					
ReverseFailSignal					
ExternalControlSignal					
LocalPanel	LPMoDeDISignal				
	ZERODISignal				
	CSDISignal				

Element Name	Child Element Name	Name	Type	Default value	Description
	OffDISignal				
	OpenDISignal				
	CloseDISignal				
	PBLightDOSignal				
HighLimitSwitch	DISignal	Type	Boolean	Null	1 = Includes the quality signal.
LowLimitSwitch	DISignal				
NOTE: Values of boolean parameters are set by using check boxes: <ul style="list-style-type: none"> Selected = True Cleared = False 					

Time

The table describes the **Time** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description	
Valve	Logic	Timeout	Duration	00:00:03	Confirmation for the timeout.	
Motor2	Logic					
Valve	Logic	ScanTime		00:00:05	Minimum duration taken by an active alarm.	
Motor2	Logic					
HighLimitSwitch	Logic	TONSP	Duration	00:00:00	ON timer set point for digital input connection.	
LowLimitSwitch	Logic					
HighLimitSwitch	Logic	TOFFSP				OFF timer set point for digital input disconnection.
LowLimitSwitch	Logic					

Composition

The `$MValve` template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$MValve` control module and the corresponding facet, which implements the service:

Control services (\$MotValved_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$MValved_CS)*	Corresponding facet template	Supervision service description
Core services						
Valve	Logic	\$MVALVED_UL	Refer to the MVALVED DFB**.	ValveTags	\$MOTVALVED_CD	Refer to Discrete Motorized Valve Management***.
Motor2	Logic	\$MOTOR2_UL	Refer to the MOTOR2 DFB**.	Motor2Tags	\$MOTOR2_CD	
Valve	—	—	—	Valve genies	\$MValved_10_CG \$MValved_S_10_CG	

Control services (\$MotValved_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$MValved_CS) *	Corresponding facet template	Supervision service description
I/O signals						
HighLimitS- witch	Logic	\$DINPUT_UL	Refer to the DINPUT DFB**.	ZSHTags	\$DINPUT_CD	
	DI Signal	DISignal_UL	Refer to the <i>DISignalCond1</i> EFB (see EcoStruxure™ Process Expert, Foundation Control Services User Guide).	—	—	—
LowLimitS- witch	Logic	\$DINPUT_UL	Refer to the DINPUT DFB**.	ZSLTags	\$DINPUT_CD	
	DI Signal	DISignal_UL	Refer to the <i>DISignalCond1</i> EFB (see EcoStruxure™ Process Expert, Foundation Control Services User Guide).	—	—	—
Optional I/O signals						
ForwardRunning Signal		\$DISignal_UL	Refer to the ZSH pin of MOTOR2 DFB**.	—	—	—
ReverseRunning Signal		\$DISignal_UL	Refer to the ZSL pin of MOTOR2 DFB**.	—	—	—
ForwardFailSignal		\$DISignal_UL	Refer to the FAIL1 pin of MOTOR2 DFB**.	—	—	—
ReverseFailSignal		\$DISignal_UL	Refer to the FAIL2 pin of MOTOR2 DFB**.	—	—	—
ExternalControl		\$DISignal_UL	Refer to the EXTCTLD pin of MOTOR2 DFB**.	—	—	—
OP1DOSignal		\$DOSignal_UL	Refer to the OP1 pin of MOTOR2 DFB**.	—	—	—
OP2DOSignal		\$DOSignal_UL	Refer to the OP2 pin of MOTOR2 DFB**.	—	—	—
Optional services						
Interlocks*		\$CONDSUM1_UL	Refer to the CONDSUM1 DFB**.	InterlockTags *	\$CONDSUM1_CD	Refer to <i>Discrete Motorized Valve Management</i> .
ForwardFailures*		\$CONDSUM_UL	Refer to the CONDSUM DFB**.	FailureTags *	\$CONDSUM_CD	
ReverseFailures*		\$CONDSUM_UL	Refer to the CONDSUM DFB**.	FailureTags *	\$CONDSUM_CD	
Maintenance		\$DEVMNT_UL	Refer to the DEVMNT DFB**.	Maintenance-Tags	\$DEVMNT_CD	
LocalPanel	Logic	\$MVALVEDLP_UL	Refer to the MVALVEDLP DFB**.	LocalPanelTags	\$MVALVEDLP_CD	Refer to <i>Discrete Motorized Valve Management</i> .
	LPMode-DISignal	\$DISignal_UL	Refer to the LPSignal pin of MVALVEDLP DFB**.	—	—	
	ZERODI-Signal	\$DISignal_UL	Refer to the ZEROSignal pin of MVALVEDLP DFB**.	—	—	
	CSDISig-nal	\$DISignal_UL	Refer to the CSSignal pin of MVALVEDLP DFB**.	—	—	
	OFFDI-Signal	\$DISignal_UL	Refer to the OFFSignal pin of MVALVEDLP DFB**.	—	—	
	OpenDI-Signal	\$DISignal_UL	Refer to the OpenSignal pin of MVALVEDLP DFB**.	—	—	

Control services (\$MotValved_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$MValved_CS)*	Corresponding facet template	Supervision service description
	CloseDI-Signal	\$DISignal_UL	Refer to the CloseSignal pin of MVALVEDLP DFB**.	–	–	
	PBLight-DOSignal	\$DOSignal_UL	Refer to the PBLightSignal pin of MVALVEDLP DFB**.	–	–	
* The service is activated by default.						
** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual						
*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
Dev1S2D	\$DEV1S2D/PO	Links to a device object
ILCK	\$Bool/Ref	EDT interface
RSPSEL	\$Bool/Ref	EDT interface
RSP	\$Bool/Ref	EDT interface
ILCKSEL	\$Bool/Ref	EDT interface
HILimitSwitch	\$Bool/Ref	EDT interface
LOLimitSwitch	\$Bool/Ref	EDT interface
MValvedName	\$MVALVED/Def	Instance name concatenated with control block name

\$Valve - On/Off Valve

General Description

The \$Valve process control module template is used to manage on/off type devices, such as on/off actuated valves.

The process template allows managing the associated devices from the sequential control, the continuous control, and/or the Supervision system. It depends on their configuration and the system needs.

Parameters

Configuration

The table describes the **Configuration** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
LowLimitSwitch		Negate	Boolean	False	True = Negates the input or output signal.
HighLimitSwitch					
EXTCTLContd					

Element Name	Child Element Name	Name	Type	Default value	Description
OPControl					
LocalPanel	LPModeDI-Signal				
	ZERODI-Signal				
	CSDISignal				
	OFFDISignal				
	ONDISignal				
	PBLight-OPSignal				
LowLimitSwitch		xSignalName	String	Null	Name used for the generated EFB/DFB and variable. x represents the element name.
HighLimitSwitch					
EXTCTLContd					
OPControl					
LocalPanel	LPModeDI-Signal				
	ZERODI-Signal				
	CSDISignal				
	OFFDISignal				
	ONDISignal				
	PBLight-OPSignal				

Time

The table describes the **Time** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
Valve	Logic	Timeout	Duration	00:00:03	Confirmation of timeout.
		ScanTime		00:00:05	Minimum duration taken by an active alarm.

Composition

The `$Valve` template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$Valve` control module and the corresponding facet, which implements the service:

Control services (\$VALVE_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$VALVE_CS)	Corre- sponding facet template	Supervision service description
Core services						
Valve	Logic	\$DEVCTL_UL	Refer to the DEVCTL DFB**.	ValveTags	\$DEVCTL_CD	Refer to On/ Off Valve Management***.
	—	—	—	Valve genies	\$hdright- tdown- valve3_ 10_CG	
					\$hdvalv- e2_10_CG	
					\$vdu- pright- valve3_ 10_CG	
					\$vdvalv- e2_10_CG	
					\$vtdval- ve2_10_ CG	
					\$vtvalv- e2_10_CG	
Optional I/O signals						
HighLimitSwitch *		\$DISignal_UL	Refer to the ZSH pin of DEVCTL DFB**.	—	—	—
LowLimitSwitch *		\$DISignal_UL	Refer to the ZSL pin of DEVCTL DFB**.	—	—	—
EXTCTLCond		\$DISignal_UL	Refer to the EXTCTLD pin of DEVCTL DFB**.	—	—	—
OPControl		\$DOSignal_UL	Refer to the OP pin of DEVCTL DFB**.	—	—	—
Optional services						
Interlocks *		\$CONDSUM1_UL	Refer to the CONDSUM1 DFB**.	InterlockTags	\$CON- DSUM1_CD	Refer to On/ Off Valve Management.
Maintenance		\$DEVMNT_UL	Refer to the DEVMNT DFB**.	Maintenance- Tags	\$DEVMNT_ CD	
LocalPanel	Logic	\$DEVLP_UL	Refer to the DEVLP DFB**.	LocalPanel- Tags	\$DEVLP_ CD	
	LPMo- deDI- Signal	\$DISignal_UL	Refer to the LPSignal pin of DEVLP DFB**.	—	—	
	ZERO- DISig- nal	\$DISignal_UL	Refer to the ZEROSignal pin of DEVLP DFB**.	—	—	
	CSDI- Signal	\$DISignal_UL	Refer to the CSSignal pin of DEVLPDFB**.	—	—	
LocalPanel	OFF- DISig- nal	\$DISignal_UL	Refer to the OFFSignal pin of DEVLP DFB**.	—	—	
	ONDI- Signal	\$DISignal_UL	Refer to the ONSignal pin of DEVLP DFB**.	—	—	
	PBLig- htOP- Signal	\$DOSignal_UL	Refer to the PBLightSignal pin of DEVLP DFB**.	—	—	
* The service is activated by default.						
** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual						
*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
ValveOP	\$Bool/Def	EDT interface
ILCK	\$Bool/Ref	EDT interface
RemoteSetpoint	\$Bool/Ref	EDT interface
LowLimitSwitch	\$Bool/Ref	EDT interface
HighLimitSwitch	\$Bool/Ref	EDT interface
OutputDOSignal	\$Bool/Def	EDT interface
ZSHPOS	\$Bool/Def	High limit position reached
ZSLPOS	\$Bool/Def	Low limit position reached
DEVCTLName	\$DEVCTL_IF/Def	Instance name concatenated with control block name

NOTE: The **ValveOP**, **ZSHPOS** and **ZSLPOS** interfaces can be used to achieve dual output functionality using \$DUALOP_UC template.

\$Valve2 - On/Off Valve With Two Outputs

General Description

The \$Valve2 process control module template is used to manage on/off type devices, such as on/off actuated valves with two outputs.

The process template allows managing the associated devices from the sequential control, the continuous control, and/or the Supervision system. It depends on their configuration and the system needs.

Parameters

Configuration

The table describes the **Configuration** parameters that you can configure:

Element Name	Child Element Name		Name	Type	Default value	Description
-	-		Mode	Enum	Output is energized until the position of limit switch is reached + the pulse time	Mode of operation.
DualOP	OPH	DOSignal	Negate	Boolean	False	True = Negates the input or output signal.
	OPL					
HighLimitSwitch	-					
LowLimitSwitch	-					
External-ControlSignal	-					
LocalPanel	LPModeDISignal					

Element Name	Child Element Name		Name	Type	Default value	Description
	ZERODISignal					
	CSDISignal					
	OFFDISignal					
	ONDISignal					
	PBLightOPSignal					
DualOP	OPH	DOSignal	Type	Boolean	Null	1 = Includes the quality signal.
	OPL					
DualOP	OPH	DOSignal	xSignalName	String	Null	Name used for the generated EFB/DFB and variables. x represents the child element name.
	OPL					
HighLimitSwitch	-					
LowLimitSwitch	-					
External-ControlSignal	-					
LocalPanel	LPModeDISignal					
	ZERODISignal					
	CSDISignal					
	OFFDISignal					
	ONDISignal					
	PBLightDISignal					
NOTE: Values of boolean parameters are set by using check boxes: <ul style="list-style-type: none">Selected = TrueCleared = False						

Time

The table describes the **Time** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
Valve2	Logic	Timeout	Duration	00:00:03	1 = Requires manual resetting of the control module when an abnormal condition is detected and cleared.
		ScanTime		00:00:05	Minimum duration taken by an active alarm.
DualOP	Logic	SwitchDelay	Duration	00:00:00	Time delay between switching of outputs.
		OPHPulseTime	Duration	00:00:03	OPH pulse time duration.
		OPLPulseTime	Duration	00:00:03	OPL pulse time duration.

NOTE: If you are not using a particular limit switch but still wish to drive the output in that direction, verify that the corresponding pulse time parameter value is greater than 0.

Composition

The \$Valve2 template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the \$Valve2 control module and the corresponding facet, which implements the service:

Control services (\$VALVED2DI2DOILCK_ UC)			Correspond- ing facet template	Control service description	Associated Supervision services (\$VALVE2_CS) *	Corre- sponding facet template	Supervi- sion service description
Core Services							
Valve2	Logic	\$VALVE2_UL	Refer to the DEVCTL DFB**.	Valve2Tags	\$DEVCTL_ CD	Refer to On/Off Valve (2 outputs) Manage- ment**.	
	—	—	—	Valve2 genies	\$hdrigh- tdown- valve3_ 10		
					\$hdvalv- e2_10		
					\$vdu- pright- valve3_ 10		
					\$vdvalv- e2_10		
					\$vtvalv- e2_10		
					\$vtdval- ve2_10		
	—	—	—	Control genies	\$DOut- put_CG		
DualOP	Logic	\$DualOP_UL	Refer to the DUALOP DFB**.	—	—	—	
Optional I/O signals							
DualOP	O- PH	Logic	\$DOUTPUT_UL	Refer to the DOUTPUT DFB**.	OpenValveTags	\$DOUT- PUT_CD	—
		DOSig- nal	\$DOSignal_ UL	Refer to the DOSIGNAL DFB (see EcoStruxure™ Process Expert, Foundation Control Services User Guide).	—	—	—
	O- PL	Logic	\$DOUTPUT_UL	Refer to the DOUTPUT DFB**.	CloseValveTags	\$DOUT- PUT_CD	—
		DOSig- nal	\$DOSignal_ UL	Refer to the DOSIGNAL DFB (see EcoStruxure™ Process Expert, Foundation Control Services User Guide).	—	—	—
HighLimitSwitch *		\$DISignal_ UL	Refer to the ZSH pin of DEVCTL DFB**.	—	—	—	
LowLimitSwitch *		\$DISignal_ UL	Refer to the ZSL pin of DEVCTL DFB**.	—	—	—	
ExternalControl		\$DISignal_ UL	Refer to the EXTCTLD pin of DEVCTL DFB**.	—	—	—	
Optional services							
Interlocks *		\$CONDSUM1_ UL	Refer to the CONDSUM1 DFB**.	InterlockTags *	\$CON- DSUM1_CD	Refer to On/Off Valve (2 outputs) Manage- ment.	
Maintenance		\$DEVMNT_UL	Refer to the DEVMNT DFB**.	Maintenance- Tags	\$DEVMNT_ CD		
LocalPa- nel	Logic	\$DEVLP_UL	Refer to the DEVLP DFB**.	LocalPanelTags	\$DEVLP_ CD		
	LPMoDeDI- Signal	\$DISignal_ UL	Refer to the LPSignal pin of DEVLP DFB**.	—	—		

Control services (\$VALVED2DI2DOILCK_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$VALVE2_CS) *	Corresponding facet template	Supervision service description
	ZeroDISignal	\$DISignal_UL	Refer to the ZEROSignal pin of DEVL P DFB**.	—	—	
	CSDISignal	\$DISignal_UL	Refer to the CSSignal pin of DEVL P DFB**.	—	—	
	OFFDISignal	\$DISignal_UL	Refer to the OFFSignal pin of DEVL P DFB**.	—	—	
	ONDISignal	\$DISignal_UL	Refer to the ONSignal pin of DEVL P DFB**.	—	—	
	PBLightoutOPSignal	\$DOSignal_UL	Refer to the PBLightSignal pin of DEVL P DFB**.	—	—	
* The service is activated by default.						
** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual						
*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

NOTE:

- When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.
- The **Mode** parameter of \$Valve2 control module template supports only two types of modes, Mode 0 and Mode 4, that are available in the \$DualOP_UC template. For further information about the **Mode** parameter available in \$DualOP_UC template, refer to the Mode input of the DUALOP DFB (see EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual).

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
OP	\$Bool/Def	EDT interface
ILCK	\$Bool/Ref	EDT interface
Fail	\$Bool/Ref	EDT interface
RSP	\$Bool/Ref	EDT interface
OPToVar	\$Bool/Ref	EDT interface
CHOUTwQuality	\$DOUTPUTSignal/PO	Links to a device object
TrackToVar	\$Bool/Ref	EDT interface
CHOUT	\$Bool/Def	EDT interface
SP	\$Bool/Def	EDT interface
Track	\$Bool/Def	EDT interface
HiLimitSwitch	\$Bool/Ref	EDT interface
LoLimitSwitch	\$Bool/Ref	EDT interface
DEVCTLName	\$DEVCTL_IF/Def	Instance name concatenated with control block name

Analog Device Control

Overview

This chapter describes the functionality of the analog device control templates and their composition.

