EcoStruxure™

Process Expert — Cement Library

Templates Reference Manual

EIO0000004524.00
06/2021

Schneider Electric™

www.se.com

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.

Table of Contents

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.

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 (English version prevails):

- 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 actually 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 functionality and features of the control services, Supervision services, and process templates.

This document only covers the functional aspects of templates, control, and Supervision services when engineering a system, using EcoStruxure™ Process Expert, and describes the dynamic objects visible from the runtime. It does not cover any operational aspects, nor does it provide information on how to use the Supervision services to monitor and operate control systems.

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

Validity Note

This document is valid for EcoStruxure™ Process Expert 2020 or later - General Purpose Library.

Related Documents

Title of Documentation	Reference Number
EcoStruxure™ Process Expert - General Purpose Library User Guide	EIO0000004045

You can download these technical publications and other technical information from our website at www.se.com/ww/en/download/ .

Technical Support

Visit https://app.schneider-electric.com/ecostruxure-hybrid-dcs for support, software updates, and latest information.

Registration required.

Product Related Information

⚠ WARNING

LOSS OF CONTROL

- Perform a Failure Mode and Effects Analysis (FMEA) 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.
- Review the implications of communication link interruptions and take actions to mitigate.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and fault conditions) according to the safety analysis and applicable codes, and regulations.
- Apply local accident prevention and safety regulations and guidelines. [1]
- Test each implementation of this library 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.

Overview

What's in This Part

This part explains general overview of EcoStruxure Process Expert Cement Library and concepts implemented for controlling equipments.

Template

What's in This Chapter

This chapter describes the basic concepts of the cement library template.

List of Templates

The following table lists the templates that are available for Cement Process family.:

Templates	Description
`GasAnalyser`	Gas Analyser
`GirthGearLubCtrl`	Gearth Gear Lubrication Control
`MultiFuelCtrl`	Multi Fuel Control

Acronyms and Definitions

The following table lists the acronyms used in this manual:

Acronym	Definition
FB	Feedback
PV	Present Value
Cmd	Command
ILCK	Interlock
ILCKOP	Interlock Output

Description

The EcoStruxure Process Expert cement 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.

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 logic facets expose the OPC (PLC - HMI variable) 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 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).

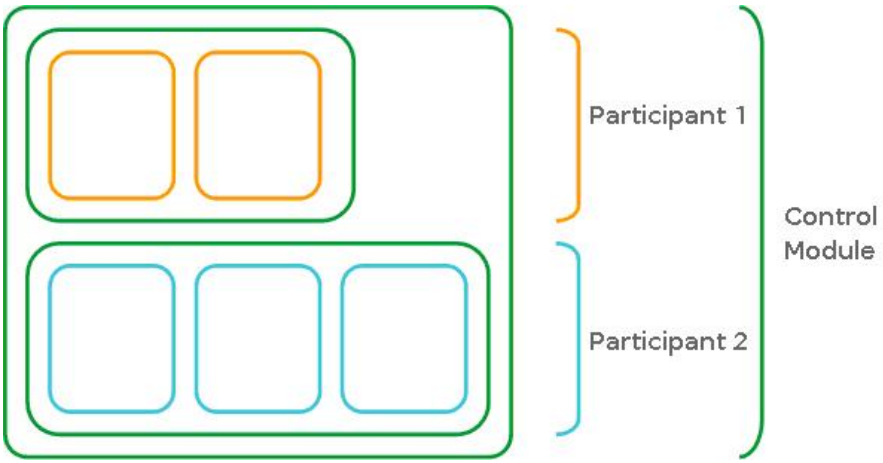
Composition Approach

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

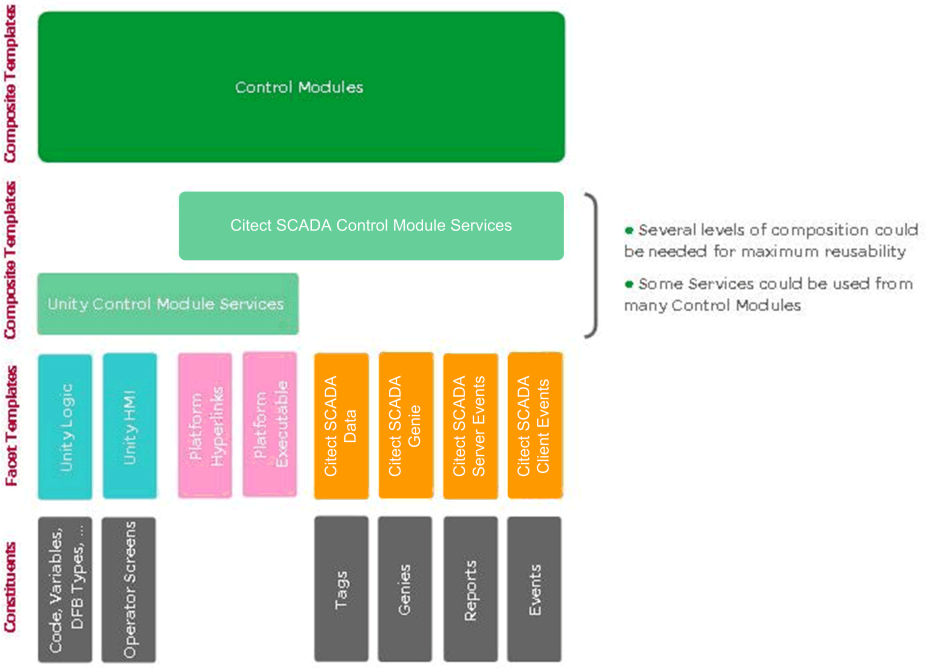
when needed. Avoid the use of design composite templates with too many nested levels to avoid big changes propagation and to facilitate their maintainability. The libraries provide facet and composite templates which could be used to create instances or as part of other composite templates. This modular 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.