These templates do not reflect any specific installation.

⚠ WARNING

LOSS OF CONTROL

- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.¹
- Test each implementation of a system for proper operation before placing it into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

¹ For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

\$ControlValve - Control Valve

General Description

The \$ControlValve process control module template manages control valves with optional position feedback (position and/or limit switches).

Parameters

Configuration

The table describes the **Configuration** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
ControlValve	Logic	BadPV	Float	0.0	Default value in engineering units to be used for the output signal when the channel is not operational.

Element Name	Child Element Name	Name	Type	Default value	Description
	PV	Type	Boolean	0	1 = Includes quality signal if applicable, for Int. Quality signal is always enabled for Real.
		AISignal-Name	String	Null	Name used for the generated EFB and variable.
	OPAOSignal	Type	Boolean	0	Includes signal quality and conditioning.
		AOSignal-Name	String	Null	Name used for the generated DFB and variables.
ControlValve	Open	Negate	Boolean	False	True = Negates the input or output signal.
	Close				
LocalPanel	LPModeDISignal				
	ZERODISignal				
	CSDISignal				
	OpenDISignal				
	CloseDISignal				
	PBLightOPSignal				
ControlValve	Open	xSignal-Name	String	Null	Name used for the generated EFB/DFB and variable. x represents the child element name.
	Close				
LocalPanel	LPModeDISignal				
	ZERODISignal				
	CSDISignal				
	OpenDISignal				
	CloseDISignal				
	PBLightOPSignal				
LocalPanel	Logic	DeltaSP	String	1.0	The absolute set-point increase or decrease value when the Open or Close pushbutton is pressed.
		Delta-FastSP	String	2.0	The absolute set-point increase or decrease value when the Open or Close pushbutton is held down.
		TRepeat	Duration	00:00:00:25	Determines the time for DeltaFastSP increase or decrease to be applied, when the Open or Close pushbutton is held down.
		Mode-SignalsEN	Boolean	True	True = Enables the use of LocalPanel and ControlSystem mode switch on the local panel.
		PBSignal-sEN			True = Enables the use of Open and Close push buttons on the local panel.
		ToProgramEN			True = Enables to change the owner of the valve to Program .
		LockOwnerEN			True = Enables to block the owner of the valve to remain as Operator while the local panel mode is LocalPanel .
		LockSet-pointEN		False	True = Enables to block the set point of the valve.
NOTE: Values of boolean parameters are set by using check boxes: <ul style="list-style-type: none"> Selected = True Cleared = False 					

NOTE: `$AISignal_UL` template has a parameter **ChannelDataType** which is not available for user configuration in `$ControlValve` control module PV child element. Its value is internally set to 0.

Range

The table describes the **Range** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
ControlValve	Logic	HighRaw-ChannelOut	Short	2047	High range for the CHOUT output signal.
		LowRaw-ChannelOut	Short	0	Low range for the CHOUT output signal.
		HighRaw-ChannelIn	Short	2047	High range for the CHIN input signal.
		LowRaw-ChannelIn	Short	0	Low range for the CHIN input signal.
		HighThreshold	Float	95.0	The threshold of open valve.
		LowThreshold	Float	5.0	The threshold of closed valve.

Time

The table describes the **Time** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
ControlValve	Logic	MaxDev	Float	5.0	The maximum permissible deviation between current set point and actual set point.
		TMaxDev	Duration	00:00:05	The maximum permissible time for the deviation between current set point and actual position.
		ScanTime	Duration	00:00:05	Minimum duration taken by an active alarm.

Composition

The `$ControlValve` template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$ControlValve` control module and the corresponding facet, which implements the service:

Control services (<code>\$ControlValve_UC</code>)		Corresponding facet template	Control service description	Associated Supervision services (<code>\$ControlValve_CS</code>)*	Corresponding facet template	Supervision service description
Core services						
ControlValve	Logic	<code>\$CVALVE_UL</code>	Refer to the <code>CVALVE</code> DFB**.	ControlValve-Tags	<code>\$CVALVE_CD</code>	Refer to Control Valve With Position Management***.
	—	—	—	Control valve genies	<code>\$CValve_HCRight-Down_CG</code>	
					<code>\$CValve_HC_CG</code>	

Control services (\$ControlValve_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$Control- Valve_CS)*	Corre- sponding facet template	Supervision service description
					\$CValve_ VCUp- Right_CG	
					\$CValve_ VC_CG	
Optional I/O signals						
Open		\$DISignal_UL	Refer to the ZSH pin of CVALVE DFB**.	—	—	—
Close		\$DISignal_UL	Refer to the ZSL pin of CVALVE DFB**.	—	—	—
PV		\$AISignal_UL	Refer to the CHIN pin of CVALVE DFB**.	—	—	—
OPAOSignal		\$AOSignal_UL	Refer to the OP pin of CVALVE DFB**.	—	—	—
Optional services						
Interlocks*		\$CONDSUM1_UL	Refer to the CONDSUM1 DFB**.	InterlockTags	\$CON- DSUM1_CD	Refer to <i>Control Valve With Position Management.</i>
LocalPanel	Logic	\$CVALVELP_UL	Refer to the CVALVELP DFB**.	LocalPanel- Tags	\$CVAL- VELP_CD	
	LPMo- deDISig- nal	\$DISignal_UL	Refer to the LPSignal pin of CVALVELP DFB**.	—	—	
	ZERO- DISignal	\$DISignal_UL	Refer to the ZEROSignal pin of CVALVELP DFB**.	—	—	
	CSDI- Signal	\$DISignal_UL	Refer to the CSSignal pin of CVALVELP DFB**.	—	—	
	OpenDI- Signal	\$DISignal_UL	Refer to the OpenSignal pin of CVALVELP DFB**.	—	—	
	CloseDI- Signal	\$DISignal_UL	Refer to the CloseSignal pin of CVALVELP DFB**.	—	—	
	PBLight- OPSig- nal	\$DOSignal_UL	Refer to the PBLightSignal pin of CVALVELP DFB**.	—	—	
* The service is activated by default.						
** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual						
*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
ILCK	\$Bool/Ref	EDT interface
RSP	\$Real/Ref	EDT interface
SPRange	\$Range/Ref	EDT interface
RSPRanged	\$RangedValue/Ref	EDT interface
RSPRangedwTrack	\$RSPRangedwithTracking/Ref	EDT interface
CVALVName	\$CVALVE_IF/Def	Instance name concatenated with control block name

\$MotorVS - Motor with Variable Speed Drive

General Description

The \$MotorVS process control module template is used to manage motors with a variable speed drive, independently if the speed drive is switched through communication, I/O wiring, or a mix of both.

The process template is designed to allow managing the associated devices from the sequential control, the continuous control, and/or the Supervision system depending on their configuration and on the system needs.

Optional Rearm Confirmation

The template allows you to display a dialog box upon rearming interlock conditions and the detection of abnormal conditions during operation.

To display the dialog box in runtime, select the **Enable Interlock Rearm Message Confirmation / Enable Failure Rearm Message Confirmation** parameter of the **Genies** element (Supervision services) in the **Properties** window of the instance.

Parameters

Configuration

⚠ WARNING					
UNINTENDED EQUIPMENT OPERATION					
<ul style="list-style-type: none"> The <i>CONFREARMEN</i> pin has to be set to 1 for the <i>OP</i> pin to become 0, under non-confirmed operations. The <i>FAILREARMEN</i> pin has to be set to 1 for the <i>OP</i> pin to become 0, under detected fail conditions. Changes to the configuration of these parameters have to be performed by competent personnel only. 					
Failure to follow these instructions can result in death, serious injury, or equipment damage.					

The table describes the **Configuration** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
Valve	Logic	ConfRear-men	Boolean	1	<ul style="list-style-type: none"> 1 = Withdraws the output <i>OP</i> when a non-confirmed operation is detected (<i>ALARM</i> = 1) and requires a mandatory manual resetting, once the <i>ALARM</i> condition disappears. 0 = The output <i>OP</i> remains in its previous state when a non-confirmed operation is detected (<i>ALARM</i> = 1) and does not need manual resetting, once the <i>ALARM</i> condition disappears.
Valve	Logic	FailRear-men	Boolean	1	<ul style="list-style-type: none"> 1 = Withdraws the output <i>OP</i> upon a detected failure condition (<i>FAILD</i> = 1) and requires a mandatory manual resetting, once the <i>FAIL</i> condition disappears. 0 = The output <i>OP</i> remains in its previous state upon a detected failure condition (<i>FAILD</i> = 1) and does not need manual resetting, once the <i>FAIL</i> condition disappears.
Running		Negate	Boolean	False	True = Negates the input or output signal.

Element Name	Child Element Name	Name	Type	Default value	Description
PowerRemoval					
ExternalControl					
Local-Panel	LocalModeDISignal				
	ZERODISignal				
	CSDISignal				
	OFFDISignal				
	DirectDISignal				
	ReverseDISignal				
	IncSPDISignal				
	DecSPDISignal				
	S1DISignal				
	S2DISignal				
	S3DISignal				
PBLightDOSignal					
Local-Panel	LocalModeDISignal	xSignal-Name	String	Null	Name used for the generated EFB/DFB and variable. x represents the child element name.
	ZERODISignal				
	CSDISignal				
	OFFDISignal				
	DirectDISignal				
	ReverseDISignal				
	IncSPDISignal				
	DecSPDISignal				
	S1DISignal				
	S2DISignal				
	S3DISignal				
	PBLightDOSignal				
Local-Panel	Logic	DeltaSP	Float	2.0	It is the setpoint when the push button is pressed.
		Delta-FastSP		1.0	It is the setpoint when the push button is held down.
		TRepeat	Duration	00:00:00-:25	Maximum time taken by the system to interpret that the command is repeated
NOTE: Values of boolean parameters are set by using check boxes: <ul style="list-style-type: none">Selected = TrueCleared = False					

Time

The table describes the **Time** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
Valve	Logic	Timeout	Duration	00:00:03	Confirmation for the timeout.
		ScanTime		00:00:05	Minimum duration taken by an active alarm.

Composition

The `$MotorVS` template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$MotorVS` control module and the corresponding facet, which implements the service:

Control services (\$MotorVS_UC)	Corresponding facet template		Control service description	Associated Supervision services (\$MotorVS_CS)*	Corresponding facet template	Supervision service description
Core services						
MotorVS	Logic	\$SDDEVCTL_UL	Refer to the SDDEVCTL DFB**.	MotorV-STags	\$SDD-EVCTL-L_CD	Refer to Motor With Variable Speed Management***.
	—	—	—	Motor genies	\$SDD-EVCTL-LHLM-OTOR_CG	
					\$SDD-EVCTL-LHRM-OTOR_CG	
					\$SDD-EVCTL-LVMO-TOR_CG	
					\$SDD-EVCTL-LVU-MO-TOR_CG	
	—	—	—	Pump genies	\$SDD-EVCTL-LRCP-UMP_CG	
					\$SDD-EVCTL-LPVS-P_CG	
					\$SDD-EVCTL-LPVS-POP_CG	
I/O signals						
Running	\$DISignal_UL	Refer to the RUN pin of SDDEVCTL DFB**.	—	—	—	
Fail	\$DISignal_UL	Refer to the FAIL pin of SDDEVCTL DFB**.	—	—	—	
PowerRemoval	\$DISignal_UL	Refer to the POWERRMVL pin of SDDEVCTL DFB**.	—	—	—	

Control services (\$MotorVS_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$MotorVS_CS) *	Corresponding facet template	Supervision service description
ExternalControl		\$DISignal_UL	Refer to the EXTCTLD pin of SDDEVCTL DFB**.	—	—	—
OPDOSignal		\$DOSignal_UL	Output signal of the motor.	—	—	—
DirectionDOSignal		\$DOSignal_UL	Direction signal of the motor.	—	—	—
Optional services						
Interlocks*		\$CONDSUM1_UL	Refer to the CONDSUM1 DFB**.	InterlockTags *	\$CONDSUM1_CD	Refer to Motor With Variable Speed Management.
Failures*		\$CONDSUM_UL	Refer to the CONDSUM DFB**.	FailureTags *	\$CONDSUM_CD	
Maintenance		\$DEVMNT_UL	Refer to the DEVMNT DFB**.	MaintenanceTags	\$DEVMNT_CD	
LocalPanel	Logic	\$SDDEVLP_UL	Refer to the SDDEVLP DFB**.	LocalPanelTags	\$SDDEVLP_CD	Refer to Motor With Variable Speed Management.
	LocalModeDISignal	\$DISignal_UL	Refer to the LPSignal pin of SDDEVLP DFB**.	—	—	
	ZERODISignal	\$DISignal_UL	Refer to the ZEROSignal pin of SDDEVLP DFB**.	—	—	
	CSDISignal	\$DISignal_UL	Refer to the CSSignal pin of SDDEVLP DFB**.	—	—	
	OFFDISignal	\$DISignal_UL	Refer to the OFFSignal pin of SDDEVLP DFB**.	—	—	
	DirectDISignal	\$DISignal_UL	Refer to the DirectSignal pin of SDDEVLP DFB**.	—	—	
	ReverseDISignal	\$DISignal_UL	Refer to the ReverseSignal pin of SDDEVLP DFB**.	—	—	
LocalPanel	IncSPDISignal	\$DISignal_UL	Refer to the IncSPSignal pin of SDDEVLP DFB**.	—	—	
	DecSPDISignal	\$DISignal_UL	Refer to the DecSPSignal pin of SDDEVLP DFB**.	—	—	
	S1DISignal	\$DISignal_UL	Refer to the S1Signal pin of SDDEVLP DFB**.	—	—	
	S2DISignal	\$DISignal_UL	Refer to the S2Signal pin of SDDEVLP DFB**.	—	—	

Control services (\$MotorVS_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$MotorVS_CS)*	Corresponding facet template	Supervision service description
	S3DLSignal	\$DISignal_UL	Refer to the S3Signal pin of SDDEVLP DFB**.	—	—	
	PBLightDOSignal	\$DOSignal_UL	Refer to the PBLightSignal pin of SDDEVLP DFB**.	—	—	

* The service is activated by default.

** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual

*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
ILCK	\$Bool/Ref	EDT interface
OP	\$Real/Def	EDT interface
DevVs	\$DEVVS/PO	Links to a device object
PV	\$Real/Ref	EDT interface
PVRange	\$Range/Ref	EDT interface
PVRanged	\$RangedValue/Ref	EDT interface
RSP	\$Real/Ref	EDT interface
ILCKSP	\$Real/Ref	EDT interface
OPSignal	\$Bool/Def	EDT interface
Direction	\$Bool/Def	EDT interface
SDDEVCTLName	\$SDDEVCTL_IF/Def	Instance name concatenated with control block name

\$MValveWithPos - Motorized Valve with Positioner

General Description

The \$MValveWithPos process control module template is designed to control a motorized valve or a gate with position feedback and two-rotational-directions-motor based control.

Optional Rearm Confirmation

The template allows you to display a dialog box upon rearming interlock conditions and the detection of abnormal conditions during operation.

To display the dialog box in runtime, select the **Enable Interlock Rearm Message Confirmation / Enable Failure Rearm Message Confirmation** parameter of the **Genies** element (Supervision services) in the **Properties** window of the instance.

Parameters

Configuration

⚠ WARNING
UNINTENDED EQUIPMENT OPERATION <ul style="list-style-type: none"> The <i>CONFREARMEN</i> pin has to be set to 1 for the <i>SP_MOTOR2</i> pin to become 0, under non-confirmed operations. Changes to the configuration of these parameters have to be performed by competent personnel only. <p>Failure to follow these instructions can result in death, serious injury, or equipment damage.</p>

The table describes the **Configuration** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
MValve	Logic	ConfRearmEN	Boolean	1	<ul style="list-style-type: none">1 = Withdraws the output <i>SP_MOTOR2</i> when a non-confirmed operation is detected (<i>ALARM</i> = 1) and requires a mandatory manual resetting, once the <i>ALARM</i> condition disappears.0 = The output <i>SP_MOTOR2</i> remains in its previous state when a non-confirmed operation is detected (<i>ALARM</i> = 1) and does not need manual resetting, once the <i>ALARM</i> condition disappears.
Motor2	Logic				
MValve	Logic	FailRearmEN	Boolean	1	<ul style="list-style-type: none">1 = Withdraws the output <i>SP_MOTOR2</i> upon a detected failure condition (<i>FAILD</i> = 1) andrequires a mandatory manual resetting, once the <i>FAIL</i> condition disappears.0 = The output <i>SP_MOTOR2</i> remains in its previous state upon a detected failure condition (<i>FAILD</i> = 1) and does not need manual resetting, once the <i>FAIL</i> condition disappears.
Motor2	Logic				
Position	Logic	BadPV	Float	0.0	Default value in engineering units to be used for the output signal when the channel is not operational.
HighLimitSwitch	Logic				
LowLimitSwitch	Logic				
HighLimitSwitch	Logic	Watch	Enum	Null	Watch alarm
LowLimitSwitch	Logic				
HighLimitSwitch	Logic	AlarmST	Enum	Null	PV state to be considered as alarm.
LowLimitSwitch	Logic				
HighLimitSwitch	DISignal	Negate	Boolean	False	True = Negates the input or output signal.
LowLimitSwitch	DISignal				
ForwardRunningSignal					
ReverseRunningSignal					
ForwardFailSignal					
ReverseFailSignal					
EXTCTLContd					

Element Name	Child Element Name	Name	Type	Default value	Description
OP1DOSignal					
OP2DOSignal					
LocalPanel	LPMoDeDISig-nal				
	ZERODISignal				
	CSDISignal				
	OFFDISignal				
	ONDISignal				
	OpenDISignal				
	CloseDISignal				
	PBLightOPSig-nal				
HighLimitSwitch	DISignal	Type	Boolean	Null	1 = Includes the quality signal.
LowLimitSwitch	DISignal				
HighLimitSwitch	DISignal	xSignalName	String	Null	Name used for the generated EFB/DFB and variable. x represents the element name.
LowLimitSwitch	DISignal				
ForwardRunningSignal					
ReverseRunningSignal					
ForwardFailSignal					
ReverseFailSignal					
EXTCTLContd					
OP1DOSignal					
OP2DOSignal					
LocalPanel	LPMoDeDISig-nal				
	ZERODISignal				
	CSDISignal				
	OFFDISignal				
	ONDISignal				
	OpenDISignal				
	CloseDISignal				
	PBLightOPSig-nal				
LocalPanel	Logic	DeltaSP	String	1.0	The absolute set-point increase or decrease value when the Open or Close pushbutton is pressed.
		DeltaFastSP	String	2.0	The absolute set-point increase or decrease value when the Open or Close pushbutton is held down.

Time

The table describes the **Time** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
MValve	Logic	TMax	Duration	00:00:03	Maximum time allowed to reach the Open or Close position.

Element Name	Child Element Name	Name	Type	Default value	Description
		TWatch	Duration	00:00:05	Time taken to check whether the current position has changed 1% to the right direction.
MValve	Logic	Scantime	Duration	00:00:05	Minimum duration taken by an active alarm.
Motor2	Logic				
HighLimitSwitch	Logic	TONSP	Duration	00:00:00	ON timer set point for digital input connection.
LowLimitSwitch	Logic				
HighLimitSwitch	Logic	TOFFSP	Duration	00:00:00	OFF timer set point for digital input disconnection.
LowLimitSwitch	Logic				
Motor2	Logic	Timeout	Duration	00:00:03	Confirmation for the timeout.
LocalPanel	Logic	TRepeat	Duration	00:00:00:2-5	Determines the time for DeltaFastSP increase or decrease to be applied, when the Open or Close pushbutton is held down.

Range

The table describes the **Range** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
Position	Logic	HighRaw	Short	2047	High range for the CHIN input signal.
		LowRaw	Short	0	Low range for the CHIN input signal.
		PVRangeHigh	Float	100.0	High range in engineering units.
		PVRangeLow	Float	0.0	Low range in engineering units.
		PVEngineeringUnit	String	%	Engineering units.
		PVFormat	String	####.#EU	Format type

Composition

The `$MValveWithPos` template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$MValveWithPos` control module and the corresponding facet, which implements the service:

Control services (<code>\$MValveWithPos_UC</code>)		Corresponding facet template	Control service description	Associated Supervision services (<code>\$MValveWithPos_CS</code>) [*]	Corresponding facet template	Supervision service description
Core services						
MValve	Logic	<code>\$MVALVE_UL</code>	Refer to the <code>MVALVE</code> DFB ^{**} .	MValveTags	<code>\$Mot-VALVE_CD</code>	Refer to Motorized Valve With Positioner Management ^{***} .
	—	—	—	MValve genies	<code>\$MVALVE_10_CG</code>	

Control services (\$MValveWithPos_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$MValve- WithPos_ CS) *	Corre- sponding facet template	Supervision service description
					\$MVALVE_ S_CD	
					\$MVALVE_ TREND_CD	
I/O signals						
Position	Logic	\$AINPUT_UL	Refer to the AINPUT DFB**.	PositionTags	\$AINPUT_ CD	
	Ana- logInput- Signal *	\$AISignal_UL	Refer to the AISignalCondReal1 and AISignalCond1 EFB**.	—	—	—
HighLimitSwitch	Logic	\$DINPUT_UL	Refer to the DINPUT DFB**.	HighLimitS- witchTags	\$DINPUT_ CD	
	DISignal	DISignal_UL	Refer to the DISignalCond1 EFB**.	—	—	—
LowLimitSwitch	Logic	\$DINPUT_UL	Refer to the DINPUT DFB**.	LowLimitS- witchTags	\$DINPUT_ CD	—
	DISignal	DISignal_UL	Refer to the DISignalCond1 EFB**.	—	—	—
Motor2	Logic	\$MOTOR2_UL	Refer to the MOTOR2 DFB**.	Motor2Tags	\$MOTOR2_ CD	
Optional I/O signals						
ForwardRunningSignal		\$DISignal_UL	Refer to the ZSH1 pin of MOTOR2 DFB**.	—	—	—
ReverseRunningSignal		\$DISignal_UL	Refer to the ZSH2 pin of MOTOR2 DFB**.	—	—	—
ForwardFailSignal		\$DISignal_UL	Refer to the FAIL1 pin of MOTOR2 DFB**.	—	—	—
ReverseFailSignal		\$DISignal_UL	Refer to the FAIL2 pin of MOTOR2 DFB**.	—	—	—
ExternalControl		\$DISignal_UL	Refer to the EXTCTLD pin of MOTOR2 DFB**.	—	—	—
OP1DOSignal		\$DOSignal_UL	Refer to the OP1 pin of MOTOR2 DFB**.	—	—	—
OP2DOSignal		\$DOSignal_UL	Refer to the OP2 pin of MOTOR2 DFB**.	—	—	—
Optional services						
Interlocks*		\$CONDSUM1_UL	Refer to the CONDSUM1 DFB**.	InterlockTags	\$CON- DSUM1_CD	Refer to <i>Motorized Valve With Positioner Management.</i>
ForwardFailures*		\$CONDSUM_UL	Refer to the CONDSUM DFB**.	FailureTags *	\$CON- DSUM_CD	
ReverseFailures*		\$CONDSUM_UL	Refer to the CONDSUM DFB**.	FailureTags *	\$CON- DSUM_CD	
Maintenance		\$DEVMNT_UL	Refer to the DEVMNT DFB**.	Maintenan- ceTags	\$DEVMNT_ CD	
LocalPanel	Logic	\$MVALVELP_UL	Refer to the MVALVELP DFB**.	LocalPanel- Tags	\$MVAL- VELP_CD	Refer to <i>Motorized Valve With Positioner Management.</i>
	LPMode- DISignal	\$DISignal_UL	Refer to the LPSignal pin of MVALVELP DFB**.	—	—	
	ZERODI- Signal	\$DISignal_UL	Refer to the ZEROSignal pin of MVALVELP DFB**.	—	—	
	CSDI- Signal	\$DISignal_UL	Refer to the CSSignal pin of MVALVELP DFB**.	—	—	

Control services (\$MValveWithPos_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$MValve- WithPos_ CS) *	Corre- sponding facet template	Supervision service description
	OFFDI-Signal	\$DISignal_UL	Refer to the OFFSignal pin of MVALVELP DFB**.	—	—	
	ONDI-Signal	\$DISignal_UL	Refer to the ONSignal pin of MVALVELP DFB**.	—	—	
	OpenDI-Signal	\$DISignal_UL	Refer to the OpenSignal pin of MVALVELP DFB**.	—	—	
	CloseDI-Signal	\$DISignal_UL	Refer to the CloseSignal pin of MVALVELP DFB**.	—	—	
	PBLight-OPSignal	\$DOSignal_UL	Refer to the PBLightSignal pin of MVALVELP DFB**.	—	—	
* The service is activated by default.						
** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual						
*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
Dev1S2D	\$DEV1S2D/PO	Links to a device object
RSP	\$Real/Ref	EDT interface
Rstart	\$Bool/Ref	EDT interface
ILCKSP	\$Real/Ref	EDT interface
HiLimitSwitch	\$Bool/Ref	EDT interface
LoLimitSwitch	\$Bool/Ref	EDT interface
MValveName	\$MVALVName/Def	Instance name concatenated with control block name

Process Control

Overview

This chapter describes the functionality of the process control templates and their composition.

These templates do not reflect any specific installation.

⚠ WARNING

LOSS OF CONTROL

- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.¹
- Test each implementation of a system for proper operation before placing it into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

¹ For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

\$IMC - Internal Model Controller

General Description

The \$IMC process control module template is used to condition the signals associated with the control of a standard Control library internal model controller (IMC).

It helps monitor and control the controller easily from the Supervision system and provides the operating modes used in the remaining EcoStruxure Process Expert process templates.

The IMC internal model controller is designed to control processes with pure delays as well as non-linear processes.

Parameters

OPRange

The table describes the **OPRange** parameters that you can configure:

Element Name	Name	Type	Default value	Description
Logic	OPRangeHigh	Float	100.0	High output range (in engineering units)
	OPRangeLow		0.0	Low output range (in engineering units)
	OPRangeEU	String	%	Output (in engineering units)
	OPRangeFormat		####.#EU	Output format

Time

The table describes the **Time** parameters that you can configure:

Element Name	Name	Type	Default value	Description
Logic	Interval	Duration	00:00:01	Sampling interval
	DelayScan	Integer	0	The total number of delay scans after cold start.
	Period	Duration	00:00:01	Sampling period
	DivisorValue	Float	1000.0	The value of divisor

Composition

The $\$IMC$ template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the $\$IMC$ control module and the corresponding facet, which implements the service:

Control services (\$IMC_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$IMC_CS) *	Corre- sponding facet template	Supervision service description
Core services						
InternalModelControl- ler	Logic	\$IMCTL_UL	Refer to the IMCTL DFB**.	IMController- Tags	\$IMCTL_ CD	Refer to IMC Controller Manage- ment***.
	—	—	—	IMControl genies	IMCTLS- P_CG	
					IM- CTLPVS- POP_CG	
					IMCTLPV- SP_CG	
					IM- CTLSP- P_CG	
Optional services						
Interlocks *		\$CONDSUM1_UL	Refer to the CONDSUM1 DFB**.	InterlockTags	\$CON- DSUM1_CD	
<p>* The service is activated by default.</p> <p>** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual</p> <p>*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual</p>						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
ILCK	\$Bool/Ref	EDT interface
PV	\$Real/Ref	EDT interface
PVRange	\$Range/Ref	EDT interface
PVRanged	\$RangedValue/Ref	EDT interface
RSP	\$Real/Ref	EDT interface
FF	\$Real/Ref	EDT interface
OUTD	\$Real/Def	EDT interface
DMO	\$Real/Def	EDT interface
OUT	\$Real/Def	EDT interface
OUTRange	\$Range/Def	EDT interface
OUTRanged	\$RangedValue/Def	EDT interface
OUTRangedwTrack	\$RangedValuewithTracking/Def	EDT interface
IMCTLName	\$IMCTL_IF/Def	Instance name concatenated with control block name

\$LeadLag - Lead-Lag Controller

General Description

The \$LeadLag process control module template is used to condition the signals associated to the control of a lead-lag controller.

This process template helps you monitor and control the controller easily from the Supervision system and provides the operating modes used in the remaining EcoStruxureProcess Expert process templates.

Parameters

Range

The table describes the **Range** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
LeadLag	Logic	OPRangeHigh	Float	100.0	High output range (in engineering units)
		OPRangeLow		0.0	Low output range (in engineering units)
		OPRangeEU	String	%	Output (in engineering units)
		OPRangeFormat		####.#EU	Output format

Composition

The \$LeadLag template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$LeadLag` control module and the corresponding facet, which implements the service:

Control services (\$LeadLag_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$LeadLag_CS)*	Corre- sponding facet template	Supervision service description
Core services						
LeadLag	Logic	\$LDLGCTL_UL	Refer to the LDLGCTL DFB**.	LeadLag- Tags	\$LDLGCT- L_CD	Refer to Lead-Lag Controller Manage- ment***.
	—	—	—	LeadLag genies	\$LDLGCT- LSPOP_CG	
Optional services						
Interlocks *		\$CONDSUM1_UL	Refer to the CONDSUM1 DFB**.	InterlockTags	\$CON- DSUM1_CD	
* The service is activated by default.						
** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual						
*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
ILCK	\$Bool/Ref	EDT interface
RSP	\$Real/Ref	EDT interface
SPRange	\$Range/Ref	EDT interface
SPRanged	\$RangedValue/Ref	EDT interface
SP	\$Real/Def	EDT interface
OP	\$Real/Def	EDT interface
OPRanged	\$RangedValue/Def	EDT interface
LDLGCTLName	\$LDLGCTL_IF/Def	Instance name concatenated with control block name

\$PID - PID Controller

General Description

The `$PID` process control module template is used to condition the signals associated with the control of a PIDFF-type controller.

This process template helps monitor and control the controller easily from the Supervision system and provides the operating modes used in the remaining EcoStruxure Process Expert process templates.

Parameters

Range

The table describes the **Range** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
PIDController	Logic	OPRangeHigh	Float	100.0	High output range (in engineering units)
		OPRangeLow		0.0	Low output range (in engineering units)
		OPRangeEU	String	%	Output (in engineering units)
		OPRangeFormat		####.#EU	Output format

Configuration

The table describes the **Configuration** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
PIDMultiplexer	Logic	DescriptionGroup1	String	Tuning 1	Description for tuning parameters 1
		DescriptionGroup2		Tuning 2	Description for tuning parameters 2

Composition

The `$PID` template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$PID` control module and the corresponding facet, which implements the service:

Control services (<code>\$PID_UC</code>)	Corresponding facet template	Control service description	Associated Supervision services (<code>\$PID_CS</code>) *	Corresponding facet template	Supervision service description
Core services					
PIDController	Logic	<code>\$PIDCTL_UL</code>	Refer to the <code>PIDCTL DFB</code> **.	PIDControllerTags	Refer to PID Controller Management***.
	—	—	—	PIDControllerGenies	
				<code>\$PIDCTL_CD</code>	
				<code>\$PIDPVS-P_CG</code>	
				<code>\$PIDPV-SPOP_CG</code>	
				<code>\$PID-SPOP_CG</code>	
				<code>\$PIDSP_CG</code>	
				<code>\$PIDmultiplex-er_CG</code>	

Control services (\$PID_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$PID_CS) *	Corre- sponding facet template	Supervision service description
Optional services						
PIDMultiplexer	Logic	\$PIDMUX_UL	Refer to the PIDMUX DFB**.	PIDMUX- Tags	\$PIDMUX_ CD	
Interlocks *		\$CONDSUM1_UL	Refer to the CONDSUM1 DFB**.	InterlockTags	\$CON- DSUM1_CD	
<p>* The service is activated by default.</p> <p>** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual</p> <p>*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual</p>						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
ILCK	\$Bool/Ref	EDT interface
PV	\$Real/Ref	EDT interface
PVRange	\$Range/Ref	EDT interface
PVRanged	\$RangedValue/Ref	EDT interface
RSP	\$Real/Ref	EDT interface
OUT	\$Real/Def	EDT interface
OUTRange	\$Range/Ref	EDT interface
OUTRangedwTrack	\$RangedValuewithTracking/ Def	EDT interface
OUTRanged	\$RangedValue/Def	EDT interface
PIDCTLName	\$PIDCTL_IF/Def	Instance name concatenated with control block name

\$PIDLoop - PID Closed Loop Circuit

General Description

The \$PIDLoop process control module template models a closed loop circuit, which is integrated with the process templates to condition the signals associated with the control of a PIDFF-type controller.

This process template helps monitor and control the controller easily from the Supervision system and provides the operating modes used in the remaining EcoStruxure Process Expert process templates.

Parameters

PV Configuration

The table describes the **PV** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
-	-	PVName	String	Null	Name of present value.

Element Name	Child Element Name	Name	Type	Default value	Description
		PVDesc			Description of present value.

OP Configuration

The table describes the **OP** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
-	-	OPName	String	Null	Name of output.
		OPDesc			Description of output.

NOTE: The process control module template `$PIDLoop` is an aggregation of the parameters from the following templates:

- PV - `$AnalogInput1`, page 28
- PID - `$PID`, page 85
- OP - `$AnalogOutput`, page 31

Composition

The `$PIDLoop` template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

NOTE: The process control module template `$PIDLoop` is an aggregation of the following templates:

- PV (interface referenced by `$AnalogInput1`, page 28)
- PID (interface referenced by `$PID`, page 85)
- OP (interface referenced by `$AnalogOutput`, page 31)
- For the `$PIDLoop` template, you can enter a maximum of 15 characters as `InstanceID`.

\$PWMController - Pulse Width Modulated Controller

General Description

The `$PWMController` process control module template is used to condition the signals associated with pulse-width modulation (PWM) control.

Parameters

Configuration

The table describes the **Configuration** parameters that you can configure:

Element Name	Name	Type	Default value	Description
OPIncDOSignal	Negate	Boolean	False	True = Negates the input or output signal.
OPDecDOSignal				

Element Name	Name	Type	Default value	Description
OPIncDOSignal	xSignalName	String	Null	Name used for the generated DFB and variables.
OPDecDOSignal				x represents the child element name.
<p>NOTE: Values of boolean parameters are set by using check boxes:</p> <ul style="list-style-type: none">Selected = TrueCleared = False				

Composition

The \$PWMController template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the \$PWMController control module and the corresponding facet, which implements the service:

Control services (\$PWMController_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$PWMCon- troller_CS) *	Corre- sponding facet template	Supervision service description
Core services						
PWMCTL	Logic	\$PWMCTL_UL	Refer to the PWMCTL DFB**.	PWMCTLTags	\$PWMCTL_CD	Refer to Pulse-Width Modulation Controller Management***.
	—	—	—	PWMCTLgenies	\$PWMCTL_CG	
Optional I/O signals						
OPIncDOSignal *		\$DOSignal_UL	Refer to the OPINC pin of PWMCTL DFB**.	—	—	—
OPDecDOSignal *		\$DOSignal_UL	Refer to the OPDEC pin of PWMCTL DFB**.	—	—	—
Optional services						
Interlocks *		\$CONDSUM1_UL	Refer to the CONDSUM1 DFB**.	InterlockTags	\$CON- DSUM1_CD	
* The service is activated by default.						
** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual						
*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
OPINC	\$Bool/Def	EDT interface
OPDEC	\$Bool/Def	EDT interface
ILCK	\$Bool/Ref	EDT interface
RSP	\$Real/Ref	EDT interface
SPRange	\$Range/Ref	EDT interface

Interface identifier	Type/role	Description
SPRanged	\$RangedValue/Ref	EDT interface
OPIncDOSignal	\$Bool/Def	EDT interface
OPDecDOSignal	\$Bool/Def	EDT interface
PMWCTLName	\$PWMCTL_IF/Def	Instance name concatenated with control block name

\$Ramp - Ramp

General Description

The \$Ramp process control module template is used to generate rising or falling setpoint ramps for other control blocks, such as PID regulators, direct analog outputs.

The template provides various functions, for example:

- Tracking (for example, for setpoints or measurements)
- Temporary ramp shutdown due to maximum deviation
- Management of setpoints for monitoring and/or control

Parameter Table

The \$Ramp template features no configurable parameters other than the general \$System parameters (\$Name, \$Description, and \$Area).

Composition

The \$Ramp template references composite and facet templates, which provide the following services:

Control	Core services.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the \$Ramp control module and the corresponding facet, which implements the service:

Control services \$Ramp_UC	Corresponding facet template	Control service description	Associated Supervision services (\$Ramp_CS) *	Corresponding facet template	Supervision service description
Logic	\$ARAMP_UL	Refer to the ARAMP DFB*.	ARa-mpTags	\$ARAMP_CD	Refer to Ramp Management**.
—	—	—	Ramp genres	\$ARA-MPISP_CG	
				\$ARAMP-SP_CG	
* See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual					
** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual					

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
PV	\$Real/Ref	EDT interface
PVRange	\$Range/Ref	EDT interface
PVRanged	\$RangedValue/Ref	EDT interface
TrackSP	\$Real/Ref	EDT interface
TargetSP	\$Real/Ref	EDT interface
SP	\$Real/Def	EDT interface
SPRanged	\$RangedValue/Def	EDT interface
ARAMPName	\$ARAMP_IF/Def	Instance name concatenated with control block name

\$RatioCtl - Ratio Controller

General Description

The \$RatioCtl process control module template is used to condition the signals associated with a standard Control library RATIO controller.

It helps monitor and control the controller from the Supervision system and provides the operating modes used in the remaining EcoStruxure Process Expert process templates.

The RATIO controller sets a setpoint that is based on a measurement signal and on the ratio that needs to be maintained between the measurement signal and the setpoint. This process template is especially used for dosing materials that need to have a certain ratio between each other.

Parameters

OPRange

The table describes the **OPRange** parameters that you can configure:

Element Name	Name	Type	Default value	Description
Logic	OPRangeHigh	Float	100.0	High output range (in engineering units)

Element Name	Name	Type	Default value	Description
	OPRangeLow	String	0.0	Low output range (in engineering units)
	OPRangeEU		%	Output (in engineering units)
	OPRangeFormat		####.#EU	Output format

KRange

The table describes the **KRange** parameters that you can configure:

Element Name	Name	Type	Default value	Description
Logic	KRangeHigh	Float	100.0	High K range (in engineering units)
	KRangeLow		0.0	Low K range (in engineering units)
	KRangeEUnit	String	%	K (in engineering units)
	KRangeFormat		####.#EU	K format

Composition

The `$RatioCtl` template references composite and facet templates, which provide the following services:

Control	Core services.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$RatioCtl` control module and the corresponding facet, which implements the service:

Control services <code>\$RATIOCTL_UC</code>	Corresponding facet template	Control service description	Associated Supervision services (<code>\$RA-TIOCTL_CS</code>) *	Corresponding facet template	Supervision service description
Logic	<code>\$RATIOCTL_UL</code>	Refer to the <code>RATIOCTL DFB</code> (see <i>EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual</i>).	RatioControllerTags	<code>\$RA-TIOCTL_CD</code>	Refer to Ratio Controller Management (see <i>EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual</i>).
–	–	–	RatioControllergenies	<code>\$RA-TIOCTLPV-PVTOP_CG</code>	Process Supervision Services Reference Manual).

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
PV	<code>\$Real/Ref</code>	EDT interface
PVTrack	<code>\$Real/Ref</code>	EDT interface
RK	<code>\$Real/Ref</code>	EDT interface
OP	<code>\$Real/Def</code>	EDT interface

Interface identifier	Type/role	Description
OPRanged	\$RangedValue/Def	EDT interface
RatioCtlName	\$ARATIOCTL_IF/Def	Instance name concatenated with control block name

\$SplitRange - Split-Range Controller

General Description

The \$SplitRange process control module template is used to condition the signals associated to the standard Control library `SPLRG` DFB.

It helps monitor and control the controller from the Supervision project and provides the operating modes used in the remaining EcoStruxure Process Expert process templates.

The standard Control library `SPLRG` block is used for split-range controllers (for example, cold/heat) because it allows you to generate two output signals based on a single signal, which can come, for example, from the output of a PID controller.

You can link this template with the `PIDCTL_UC` and `PIDMUX_UC` process templates by using the Links Editor (see EcoStruxure™ Process Expert, User Guide) when you want to control a process with two control modules (for example, two modulating valves) and PID control.

Parameters

Range

The table describes the **Range** parameters of the **Logic** element of **SplitRange** that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
SplitRange	Logic	OPRangeHigh	Float	100.0	High output range (in engineering units)
		OPRangeLow		0.0	Low output range (in engineering units)
		OPRangeEU	String	%	Output (in engineering units)
		OPRangeFormat		####.#EU	Output format

Composition

The \$SplitRange template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the \$SplitRange control module and the corresponding facet, which implements the service:

Control services (\$SplitRange_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$SplitRange_CS) *	Corre- sponding facet template	Supervision service description
Core services						
SplitRange	Logic	\$SPLRGCTL_UL	Refer to the SPLRGCTL DFB**.	SplitRange- Tags	\$SPLRGCTL- L_CD	Refer to Split- Range Controller Management***.
	—	—	—	SplitRange genies	\$SPLRGCTL- L_PVOP_CG	
Optional services						
Interlocks *		\$CONDSUM1_UL	Refer to the CONDSUM1 DFB**.	InterlockTags *	\$CON- DSUM1_CD	
* The service is activated by default.						
** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual						
*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual						

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
ILCK	\$Bool/Ref	EDT interface
RSP	\$Real/Ref	EDT interface
SPRange	\$Range/Ref	EDT interface
RSPRanged	\$RangedValue/Ref	EDT interface
OP1	\$Real/Def	EDT interface
OP2	\$Real/Def	EDT interface
OPRange	\$Range/Def	EDT interface
OP1Ranged	\$RangedValue/Def	EDT interface
OP2Ranged	\$RangedValue/Def	EDT interface
SPLRGCTLName	\$SPLRGCTL/Def	Instance name concatenated with control block name

\$Step3Ctl - Three-Step Controller/Positioner

General Description

The \$Step3Ctl process control module template is used to condition the signals associated with the control of a three-step-type controller.

It helps you to monitor and control the controller easily from the Supervision system and provides the operating modes used in the remaining EcoStruxure Process Expert process templates.

The template controls process variables through discrete control modules (for example, temperature control through hot/cold on-off valves) or is used for the positioning of analog components with discrete drives (for example, a motorized valve or a gate with position and control feedback through a two-direction motor).

Common Links

Depending on the application, link the `$Step3Ctl` process template with templates such as `$AnalogInput`, page 26, `$DEVCTL UC`, `$Motor2`, page 53, `$DigitalOutput`, page 38 by using the Links Editor (see EcoStruxure™ Process Expert, User Guide).

Parameters

Configuration

The table describes the **Configuration** parameters of the **OPIncDOSignal**, and **OPDecDOSignal** elements that you can configure:

Element Name	Name	Type	Default value	Description
OPIncDOSignal	Negate	Boolean	False	True = Negates the input or output signal.
OPDecDOSignal				
OPIncDOSignal	DOSignalName	String	Null	Name used for the generated DFB and variables.
OPDecDOSignal				
<p>NOTE: Values of boolean parameters are set by using check boxes:</p> <ul style="list-style-type: none">Selected = TrueCleared = False				

Composition

The `$Step3Ctl` template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$Step3Controller` control module and the corresponding facet, which implements the service:

Control services (\$STEP3Controller_UC)		Corresponding facet template	Control service description	Associated Supervision services (\$STEP3Controller_CS)*	Corresponding facet template	Supervision service description
Core services						
STEP3-CTL	Logic	\$STEP3CTL_UL	Refer to the STEP3CTL DFB**.	Step3ControllerTags	\$STEP3CTL_CD	Refer to Three-Step Controller Management***.
	—	—	—	Step3Controllergenies	\$STEP3CTL_PVSP_CG	
					\$STEP3CTL_LSP_CG	
Optional I/O signals						
OPIncDOSignal *		\$DOSignal_UL	Refer to the OPINC pin of STEP3CTL DFB**.	—	—	—
OPDecDOSignal *		\$DOSignal_UL	Refer to the OPDEC pin of STEP3CTL DFB**.	—	—	—
Optional services						

Control services (\$STEP3Controller_UC)	Corresponding facet template	Control service description	Associated Supervision services (\$STEP3Controller_CS)*	Corresponding facet template	Supervision service description
Interlocks *	\$CONDSUM1_UL	Refer to the CONDSUM1 DFB**.	InterlockTags *	\$CONDSUM1_CD	
<p>* The service is activated by default.</p> <p>** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual</p> <p>*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual</p>					

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
OPINC	\$Bool/Def	EDT interface
OPDEC	\$Bool/Ref	EDT interface
ILCK	\$Bool/Ref	EDT interface
PV	\$Real/Ref	EDT interface
RSP	\$Real/Ref	EDT interface
PVRange	\$Range/Ref	EDT interface
PVRanged	\$RangedValue/Ref	EDT interface
DEV	\$Real/Def	EDT interface
SP	\$Real/Def	EDT interface
OPIncVar	\$Bool/Def	EDT interface
OPDecVar	\$Bool/Def	EDT interface
Step3CtlName	\$STEP3CTL_IF/Def	Instance name concatenated with control block name

Sequential Control

\$SequenceDFB - Sequence Advanced Control

General Description

The \$SequenceDFB process control module template is used to monitor and manage control sequences.

These templates do not reflect any specific installation.

⚠ WARNING

LOSS OF CONTROL

- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.¹
- Test each implementation of a system for proper operation before placing it into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

¹ For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

Modifying the Sections of the DFB Type Managing the Subsequences

To use sequential control, you need to modify the sections of the DFB type that manages the subsequences (for example, *RUNNING*).

You can use the **USERSEQ_DFB Type** parameter, page 98 to create a user-defined DFB type.

For more information on how to modify the sections of the DFB type, refer to the sequential control function description and example use case (see EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual).

Parameters

Configuration

The table describes the **Configuration** parameters that you can configure:

Element Name	Name	Type	Default value	Description
Control	Number of Parameters	Integer	5	<p>Number of parameters that the sequence manages.</p> <p>Depending on the value that you enter, the following services are activated in addition to the services that are not parameter-dependent:</p> <ul style="list-style-type: none"> 0: Only <i>UserSeq0Par</i> (Control). 1 to 5: <ul style="list-style-type: none"> <i>5Parameters</i> (Control) <i>UserSeq5Par</i> (Control) <i>Parameters1to5</i> (Supervision) 6 to 10: <ul style="list-style-type: none"> <i>10Parameters</i> (Control) <i>UserSeq10Par</i> (Control) <i>Parameters6to10</i> (Supervision) 11 and higher: <ul style="list-style-type: none"> <i>16Parameters</i> (Control) <i>UserSeq16Par</i> (Control) <i>Parameters11to16</i> (Supervision)
	USERSEQ_DFB Type	String	Null	<p>Name of the DFB type that is generated by the <i>\$USERSEQxx_UL</i> facet, where <i>xx</i> is 0, 5, 10, or 16 depending on the value of Number of Parameters.</p> <p>If you leave the parameter blank, it generates a DFB type, which as the identifier of the instance as name. In any case, an instance of the DFB type is created in the section.</p> <p>NOTE: In any case, a <i>USERSEQxx</i> DFB type is created but not used (where <i>xx</i> is 0, 5, 10, or 16 depending on the Number of Parameters).</p>
Genies	Advseqctl_PAR1 and Advseqctl_PAR2	String	Null	<p>Parameters that are displayed on the genie (description and value).</p> <p>Enter the name of up to two variables that are used respectively as <i>Data 1</i> and <i>Data 2</i> of the genie properties.</p>
	Advseqctl_x	Boolean	0	<p>1 = Hides the corresponding button on the genie and faceplate.</p> <p><i>x</i> represents the name of the command. For example, Advseqctl_Stop.</p>
	Advseqctl_S1TXT to Advseqctl_S8TXT	String	Null	<p>Name of the strategy to appear on the Supervision components during runtime, where <i>S1</i> represents strategy 1, <i>S2</i> strategy 2, and so on.</p>
	Advseqctl_S1PAR to Advseqctl_S8PAR	String	Null	<p>Parameters that are applied to the strategy that is defined by the Advseqctl_SxTXT parameter.</p> <p>Requires a list of comma-separated parameter numbers. Parameters from 2 to 16.</p> <p>NOTE: Parameter 1 is reserved for the strategy when the feature is supported and enabled. When it is disabled, the parameter can be used as a regular parameter.</p>

Historize

The table describes the **Historize** parameters that you can configure:

Element Name	Name	Type	Default value	Description				
SEQCTL1-Tags	Historize_x	Boolean	0	<p>Corresponds to the <i>Historize</i> property of advanced alarm tags of the Supervision Participant.</p> <p>1 = Sets the <i>Historize</i> property of the corresponding tag to <i>True</i>.</p> <p>x represents the child elements of the <i>SEQCTL1_ST</i> and <i>SEQCTL1_CFG</i> data structures, and the <i>SEQCTL1.CSTEPD</i> variable.</p> <p>Refer to <i>Advanced Alarm Properties</i> in the help of the Supervision Participant for details.</p>				
Parameter-sXtoY	Historize_STW	Boolean	0	<p>Corresponds to the <i>Historize</i> property of advanced alarm tags of the Supervision Participant.</p> <p>1 = Sets the <i>Historize</i> property of the corresponding tag to <i>True</i>.</p> <p>Refer to <i>Advanced Alarm Properties</i> in the help of the Supervision Participant for details.</p>				
	Historize_CFGW							
	Historize_IPX to Historize_IPY	Boolean	0	<p>Corresponds to the <i>Historize</i> property of advanced alarm tags of the Supervision Participant.</p> <p>1 = Sets the <i>Historize</i> property of the corresponding tag to <i>True</i>.</p> <p>Refer to <i>Advanced Alarm Properties</i> in the help of the Supervision Participant for details.</p>				
	Historize_OPX to Historize_OPY							
	Historize_RPTX to Historize_RPTY							
			<table><tr><td>STW</td><td>Status word</td></tr><tr><td>CFGW</td><td>Configuration word</td></tr></table>	STW	Status word	CFGW	Configuration word	
	STW	Status word						
CFGW	Configuration word							
		<table><tr><td>IP</td><td>Input parameter</td></tr><tr><td>OP</td><td>Output parameter</td></tr><tr><td>RPT</td><td>Report parameter</td></tr></table>	IP	Input parameter	OP	Output parameter	RPT	Report parameter
IP	Input parameter							
OP	Output parameter							
RPT	Report parameter							

NOTE: For example, if the value of *Number of Parameters* is 7, only elements *Parameters1to5* and *Parameters6to10* are created with editable parameters.

Input Parameter Configuration

The table describes the parameters of **Input Parameter Configuration** category that you can configure,

Element Name	Name	Type	Default value	Description
ParametersXtoY	IPX_Description to IPY_Description	String	Null	Text that appears on the corresponding Supervision components where the input parameter value is displayed during runtime.
	IPX_Format to IPY_Format	Boolean	#####EU	Engineering unit format that is used to display the corresponding input parameter value.
NOTE: For example, if the value of <i>Number of Parameters</i> is 7, only elements <i>Parameters1to5</i> and <i>Parameters6to10</i> are created with editable parameters.				

Output Parameter Configuration

The table describes the parameters of **Output Parameter Configuration** category that you can configure,

Element Name	Name	Type	Default value	Description
ParametersXtoY	OPX_Description to OPY_Description	String	Null	Text that appears on the corresponding Supervision components where the output parameter value is displayed during runtime.

Element Name	Name	Type	Default value	Description
	OPX_Format to OPY_Format	Boolean	#####EU	Engineering unit format that is used to display the corresponding output parameter value.
NOTE: For example, if the value of <i>Number of Parameters</i> is 7, only elements <i>Parameters1to5</i> and <i>Parameters6to10</i> are created with editable parameters.				

Report Parameter Configuration

The table describes the parameters of **Report Parameter Configuration** category that you can configure,

Element Name	Name	Type	Default value	Description
ParametersXtoY	RPTX_Description to RPTY_Description	String	Null	Text that appears on the corresponding Supervision components where the parameter value is displayed during runtime.
	RPTX_Format to RPTY_Format	Boolean	#####EU	Engineering unit format that is used to display the corresponding parameter value.

NOTE: Refer to *Sequential Control Management* for details on which genies support the various features.

The following optional elements feature configurable parameters:

- : Messages associated to initial conditions, to appear on the Initial Conditions tab of the faceplate for sequential control management.
-

Composition

The `$SequenceDFB` template references composite and facet templates, which provide the following services:

Control	Core services, plus additional optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$SequenceDFB` control module and the corresponding facet, which implements the service:

Control services (<code>\$SequenceDFB_UC</code>)	Corresponding facet template	Control service description	Associated Supervision services (<code>\$SequenceDFB_CS</code>) *	Corresponding facet template	Supervision service description
Core services					
<i>Logic</i>	<code>\$SEQCTL1_UL</code>	Refer to the <code>SEQCTL1 DFB</code> **.	<code>SEQCTL1-Tags</code>	<code>\$SEQCTL1_CD</code>	Refer to Sequential Control Management***.
–	–	–	Genies	<code>\$ADV-SEQCTL_CG</code>	
				<code>\$ADV-SEQCTL1_CG</code>	
				<code>\$ADV-SEQCTL2_CG</code>	
				<code>\$ADV-SEQCTL3_CG</code>	

Control services (\$SequenceDFB_ UC)	Corresponding facet template	Control service description	Associated Supervision services (\$Sequen- ceDFB_CS) *	Corre- sponding facet template	Supervision service description														
				\$ADV- SEQCTL4_ CG															
Optional Services																			
xParameters ⁽¹⁾	\$SEQPAR05_UL ⁽³⁾	Refer to the SEQPARxx DFB**.	Parameter- s1to5	\$SEQ- PAR05_CD (4)	–														
	\$SEQPAR10_UL ⁽³⁾		Parameter- s6to10	\$SEQ- PAR10_CD (4)															
	\$SEQPAR16_UL ⁽³⁾		Parameter- s11to16	\$SEQ- PAR16_CD (4)															
UserSeqxPar ⁽²⁾	\$USERSEQ0_UL ⁽³⁾	–	–	–	–														
	\$USERSEQ5_UL ⁽³⁾		–	–															
	\$USERSEQ10_UL ⁽³⁾		–	–															
	\$USERSEQ16_UL ⁽³⁾		–	–															
InitialConditions	\$CONDSUM_UL	Refer to the CONDSUM DFB**.	InitialCondi- tionTags *	\$INIT- CONDSUM_ CD	–														
FailureConditions	\$CONDSUM_UL	Refer to the CONDSUM DFB**.	FailureCondi- tionTags *	\$CON- DSUM_CD															
<table><tr><td>*</td><td>The service is activated by default.</td></tr><tr><td>**</td><td>See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual</td></tr><tr><td>***</td><td>See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual</td></tr><tr><td>(1)</td><td>Where x is either 5, 10, or 16, depending on the number of parameters that you configure.</td></tr><tr><td>(2)</td><td>Where x is either 0, 5, 10, or 16, depending on the number of parameters that you configure.</td></tr><tr><td>(3)</td><td>Only 1 facet for each service is created. It is the one that encompasses the number of parameters entered. For example, when <i>Number of Parameters</i> is 7, \$SEQPAR10_UL and \$USERSEQ10_UL are created.</td></tr><tr><td>(4)</td><td>As many facets as necessary to encompass the number of parameters entered are created. For example, when <i>Number of Parameters</i> is 7, \$SEQPAR05_CD and \$SEQPAR10_CD are created.</td></tr></table>						*	The service is activated by default.	**	See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual	***	See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual	(1)	Where x is either 5, 10, or 16, depending on the number of parameters that you configure.	(2)	Where x is either 0, 5, 10, or 16, depending on the number of parameters that you configure.	(3)	Only 1 facet for each service is created. It is the one that encompasses the number of parameters entered. For example, when <i>Number of Parameters</i> is 7, \$SEQPAR10_UL and \$USERSEQ10_UL are created.	(4)	As many facets as necessary to encompass the number of parameters entered are created. For example, when <i>Number of Parameters</i> is 7, \$SEQPAR05_CD and \$SEQPAR10_CD are created.
*	The service is activated by default.																		
**	See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual																		
***	See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual																		
(1)	Where x is either 5, 10, or 16, depending on the number of parameters that you configure.																		
(2)	Where x is either 0, 5, 10, or 16, depending on the number of parameters that you configure.																		
(3)	Only 1 facet for each service is created. It is the one that encompasses the number of parameters entered. For example, when <i>Number of Parameters</i> is 7, \$SEQPAR10_UL and \$USERSEQ10_UL are created.																		
(4)	As many facets as necessary to encompass the number of parameters entered are created. For example, when <i>Number of Parameters</i> is 7, \$SEQPAR05_CD and \$SEQPAR10_CD are created.																		

NOTE:

- When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.
- You have to provide descriptions for enabling their respective tags. Also, you need to define integer value in the parameter called **Number of Parameters** for enabling the number of sequences that has to be managed. Hence, for values 1 to 5, SEQPAR05 DFB will be enabled. For values 6 to 10, SEQPAR10 DFB will be enabled and for numbers greater than 10 SEQPAR16 DFB will be enabled.

The control module template exposes the following interface.

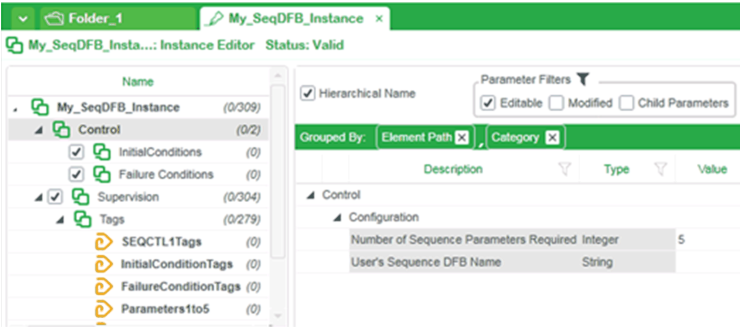

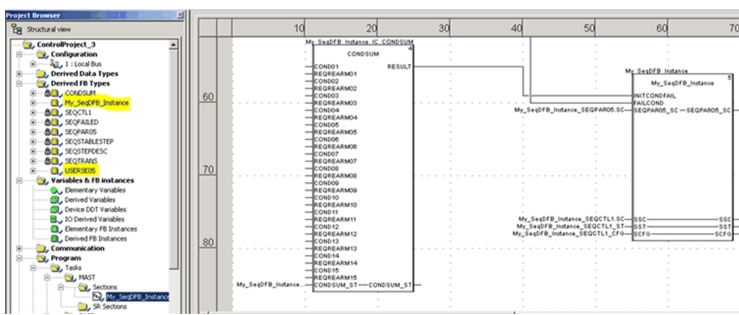
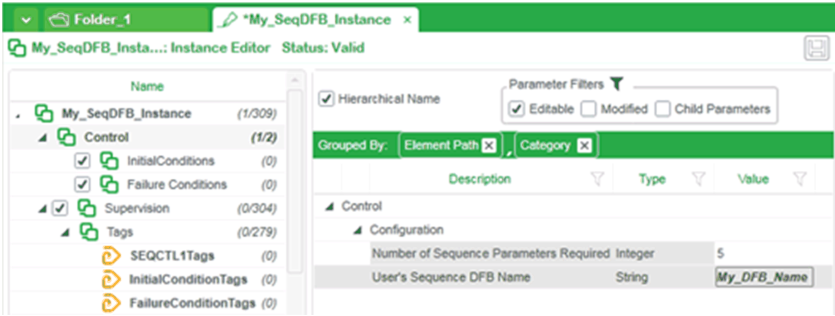
Interface Identifier	Type/Role	Description
SeqCtl1Name	\$SEQCTL1Name_IF/Def	Instance name concatenated with control block name

Using the \$SequenceDFB Template

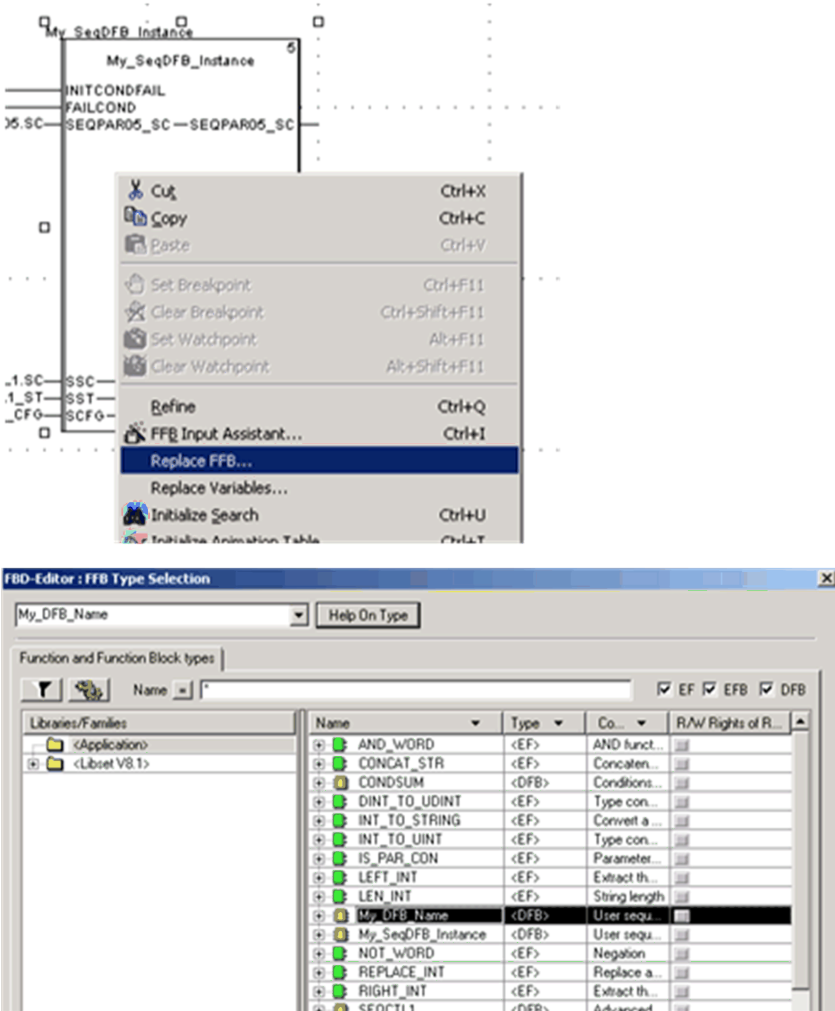
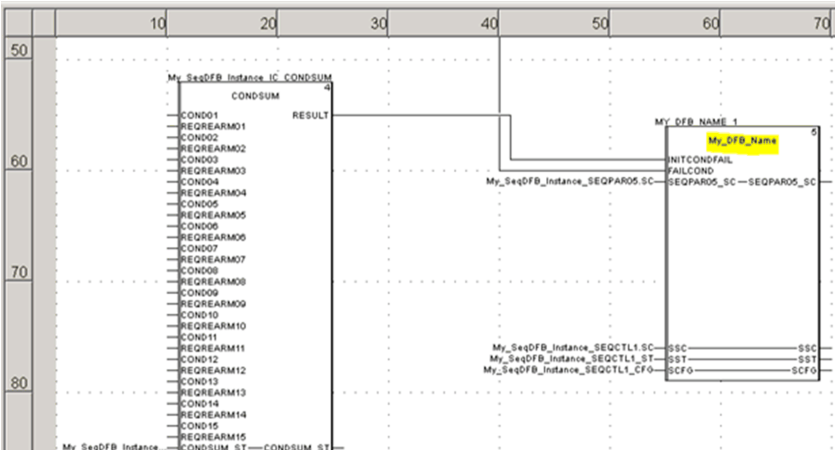
Overview

This topic explains how to use the `$SequenceDFB` template instance with the help of two examples.

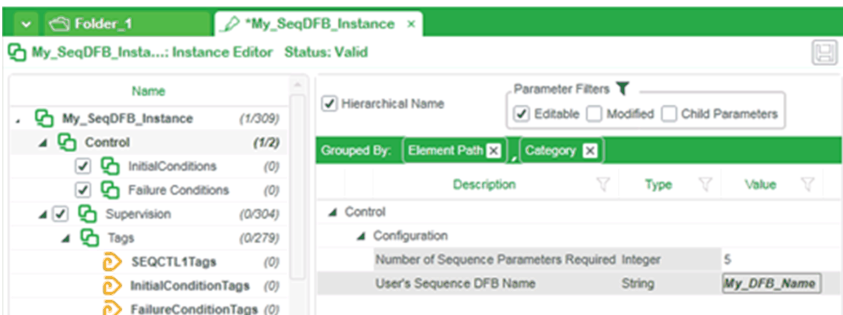
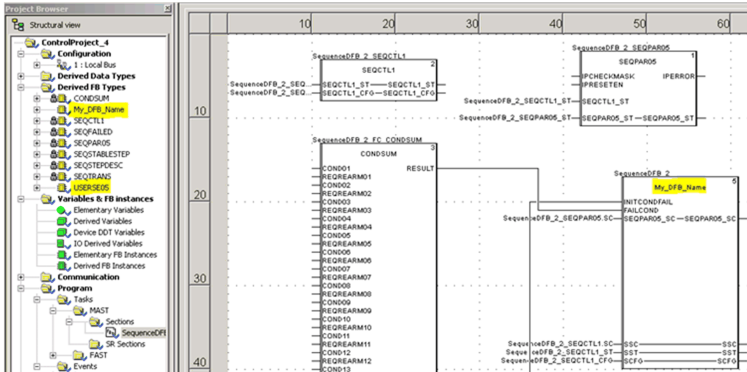
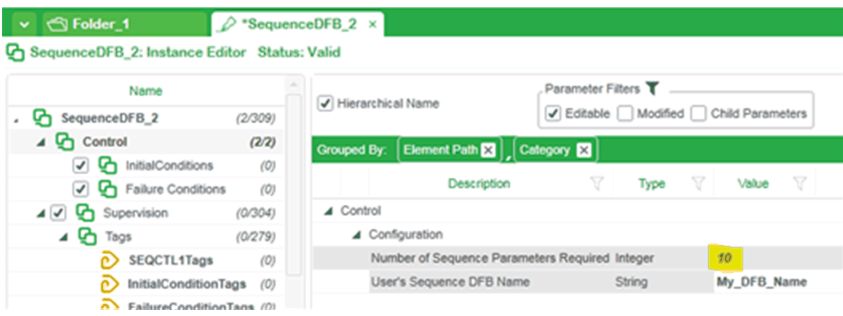

Example 1: Instantiation with Default Configuration and Subsequent Customization of the Type Name

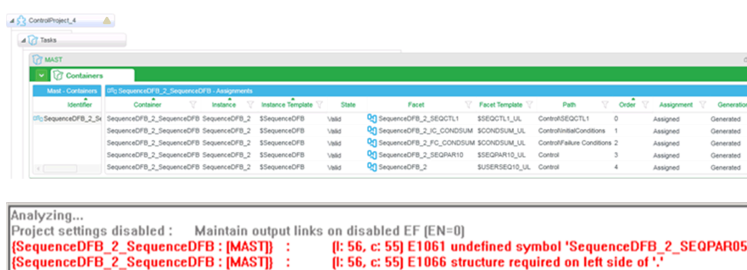
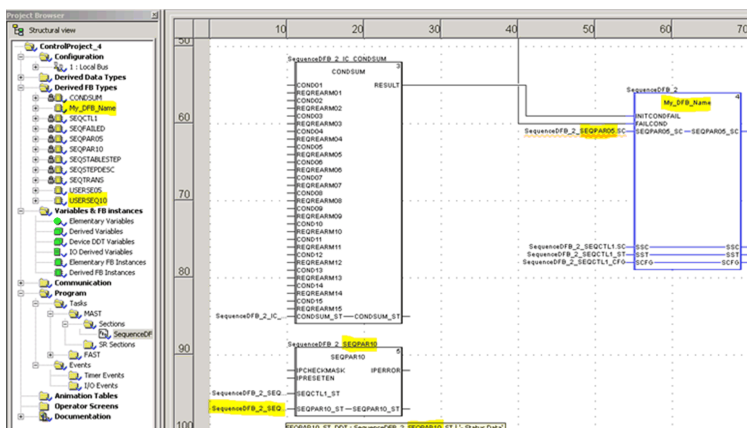
Step	Action
1	<p>Instance identifier: <code>My_SeqDFB_Instance</code>. Default type name (the User's <code>SequenceDFB</code> Name parameter is left blank) and default number of parameters.</p> 
2	<p>Assign facets of <code>My_SeqDFB_Instance</code>.</p> 
3	<p>Generate the project.</p> <p>a The default sequence DFB type <code>USERSE05</code> is created but not used.</p> <p>b A clone of the <code>USERSE05</code> type is created, which has the identifier of the instance as name (<code>My_SeqDFB_Instance</code>) and an instance of this DFB type is created.</p> 
4	<p>Modify the type name in the Instance Editor.</p> 

Step	Action																																																											
5	Impact on assignments.																																																											
	<p>a</p> <p>The facet that contains the USERSE05 DFB type goes to Out Of Date.</p> <table border="1"> <thead> <tr> <th>Container</th><th>Instance</th><th>Instance Template</th><th>State</th><th>Facet</th><th>Facet Template</th><th>Path</th><th>Order</th><th>Assignment</th><th>Generation</th></tr> </thead> <tbody> <tr> <td>My_SeqDFB_Instance_SequenceDFB</td><td>My_SeqDFB_Instance_SequenceDFB</td><td>My_SeqDFB_Instance_SequenceDFB</td><td>Valid</td><td>My_SeqDFB_Instance_SEQCTL1</td><td>SEQCTL1_UL</td><td>Control/SEQCTL1</td><td>0</td><td>Assigned</td><td>Generated</td></tr> <tr> <td>My_SeqDFB_Instance_SequenceDFB</td><td>My_SeqDFB_Instance_SequenceDFB</td><td>My_SeqDFB_Instance_SequenceDFB</td><td>Valid</td><td>My_SeqDFB_Instance_SEQCONDUM</td><td>SEQCONDUM_UL</td><td>Control/InitialConditions</td><td>1</td><td>Assigned</td><td>Generated</td></tr> <tr> <td>My_SeqDFB_Instance_SequenceDFB</td><td>My_SeqDFB_Instance_SequenceDFB</td><td>My_SeqDFB_Instance_SequenceDFB</td><td>Valid</td><td>My_SeqDFB_Instance_SEQPARMS</td><td>SEQPARMS_UL</td><td>Control/Failure Conditions</td><td>2</td><td>Assigned</td><td>Generated</td></tr> <tr> <td>My_SeqDFB_Instance_SequenceDFB</td><td>My_SeqDFB_Instance_SequenceDFB</td><td>My_SeqDFB_Instance_SequenceDFB</td><td>Valid</td><td>My_SeqDFB_Instance</td><td>USERSE05_UL</td><td>Control</td><td>3</td><td>Assigned</td><td>Generated</td></tr> <tr> <td>My_SeqDFB_Instance_SequenceDFB</td><td>My_SeqDFB_Instance_SequenceDFB</td><td>My_SeqDFB_Instance_SequenceDFB</td><td>Valid</td><td>My_SeqDFB_Instance</td><td>USERSE05_UL</td><td>Control</td><td>4</td><td>Out Of Date</td><td>Generated</td></tr> </tbody> </table>	Container	Instance	Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generation	My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	Valid	My_SeqDFB_Instance_SEQCTL1	SEQCTL1_UL	Control/SEQCTL1	0	Assigned	Generated	My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	Valid	My_SeqDFB_Instance_SEQCONDUM	SEQCONDUM_UL	Control/InitialConditions	1	Assigned	Generated	My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	Valid	My_SeqDFB_Instance_SEQPARMS	SEQPARMS_UL	Control/Failure Conditions	2	Assigned	Generated	My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	Valid	My_SeqDFB_Instance	USERSE05_UL	Control	3	Assigned	Generated	My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	Valid	My_SeqDFB_Instance	USERSE05_UL	Control	4	Out Of Date
Container	Instance	Instance Template	State	Facet	Facet Template	Path	Order	Assignment	Generation																																																			
My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	Valid	My_SeqDFB_Instance_SEQCTL1	SEQCTL1_UL	Control/SEQCTL1	0	Assigned	Generated																																																			
My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	Valid	My_SeqDFB_Instance_SEQCONDUM	SEQCONDUM_UL	Control/InitialConditions	1	Assigned	Generated																																																			
My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	Valid	My_SeqDFB_Instance_SEQPARMS	SEQPARMS_UL	Control/Failure Conditions	2	Assigned	Generated																																																			
My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	Valid	My_SeqDFB_Instance	USERSE05_UL	Control	3	Assigned	Generated																																																			
My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	My_SeqDFB_Instance_SequenceDFB	Valid	My_SeqDFB_Instance	USERSE05_UL	Control	4	Out Of Date	Generated																																																			
6	Generate the project.																																																											
	a																																																											
	Both USERSE05 and <i>My_SeqDFB_Instance</i> DFB types remain.																																																											
b	A new type is created, which has the name you entered in the Instance Editor: <i>My_DFB_Name</i> .																																																											
	c																																																											
	The DFB instance is not replaced inside the section.																																																											

Step	Action
7	<p>Solution: Replace the DFB inside the section manually to use the <i>My_DFB_Name</i> type.</p>  <p>The screenshot shows the FBD-Editor interface. A context menu is open for the 'My_SeqDFB_Instance' block. The menu options include Cut, Copy, Paste, Set Breakpoint, Clear Breakpoint, Set Watchpoint, Clear Watchpoint, Refine, FFB Input Assistant..., Replace FFB..., Replace Variables..., Initialize Search, and Initialize Animation Table. The 'Replace FFB...' option is highlighted. Below the menu, the 'FBD-Editor : FFB Type Selection' dialog is displayed. It shows a list of function blocks with columns for Name, Type, Co., and RAW Rights of R. The 'My_DFB_Name' block is selected in the list.</p>
8	<p>Result:</p>  <p>The screenshot shows the FBD-Editor interface after the replacement. The 'My_SeqDFB_Instance' block is now labeled 'My_DFB_Name' and is connected to the 'COND01' block. The 'COND01' block is part of a larger structure labeled 'My_SeqDFB_Instance_IC_CONDSUM'. The 'COND01' block is connected to the 'COND02' block, which is connected to the 'COND03' block, and so on. The 'COND01' block is also connected to the 'COND01' block of the 'My_SeqDFB_Instance_IC_CONDSUM' structure. The 'COND01' block is also connected to the 'COND01' block of the 'My_SeqDFB_Instance_IC_CONDSUM' structure. The 'COND01' block is also connected to the 'COND01' block of the 'My_SeqDFB_Instance_IC_CONDSUM' structure.</p>

Example 2: Instantiation with Custom Type Name and Subsequent Modification of Parameters

Step	Action
1	<p>Modify the type name in the Instance Editor: <i>My_DFB_Name</i>.</p> 
2	<p>Assign the facets of the instance and generate the project.</p> <p>a The default sequence DFB type USERSE05 is created but not used.</p> <p>b A clone of the USERSE05 type is created, which has the name that you have entered in the Instance Editor (<i>My_DFB_Name</i>) and an instance of this DFB type is created.</p> 
3	<p>Modify the number of parameters from 5 to 10 in the Instance Editor.</p> 
4	<p>Assign the new facets.</p> <p>a Two new facets are created, which generate code related to the new number of parameters.</p> <p>b The two facets that generate code related to the previous number of parameters (5) are deleted.</p> 
5	<p>Generate the project.</p> <p>a The analysis of the project detects an error.</p>

Step	Action																																
	<div></div> <p>Analyzing...</p> <p>Project settings disabled : Maintain output links on disabled EF (EN=0)</p> <p>(SequenceDFB_2_SequenceDFB : [MAST]) : (l: 56, c: 55) E1061 undefined symbol 'SequenceDFB_2_SEQPAR05'</p> <p>(SequenceDFB_2_SequenceDFB : [MAST]) : (l: 56, c: 55) E1066 structure required on left side of '!'</p>																																
6	<p>Refine the project.</p> <p>a A new sequence DFB type that corresponds to the new number of parameters is created (USERSEQ10) but is not used.</p> <p>b <i>My_DFB_Name</i> type remains and its instance is not replaced inside the section.</p> <p>c The DFB that manages the parameters (<i>SEQPAR05</i>, see step 2) has been replaced by a new DFB (<i>SEQPAR10</i>) to manage the new number of parameters. A new variable has been created (<i>SequenceDFB_2_SEQPAR10_ST</i>).</p> <div></div> <p>d However, the variable reference on <i>My_DFB_Name</i> has not been updated and still refers to the previous one (<i>SequenceDFB_2_SEQPAR05.SC</i>), which was created to manage 5 parameters. <i>My_DFB_Name</i> will not be managed by EcoStruxure Process Expert. User has to manually change the required references to support 10 parameters.</p>																																
7	<p>Solution:</p> <p>a Refine the project, goto DataEditor window and select DFBTypes.</p> <p>b Expand inputs/outputs. Replace <i>SEQPAR05_SC</i> with <i>SEQPAR10_SC</i> and datatype <i>SEQPAR05_SC_DDT</i> with <i>SEQPAR10_SC_DDT</i>.</p> <table><tr><th colspan="3">My_DFB_Name</th><th><DFB></th></tr><tr><td colspan="3"><inputs></td><td></td></tr><tr><td colspan="3"><outputs></td><td></td></tr><tr><td colspan="3"><inputs/outputs></td><td></td></tr><tr><td>+</td><td>SEQPAR10_SC</td><td>3</td><td>SEQPAR10_SC_DDT</td></tr><tr><td>+</td><td>SSC</td><td>18</td><td>SEQCTL1_SC_DDT</td></tr><tr><td>+</td><td>SST</td><td>19</td><td>SEQCTL1_ST_DDT</td></tr><tr><td>+</td><td>SCFG</td><td>20</td><td>SEQCTL1_CFG_DDT</td></tr></table> <p>c Analyze the DFB.</p> <p>d Find the <i>MY_DFB_Name</i> with the above changes under section. Edit variable to <i>SequenceDFB_2_SEQPAR10.SC</i>.</p> <p>e Build the project.</p>	My_DFB_Name			<DFB>	<inputs>				<outputs>				<inputs/outputs>				+	SEQPAR10_SC	3	SEQPAR10_SC_DDT	+	SSC	18	SEQCTL1_SC_DDT	+	SST	19	SEQCTL1_ST_DDT	+	SCFG	20	SEQCTL1_CFG_DDT
My_DFB_Name			<DFB>																														
<inputs>																																	
<outputs>																																	
<inputs/outputs>																																	
+	SEQPAR10_SC	3	SEQPAR10_SC_DDT																														
+	SSC	18	SEQCTL1_SC_DDT																														
+	SST	19	SEQCTL1_ST_DDT																														
+	SCFG	20	SEQCTL1_CFG_DDT																														

Batch Phase Manager

\$IBPhase - Individual Batch Phase Manager

General Description

The \$IBPhase process control module template is used to monitor and manage batch phases.

These templates do not reflect any specific installation.

⚠ WARNING

LOSS OF CONTROL

- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.¹
- Test each implementation of a system for proper operation before placing it into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

¹ For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

Modifying the Sections of the DFB Type Managing the Sub phases

To use batch phase control, you need to modify the sections of the DFB type that manages the subphases (for example, *RUNNING*).

You can use the **USERPHASE_DFBType** parameter, page 108 to create a user-defined DFB type.

NOTE: The entry has to be compliant with naming convention of Control Expert.

For more information on how to modify the sections of the DFB type, refer to the InBatch control function description and example use case (EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual).

Parameters

Configuration

The table describes the **Configuration** parameters that you can configure:

Element Name	Name	Type	Default value	Description
Control	Maximum number of phase parameters	Enum	Upto 5	<p>Maximum number of phase parameters that the phase manages.</p> <p>Depending on the value that you select, the following services are activated in addition to the services that are not parameter-dependent:</p> <ul style="list-style-type: none"> No parameters: Only <i>UserPhase0Par</i> (Control). <p>NOTE: With configurable buttons and data, and with control and data genies are not available if this selection is made.</p> Upto 5: <ul style="list-style-type: none"> <i>5Parameters</i> (Control) <i>UserPhase5Par</i> (Control) <i>Parameters1to5</i> (Supervision) Upto 10: <ul style="list-style-type: none"> <i>10Parameters</i> (Control) <i>UserPhase10Par</i> (Control) <i>Parameters1to5</i> and <i>Parameters6to10</i> (Supervision) Upto 16: <ul style="list-style-type: none"> <i>16Parameters</i> (Control) <i>UserPhase16Par</i> (Control) <i>Parameters1to5</i>, <i>Parameters6to10</i> and <i>Parameters11to16</i> (Supervision)
	USERPHASE_DFB Type	String	Null	<p>Name of the DFB type that is generated by the <i>\$USERPHASExx_UL</i> facet, where <i>xx</i> is 00, 05, 10 or 16 depending on the value of Maximum number of phase parameters.</p> <p>If you leave the parameter blank, it generates a DFB type, which has the identifier of the instance as name. In any case, an instance of the DFB type is created in the section.</p> <p>NOTE:</p> <ul style="list-style-type: none"> In any case, a <i>\$USERPHASExx_UL</i> DFB type is created but not used (where <i>xx</i> is 00, 05, 10 or 16 depending on the Maximum number of phase parameters). You have to verify that the parameter is configured prior to the generation of the respective instance.
	AutoResetDis	Boolean	FALSE	Disable auto reset.
Supervision	Communication Approach	Enum	Through OFS	InBatch communication with IBPhase manager. For, additional details refer Batch Phase Manager (see EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual).
Genies	IBPHASE_PAR1 and IBPHASE_PAR2	String	Null	<p>Parameters that are displayed on the genie (description and value).</p> <p>Enter the name of up to two variables that are used respectively as <i>Data 1</i> and <i>Data 2</i> of the genie properties.</p> <p>NOTE: The configured string should be available as variable tag in the project.</p>
	IBPHASE CB1_Label and IBPHASE CB2_Label	String	Null	<p>Control Button and its labels are displayed on both control button genie and faceplate.</p> <p>NOTE: If Control Button label is not configured then buttons will not be available for operation.</p>
	IBPHASE_x	Boolean	0	1 = Hides the corresponding button on the genie and faceplate.

Element Name	Name	Type	Default value	Description
				x represents (Start, Pause, Hold, Abort, Stop, Restart, Reset) the name of the command. For example, IBPhase_Stop .
Strategy Parameters	IBPHASE_S1TXT to IBPHASE_S8TXT	String	Null	Name of the strategy to appear on the Supervision components during runtime, where S1 represents strategy 1, S2 strategy 2, and so on.
	IBPHASE_S1PAR to IBPHASE_S8PAR	String	Null	Parameters that are applied to the strategy that is defined by the IBPhase_SxTXT parameter. Requires a list of comma-separated parameter numbers. Parameters from 2 to 16. NOTE: Parameter 1 is reserved for the strategy when the feature is supported and enabled. When it is disabled, the parameter can be used as a regular parameter.

Historize

The table describes the **Historize** parameters that you can configure:

Element Name	Name	Type	Default value	Description				
IBPhaseTags	Historize_x	Boolean	0	Corresponds to the <i>Historize</i> property of advanced alarm tags of the Supervision Participant. 1 = Sets the <i>Historize</i> property of the corresponding tag to <i>True</i> . x represents the child elements of the <i>IBPhase_ST</i> and <i>IBPhase_CFG</i> data structures, and the <i>IBPhase.CSTEPD</i> variable. Refer to <i>Advanced Alarm Properties</i> in the help of the Supervision Participant for details.				
	Historize_STW	Boolean	0	Corresponds to the <i>Historize</i> property of advanced alarm tags of the Supervision Participant. 1 = Sets the <i>Historize</i> property of the corresponding tag to <i>True</i> . Refer to <i>Advanced Alarm Properties</i> in the help of the Supervision Participant for details.				
Historize_CFGW								
Parameter-sXtoY	Historize_IPX to Historize_IPY	Boolean	0	Corresponds to the <i>Historize</i> property of advanced alarm tags of the Supervision Participant. 1 = Sets the <i>Historize</i> property of the corresponding tag to <i>True</i> . Refer to <i>Advanced Alarm Properties</i> in the help of the Supervision Participant for details.				
	Historize_OPX to Historize_OPY							
<table><tr><td>STW</td><td>Status word</td></tr><tr><td>CFGW</td><td>Configuration word</td></tr></table>					STW	Status word	CFGW	Configuration word
STW	Status word							
CFGW	Configuration word							
<table><tr><td>IP</td><td>Input parameter</td></tr><tr><td>OP</td><td>Output parameter</td></tr></table>					IP	Input parameter	OP	Output parameter
IP	Input parameter							
OP	Output parameter							

NOTE: For example, if the value of *Maximum number of phase parameters* is **upto 10**, only elements *Parameters1to5* and *Parameters6to10* are created with parameters that can be edited.

Input Parameter Configuration

The table describes the parameters of **Input Parameter Configuration** category that you can configure,

Element Name	Name	Type	Default value	Description
ParametersXtoY	IPX_Description to IPY_Description	String	Null	Text that appears on the corresponding Supervision components where the input parameter value is displayed during runtime. NOTE: If this field is null, then corresponding tag will not be created.
	IPX_Format to IPY_Format	Boolean	#####EU	Engineering unit and format that is used to display the corresponding input parameter value.
NOTE: For example, if the value of <i>Maximum number of phase parameters</i> is upto 10 , only elements <i>Parameters1to5</i> and <i>Parameters6to10</i> are created with parameters that can be edited.				

Output Parameter Configuration

The table describes the parameters of **Output Parameter Configuration** category that you can configure,

Element Name	Name	Type	Default value	Description
ParametersXtoY	OPX_Description to OPY_Description	String	Null	Text that appears on the corresponding Supervision components where the output parameter value is displayed during runtime. NOTE: If this field is null, then corresponding tag will not be created.
	OPX_Format to OPY_Format	Boolean	#####EU	Engineering unit and format that is used to display the corresponding output parameter value.
NOTE: For example, if the value of <i>Maximum number of phase parameters</i> is upto 10 , only elements <i>Parameters1to5</i> and <i>Parameters6to10</i> are created with editable parameters.				

NOTE: Refer to *Batch Phase Manager* for details on which genies support the various features.

The following optional elements feature configurable parameters:

- : Messages associated to initial conditions, to appear on the Initial Conditions tab of the faceplate for batch phase manager.
- : Messages associated to detected *failure* conditions, to appear on the *Failure* Conditions tab of the faceplate for batch phase manager.

NOTE: It is mandatory to configure conditions to populate message tags in `UIMessage.DBF`

Composition

The `$IBPhase` template references composite and facet templates, which provide the following services:

Control	Core services, plus additional optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$IBPhase` control module and the corresponding facet, which implements the service:

Control services (\$IBPhase_UC)	Corresponding facet template	Control service description	Associated Supervision services (\$IBPhase_CS)*	Corre- sponding facet template	Supervision service description
Core services					
Logic	\$IBPhase_UL	Refer to the IBPhase DFB**.	IBPhaseTags Genies	\$InBatch-Phase_CD	Refer to Batch Phase Manager***.
				\$IB-Phase1_CG	
				\$IB-Phase2_CG	
				\$IB-Phase3_CG	
				\$IB-Phase4_CG	
				\$IB-Phase5_CG	
				\$IB-Phase6_CG	
				Optional Services	
xParameters ⁽¹⁾	\$IBPAR05_UL ⁽³⁾	Refer to the IBPARXX DFB**.	Parameter-s1to5	\$IB-PAR05_CD ⁽⁴⁾	Refer to Batch Phase Manager***.
	\$IBPAR10_UL ⁽³⁾		Parameter-s6to10	\$IB-PAR10_CD ⁽⁴⁾	
	\$IBPAR16_UL ⁽³⁾		Parameter-s11to16	IBPAR16_CD ⁽⁴⁾	
USERPHASEx	\$USERPHASE0_UL ⁽³⁾	—	—	—	—
	\$USERPHASE5_UL ⁽³⁾		—	—	
	\$USERPHASE10_UL ⁽³⁾		—	—	
	\$USERPHASE16_UL ⁽³⁾		—	—	
InitialConditions	\$CONDSUM_UL	Refer to the CONDSUM DFB**.	InitialCondi- tionTags *	\$INIT- CONDSUM_CD	
FailureConditions	\$CONDSUM_UL	Refer to the CONDSUM DFB**.	FailureCondi- tionTags *	\$CON- DSUM_CD	
*		The service is activated by default.			
**		See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual			
***		See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual			
(1)		Where x is either 5, 10, or 16, depending on the number of parameters that you configure.			
(2)		Where x is either 0, 5, 10, or 16, depending on the number of parameters that you configure.			
(3)		Only 1 facet for each service is created. It is the one that encompasses the number of parameters entered. For example, when <i>Maximum number of phase parameters</i> is upto 10, \$IBPhase10_UL and \$USERPHASE10_UL are created.			
(4)		As many facets as necessary to encompass the number of phase parameters entered are created. For example, when <i>Maximum Number of Parameters</i> is upto 10, \$IBPhase05_CD and \$IBPhase10_CD are created.			

NOTE:

- When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.
- You have to provide descriptions for enabling their respective tags. Also, you need to select range of maximum parameter for enabling the number of sequences that has to be managed. Hence, for values up to 5, IBPAR05 DFB will be enabled. For values up to 10, IBPAR10 DFB will be enabled and for numbers greater than 10 IBPAR16 DFB will be enabled.

The control module template exposes the following interface.

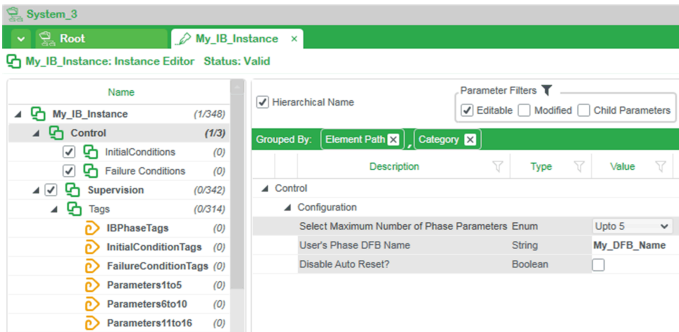
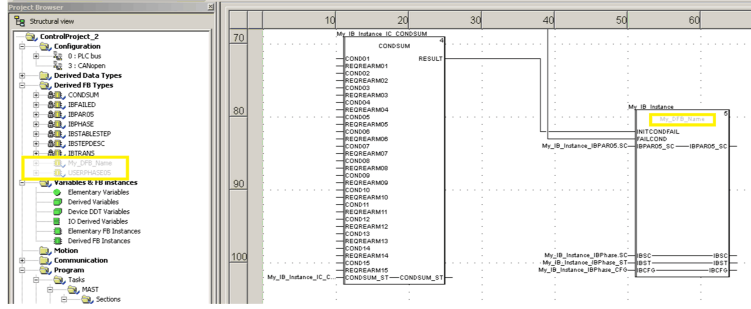
Interface Identifier	Type/Role	Description
IBPhaseName	\$IBPhaseName/Def	Instance name concatenated with control block name

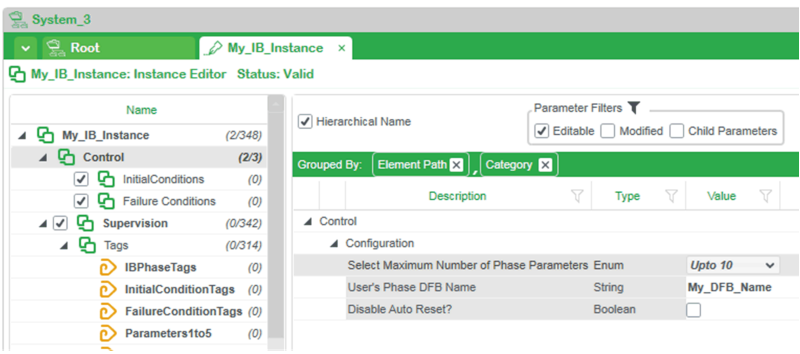
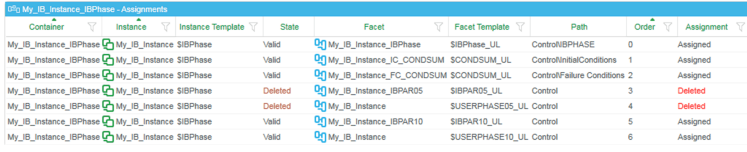
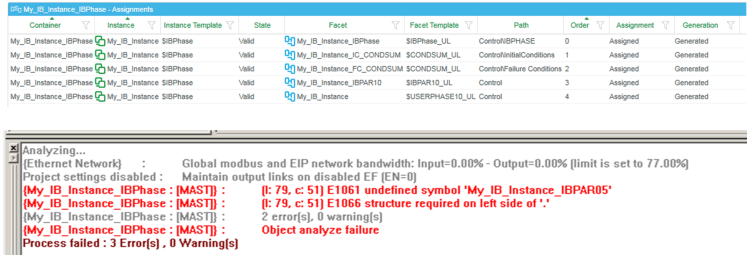
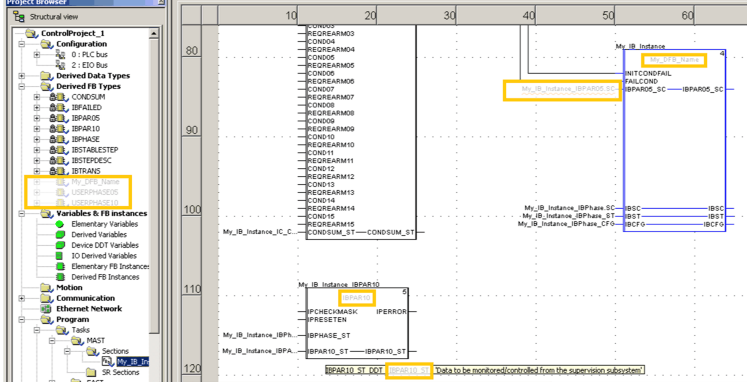
Using the \$IBPhase Template

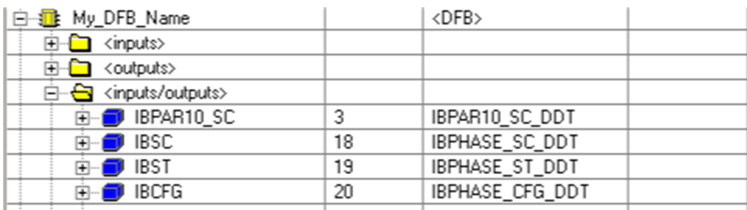
Overview

This topic explains how to use the *\$IBPhase* template instance with the help of given example.

Instantiation with Custom Type Name and Subsequent Modification of Parameters

Step	Action
1	<p>Modify the type name in the Instance Editor: <i>My_DFB_Name</i>.</p> 
2	<p>Assign the facets of the instance and generate the control project.</p> <p>a The default phase DFB type USERPHASE05 is created but not used.</p> <p>b A clone of the USERPHASE05 type is created, which has the name that you have entered in the Instance Editor (<i>My_DFB_Name</i>) and an instance of this DFB type is created.</p> <p>Go to refine offline of the control project to check the DFB name as show in the below screen shot.</p> 

Step	Action
3	<p>You can modify the maximum number of phase parameters in the Instance Editor (for example, Upto 10).</p> 
4	<p>Assign the new facets.</p> <p>a Two new facets are created, which generate code related to the new number of parameters.</p> <p>b The two facets that generate code related to the previous number of parameters are deleted.</p> 
5	<p>Generate the project.</p> <p>a The analysis of the project detects an error.</p> 
6	<p>Refine the project.</p> <p>a A new phase DFB type that corresponds to the new number of parameters is created (USERPHASE10) but is not used.</p> <p>b <i>My_DFB_Name</i> type remains and its instance is not replaced inside the section.</p> <p>c The DFB that manages the parameters (<i>IBPAR05</i>, see step 2) has been replaced by a new DFB (<i>IBPAR10</i>) to manage the new number of parameters. A new variable has been created (<i>My_IB_Instance_IBPAR10_ST</i>).</p>  <p>d However, the variable reference on <i>My_DFB_Name</i> has not been updated and still refers to the previous one (<i>My_IB_Instance_IBPAR05.SC</i>), which was created to manage 5 parameters. <i>My_DFB_Name</i> will not be managed by</p>

Step	Action
	EcoStruxure Process Expert. User has to manually change the required references to support 10 parameters.
7	<p>End Result:</p> <p>a Refine the project, go to DataEditor window and select DFBTypes.</p> <p>b Expand inputs/outputs. Replace <i>IBPAR05_SC</i> with <i>IBPAR10_SC</i> and datatype <i>IBPAR05_SC_DDT</i> with <i>IBPAR10_SC_DDT</i>.</p>  <p>c Analyze the DFB.</p> <p>d Find the <i>MY_DFB_Name</i> with the above changes under section. Edit variable to <i>My_IB_Instance_IBPAR05.SC</i>.</p> <p>e Build the project.</p>

Auxiliary Functions

Overview

This chapter describes the functionality of the auxiliary functions process templates and their composition.

These templates do not reflect any specific installation.

⚠ WARNING

LOSS OF CONTROL

- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.¹
- Test each implementation of a system for proper operation before placing it into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

¹ For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

\$AlarmSummary - Alarm Summary

General Description

The \$AlarmSummary process control module template is the combination of the \$CONDSUM_UC and the \$DigitalInput process templates.

It allows you to enable and disable the evaluation of alarms with up to 15 external alarm conditions.

Parameters

Configuration

The table describes the **Configuration** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
DigitalInput	Logic	BadPV	Boolean	1	1 = Sets the output signal (<i>PV</i>) to 1 when the channel is not operational.
		Watch	Enum	Null	Watch alarm
		AlarmST			PV state to be considered as alarm.
NOTE: Values of boolean parameters are set by using check boxes: <ul style="list-style-type: none">Selected = TrueCleared = False					

Time

The table describes the **Time** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
DigitalInput	Logic	TONSP	Duration	00:00:00	ON timer set point for digital input connection.
		TOFFSP			OFF timer set point for digital input disconnection.

Composition

The `$AlarmSummary` template references composite and facet templates, which provide the following services:

Control	Includes core services plus additional, optional services, which you can activate if needed.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$AlarmSummary` control module and the corresponding facet, which implements the service:

Control services (<code>\$AlarmSummary_UC</code>)		Composite template	Corresponding facet template	Control service description	Associated Supervision services (<code>\$AlarmSummary_CS</code>) *	Corresponding facet template	Supervision service description
Core services							
DigitalInput	Logic	<code>\$DINPUT_UC</code>	<code>\$DINPUT_UL</code>	Refer to the <code>DINPUT DFB</code> **.	DInputTags	<code>\$DINPUT_CD</code>	Refer to Diagnostic Information Management**.
	DISignal		<code>\$DISignal_UL</code>	Refer to the <code>DISignalCond1 EFB</code> **.	—	—	
	—	—	—	—	DInput genies	<code>\$DALARM_CG</code>	
Alarm		<code>\$CONDSUM_UC</code>	<code>\$CONDSUM_UL</code>	Refer to the <code>CONDSUM DFB</code> **.	AlarmTags	<code>\$CONDSUM_CD</code>	
<p>* The service is activated by default.</p> <p>** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual</p> <p>*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual</p>							

NOTE: When you activate an optional Control service, the corresponding Supervision service is automatically activated, if available.

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
Alarm	\$Bool/Def	EDT interface

\$AnalogSelect - Analog Signal Selector

General Description

The \$AnalogSelect process control module template is used to select an analog signal among a maximum of four signals depending on the chosen selection criterion.

You can link this process template, for example, with the \$PID, page 85 and \$AnalogOutput, page 31 process templates by using the Links Editor (see EcoStruxure™ Process Expert, User Guide).

Parameters

Configuration

The table describes the **Configuration** parameters that you can configure:

Element Name	Name	Type	Default value	Description
Logic	RSPSelection	UnsignedShort	0	Remote setpoint selection: <ul style="list-style-type: none"> 0 = Minimum of signals 1 = SP1 2 = SP2 3 = SP3 4 = SP4 5 = Maximum of signals

Composition

The \$AnalogSelect template references composite and facet templates, which provide the core Control services.

The table describes the services that are available from the \$AnalogSelect control module and the corresponding facet, which implements the service:

Control services		Composite template	Corresponding facet template	Control service description
AnalogSelect	Logic	\$Analog-Select_UC	\$ASELECT_UL	Refer to the ASELECT DFB (see EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual).

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
SP1	\$Real/Ref	EDT interface
SP2	\$Real/Ref	EDT interface
SP3	\$Real/Ref	EDT interface

Interface identifier	Type/role	Description
SP4	\$Real/Ref	EDT interface
SP	\$Real/Def	EDT interface

\$AnalogSelect1 - Analog Selector with Monitoring Interface

General Description

The \$AnalogSelect1 process control module template is used to select an analog signal among a maximum of four signals depending on the chosen selection criterion.

The process template has variables designed for monitoring the included DFB and for owner management purposes.

You can link this process template, for example, with the \$PID, page 85 and \$AnalogOutput, page 31 process templates by using the Links Editor (see EcoStruxure™ Process Expert, User Guide).

Parameters

Configuration

The table describes the **Configuration** parameter that you can configure:

Element Name	Name	Type	Default value	Description
Logic	RSPSelection	UnsignedShort	0	Remote setpoint selection: <ul style="list-style-type: none"> 0 = Minimum of signals 1 = SP1 2 = SP2 3 = SP3 4 = SP4 5 = Maximum of signals

Composition

The \$AnalogSelect1 template references composite and facet templates, which provide the following services:

Control	Core services.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the \$AnalogSelect1 control module and the corresponding facet, which implements the service:

Control services		Composite template	Corresponding facet template	Control service description	Associated Supervision services (\$Analog-Select1_CS) *	Corresponding facet template	Supervision service description
AnalogSelect1	Logic	\$ASELECT1_UC	\$ASELECT1_UL	Refer to the ASELECT1 DFB**.	Tags	\$ASELECT1_CD	Refer to Analog Signal Selection Management***.
	—	—	—	—	Genies	\$ASELECT1_CG	
						\$ASELECT1-SP_CG	
<p>* The service is activated by default.</p> <p>** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual</p> <p>*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual</p>							

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
SP1	\$Real/Ref	EDT interface
SP2	\$Real/Ref	EDT interface
SP3	\$Real/Ref	EDT interface
SP4	\$Real/Ref	EDT interface
SP	\$Real/Def	EDT interface

\$MessageBox - Messages to the Operator

General Description

The \$MessageBox process control module template offers a standard user interface mechanism and allows messages for the operator to be displayed. It allows requesting confirmations and/or data without having to configure customized interfaces to channel the messages.

The template, when combined with other templates designed for this purpose, allows you to configure the format of the message from the program in the Control Participant. The format can include icons, buttons, and data entry fields.

Parameter Table

The \$MessageBox template features no configurable parameters other than the general \$System parameters (\$Name, \$Description, and \$Area).

Composition

The \$MessageBox template references composite and facet templates, which provide the following services:

Control	Core services.
Supervision	These services complement the Control services. Supervision services are optional.

The table describes the services that are available from the `$MessageBox` control module and the corresponding facet, which implements the service:

Control services	Composite template	Corresponding facet template	Control service description	Associated Supervision services (<code>\$MessageBox_CS</code>)	Corresponding facet template	Supervision service description
Logic	<code>\$MSGBOX_UC</code>	<code>\$MSGBOX_UL</code>	Refer to the <code>MSGBOX DFB</code> **.	Tags	<code>\$MSGBOX_X_CD</code>	Refer to Operator Message Management***.
–	–	–	–	Genies	<code>\$MSGBOX_X_CG</code>	
					<code>\$MSGBOX_XLARGE_CG</code>	
					<code>\$MSGBOX_X1_CG</code>	
<p>* The service is activated by default.</p> <p>** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Control Services Reference Manual</p> <p>*** See EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual</p>						

\$SPBool

General Description

The `$SPBool` process control module template is used to enter a setpoint/value from the Supervision runtime.

The template creates a variable of data type `BOOL` whose value you can set by using various types of genies that are referenced by the template.

The output varies depending on the genie that you select.

Parameters

Configuration

The table describes the **Configuration** parameters that you can configure:

Element Name	Child Element Name	Name	Type	Default value	Description
Genies		Bool Type	Enum	CheckBox	<p>Allows you to select the genie that is displayed during runtime to enter the setpoint/value.</p> <p>Values:</p> <ul style="list-style-type: none"> Latch: The output is set (1) when the genie is latched and remains 1, until unlatched. CheckBox: The output is set (1) when the checkbox is selected and remains 1, until unselected. Pulse: When clicked, the output is set (1) for a period that you can configure.

Element Name	Child Element Name	Name	Type	Default value	Description
		Legend	String	Empty	Text that you enter here is displayed in a field to the right of the value during runtime if <i>Hide Legend</i> is false.
		HideLegend	Boolean	0	Select the parameter to hide the legend. If <i>Hide Legend</i> is false and the value of <i>Legend</i> is empty (NULL), the description of the instance is displayed in the legend field.
Genies	Pulse	PulseTime	Duration	00:00:05	Period during which the output remains 1 after the value is set.

Historize

The table describes the **Historize** parameters that you can configure:

Element Name	Name	Type	Default value	Description
Tags	Historize_Tag	Boolean	0	Corresponds to the <i>Historize</i> property of advanced alarms of the Supervision Participant. 1 = Sets the <i>Historize</i> property of the corresponding tag to <i>True</i> . Refer to <i>Advanced Alarm Properties</i> in the help of the Supervision Participant for details.

Composition

The `$SPBool` template references composite and facet templates, which provide the following services:

Control	Core services.
Supervision	These services complement the Control services.

The table describes the services that are available from the `$SPBool` control module and the corresponding facet, which implements the service:

Control services (<code>\$SPBool_UC</code>)	Corresponding facet template	Control service description	Associated Supervision services (<code>\$SPBool_CS</code>) *	Corresponding facet template	Supervision service description
Core services					
Logic	<code>\$SPBool_UL</code>	—	Tags	<code>\$Variable-Tag_CD</code>	Refer to Boolean Setpoint Management (see EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual).
—	—	—	Genies	<code>\$SPBool_Check-box_CG</code>	
				<code>\$SPBool_Latch_CG</code>	
				<code>\$SPBool_Pulse_CG</code>	

The control module template exposes the following interface:

Interface identifier	Type/role	Description
Bool	\$Bool/Def	EDT interface

\$SPReal

General Description

The \$SPReal process control module template is used to enter a setpoint/value from the Supervision runtime.

The template creates a variable of data type REAL whose value you can set by using a genie that is referenced by the template.

An entry of up to 10 digits is allowed for the value. You can use up to 2 decimal places.

Parameters

Configuration

The table describes the **Configuration** parameters that you can configure:

Element Name	Name	Type	Default value	Description
Genies	Legend	String	Empty	Text that you enter here is displayed in a field to the right of the value during runtime if <i>Hide Legend</i> is false.
	HideLegend	Boolean	0	Select the parameter to hide the legend. If <i>Hide Legend</i> is false and the value of <i>Legend</i> is empty (NULL), the description of the instance is displayed in the legend field.
Tags	RawZero	String	0	Allows you to enter the low end of the range for the setpoint.
	RawFull	String	0	Allows you to enter the high end of the range for the setpoint.
	EngUnits	String	Empty	Allows you to define the engineering units (EU) with up to 6 character string length.
	Format	String	Empty	Allows you to define the display format (for example, #####.##.) NOTE: Do not include EU in the display format as it causes duplication of the EU.

Historize

The table describes the **Historize** parameters that you can configure:

Element Name	Name	Type	Default value	Description
Tags	Historize_Tag	Boolean	0	<p>Corresponds to the <i>Historize</i> property of advanced alarms of the Supervision Participant.</p> <p>1 = Sets the <i>Historize</i> property of the corresponding tag to <i>True</i>.</p> <p>Refer to <i>Advanced Alarm Properties</i> in the help of the Supervision Participant for details.</p>

Composition

The `$SPReal` template references composite and facet templates, which provide the following services:

Control	Core services.
Supervision	These services complement the Control services.

The table describes the services that are available from the `$SPReal` control module and the corresponding facet, which implements the service:

Control services (<code>\$SPReal_UC</code>)	Corresponding facet template	Control service description	Associated Supervision services (<code>\$SPReal_1_CS</code>) *	Corresponding facet template	Supervision service description
Core services					
Logic	<code>\$SPReal_UL</code>	–	Tags	<code>\$VariableTag_CD</code>	Refer to REAL Setpoint Management (see EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual).
–	–	–	Genies	<code>\$SPReal_10_CG</code>	

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
Real	<code>\$Real/Def</code>	EDT interface

\$SPInt

General Description

The `$SPInt` process control module template is used to enter a setpoint/value from the Supervision runtime.

The template creates a variable of data type INTEGER whose value you can set by using a genie that is referenced by the template.

NOTE: The corresponding EcoStruxure Process Expert data type that is used for data entry is SHORT (see EcoStruxure™ Process Expert, User Guide).

Parameters

Configuration

The table describes the **Configuration** parameters that you can configure:

Element Name	Name	Type	Default value	Description
Genies	Legend	String	Empty	Text that you enter here is displayed in a field to the right of the value during runtime if <i>Hide Legend</i> is false.
	HideLegend	Boolean	0	Select the parameter to hide the legend. If <i>Hide Legend</i> is false and the value of <i>Legend</i> is empty (NULL), the description of the instance is displayed in the legend field.
Tags	RawZero	String	0	Allows you to enter the low end of the range for the setpoint.
	RawFull	String	0	Allows you to enter the high end of the range for the setpoint.
	EngUnits	String	Empty	Allows you to define the engineering units (EU) with up to 6 character string length.
	Format	String	Empty	Allows you to define the display format (for example, #####.##.) NOTE: Do not include EU in the display format as it causes duplication of the EU.

Historize

The table describes the **Historize** parameters that you can configure:

Element Name	Name	Type	Default value	Description
Tags	Historize_Tag	Boolean	0	Corresponds to the <i>Historize</i> property of advanced alarms of the Supervision Participant. 1 = Sets the <i>Historize</i> property of the corresponding tag to <i>True</i> . Refer to <i>Advanced Alarm Properties</i> in the help of the Supervision Participant for details.

Composition

The \$SPInt template references composite and facet templates, which provide the following services:

Control	Core services.
Supervision	These services complement the Control services.

The table describes the services that are available from the `$SPInt` control module and the corresponding facet, which implements the service:

Control services (<code>\$SPInt_UC</code>)	Corresponding facet template	Control service description	Associated Supervision services (<code>\$SPInt_CS</code>) *	Corresponding facet template	Supervision service description
Core services					
Logic	<code>\$SPInt_UL</code>	–	Tags	<code>\$Variable-Tag_CD</code>	Refer to INT Setpoint Management (EcoStruxure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual).
–	–	–	Genies	<code>\$SPInt_10_CG</code>	

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
Int	<code>\$Int/Def</code>	EDT interface

\$SPDuration

General Description

The `$SPDuration` process control module template is used to enter a setpoint/ value from the Supervision runtime.

The template creates a variable of data type TIME whose value you can set by using a genie that is referenced by the template.

NOTE: The corresponding EcoStruxure Process Expert data type that is used for data entry is Duration (see EcoStruxure™ Process Expert, User Guide).

The equivalent data type of the Supervision Participant is ULONG.

Parameters

Configuration

The table describes the **Configuration** parameters that you can configure:

Element Name	Name	Type	Default value	Description
Genies	Legend	String	Empty	Text that you enter here is displayed in a field to the right of the value during runtime if <i>Hide Legend</i> is false.
	HideLegend	Boolean	0	Select the parameter to hide the legend. If <i>Hide Legend</i> is false and the value of <i>Legend</i> is empty (NULL), the description of the instance is displayed in the legend field.

Element Name	Name	Type	Default value	Description
Tags	RawZero	Duration	00:00:00	Allows you to enter the low end of the range for the setpoint.
	RawFull	Duration	00:00:00	Allows you to enter the high end of the range for the setpoint. NOTE: Verify that the value does not exceed 24.20:31:23.647 to comply with the maximum value supported by the ULONG data type. This is the data type used by the tag managing the value.

Historize

The table describes the **Historize** parameters that you can configure:

Element Name	Name	Type	Default value	Description
SPDuration Tags	Historize_Tag	Boolean	0	Corresponds to the <i>Historize</i> property of advanced alarms of the Supervision Participant. 1 = Sets the <i>Historize</i> property of the corresponding tag to <i>True</i> . Refer to <i>Advanced Alarm Properties</i> in the help of the Supervision Participant for details.

Composition

The `$SPDuration` template references composite and facet templates, which provide the following services:

Control	Core services.
Supervision	These services complement the Control services.

The table describes the services that are available from the `$SPDuration` control module and the corresponding facet, which implements the service:

Control services (<code>\$SPDuration_UC</code>)	Corresponding facet template	Control service description	Associated Supervision services (<code>\$SPDuration_CS</code>) [*]	Corresponding facet template	Supervision service description
Core services					
Logic	<code>\$Time_UL</code>	—	Tags	<code>\$VariableTag_CD</code>	Refer to Duration Setpoint Management (see EcoStructure™ Process Expert - General Purpose Library Classic Process Supervision Services Reference Manual).
—	—	—	Genies	<code>\$SPDuration_10_CG</code>	

The control module template exposes the following interfaces:

Interface identifier	Type/role	Description
Time	<code>\$Time/Def</code>	EDT interface

Index

\$AlarmSummary	
template description	115
\$AnalogCalc	
template description	33
\$AnalogInput	
template description	26
\$AnalogInput1	
template description	28
\$AnalogLinear	
template description	30
\$AnalogOutput	
template description	31
\$AnalogSelect	
template description	117
\$AnalogSelect1	
template description	118
\$ControlValve	
template description	68
\$DigitalCalc	
template description	35
\$DigitalInput	
template description	36
\$DigitalOutput	
template description	38
\$HandValve	
template description	45
\$IBPhase	
template description	107, 112
\$IMC	
template description	82
\$LeadLag	
template description	84
\$MAnalogInput1	
template description	39
\$MessageBox	
template description	119
\$Motor	
control and supervision parameters	47
template description	47
\$Motor2	
template description	53
\$MotorVS	
template description	72
\$MValve	
template description	57
\$MValveWithPos	
template description	76
\$PID	
template description	85
\$PIDLoop	
template description	87
\$PWMController	
template description	88
\$Ramp	
template description	90
\$Range	
template description	41, 93
\$RatioCtl	
template description	91
\$SequenceDFB	
template description	97, 102
\$SPBool	
template description	120
\$SPDuration	
template description	125
\$SPInt	

template description	123
\$SPReal	
template description	122
\$Step3Ctl	
template description	94
\$TOTAL	
template description	41
\$Valve	
template description	61
\$Valve2	
template description	64

A

alarms	
\$AlarmSummary template	115
analog device control templates	
\$ControlValve	68
\$MotorVS	72
\$MValveWithPos	76
analog input signals	
acquisition use cases	23
analog inputs	
\$AnalogCalc template	33
\$AnalogInput template	26
\$AnalogInput1 template	28
\$AnalogLinear template	30
\$MAnalogInput1 template	39
analog outputs	
\$AnalogOutput template	31
analog signals	
\$Range template	41
auxiliary functions templates	
\$AlarmSummary	115
\$AnalogSelect	117
\$AnalogSelect1	118
\$MessageBox	119

B

batch phase control template	
\$IBPhase	107
batch phase manager template	
\$IBPhase Example Use Cases	112

C

controllers	
\$IMC template	82
\$LeadLag template	84
\$SplitRange template	93
\$Step3Ctl template	94

D

digital inputs	
\$DigitalInput template	36
digital outputs	
\$DigitalOutput template	38
digital signals	
\$DigitalCalc template	35

M

mapping analog input signals	
acquisition use cases	23

message boxes	
\$MessageBox template	119
motors	
\$Motor template	47
\$Motor2 template	53
\$MotorVS template	72

O

on/off device control templates	
\$HandValve	45
\$Motor	47
\$Motor2	53
\$MValve	57
\$Valve	61
\$Valve2	64

P

PIDs	
\$PID template	85
\$PIDLoop template	87
process control templates	
\$IMC	82
\$LeadLag	84
\$PID	85
\$PIDLoop	87
\$PWMController	88
\$Ramp	90
\$RatioCtl	91
\$SplitRange	93
\$Step3Ctl	94
process templates	
composition strategy	12
description	12
list of templates	19
naming convention	15
PWM	
\$PWMController template	88

R

ramps	
\$Ramp template	90
ratio	
\$RatioCtl template	91

S

sequential control template	
\$SequenceDFB	97
\$SequenceDFB Example Use Cases	102
setpoint management	
\$SPBool	120
\$SPDuration	125
\$SPInt	123
\$SPReal	122
signal processing templates	
\$AnalogCalc	33
\$AnalogInput	26
\$AnalogInput1	28
\$AnalogLinear	30
\$AnalogOutput	31
\$DigitalCalc	35
\$DigitalInput	36
\$DigitalOutput	38
\$MAnalogInput1	39

\$Range	41
signal selection	
\$AnalogSelect template	117
\$AnalogSelect1 template	118

T

templates	
list of process templates	19
Totalizer	
\$TOTAL	41

V

valves	
\$ControlValve template	68
\$HandValve template	45
\$MValve template	57
\$MValveWithPos template	76
\$Valve template	61
\$Valve2 template	64

Schneider Electric
35 rue Joseph Monier
92500 Rueil Malmaison
France

+ 33 (0) 1 41 29 70 00

www.se.com

As standards, specifications, and design change from time to time,
please ask for confirmation of the information given in this publication.

© 2023 Schneider Electric. All rights reserved.

EIO0000000987.17