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 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.

  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. That is also confirming the compatibility with previously created resources.

NOTE: You can use up to 18 characters while naming.

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 `$GasAnalyser` and enters the name 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).

  NOTE: You can use up to 18 characters while naming.

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.

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, `$GasAnalyser` for exchanging the name of a Control variable of type `$GasAnalyser`).
- 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, **$GasAnalyser** for exchanging the names of the OPC items in the Control 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 logic facet templates.
  - `_CD` for Citect SCADA data facet templates.
  - `_CG` for Citect SCADA genie 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, `$GasAnalyser`).
- Control module services: The contained functionality is provided by the same participant for covering some services required for implementing one or more types of control modules. In such case, a suffix is added for expressing their scope (that is, the suffix `_UC` for the template `$GasAnalyser_UC` is expressing that it encapsulates functionality required from the perspective of the Control for implementing a motor 1 speed 1 direction).

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

  - `_UC` for Control logic composite templates.
  - `_CS` for Citect SCADA data composite templates.
  - `_CG` for Citect SCADA genie composite templates.

Common Parameters

Asset and Display

The table describes the asset and display parameters of supervision:

Section	Parameter description	Type	Default	Additional remarks
Asset	Asset Priority	*Enum*	Normal (4)	Priority of the asset on a scale of 1 to 5 and 9 where 1 is the highest priority.
	Physical location	*String*	–	Asset location and properties have no specific impact on the library operation as delivered but are instead available for user or system data.
	Custom 1 (alarm filter)		–	User defined properties for the asset as well as each of its alarms. The asset filters allow the asset type to be used to enable rapid filtering of alarm and event data.
	Custom 2 (alarm filter)			
	Custom 3 (alarm filter)			
	Custom 4 (user defined)		–	User defined Properties for the asset.
	Custom 5 (user defined)			
	Custom 6 (user defined)			
Display	Process page name	*String*	–	Process graphic page name where equipment genie is available.
	Other page names (optional list)			Overview graphic page associated with the equipment.

Common Control Module Template Interfaces

The table describes the common control module template interfaces:

Interface identifier	Type/Role	Description	Example
`Bool`	`$Bool`	An input interface of boolean type variable.	Different type of variable can be connected to different object.
`INT`	`$Integer`	An input interface of integer type variable.	
`Real`	`$Real`	An input interface of real type variable.	
`Time`	`$Time`	An input interface of time type variable.	

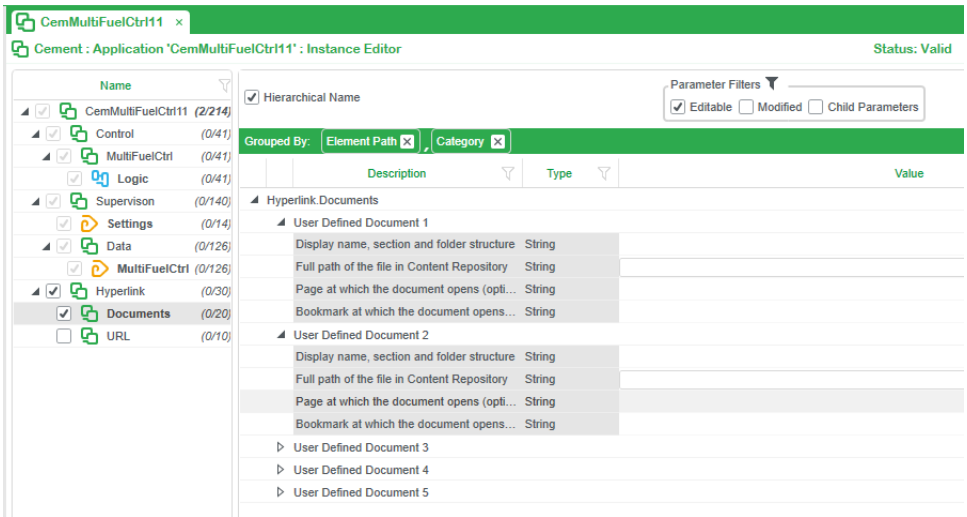
Hyperlink Services

EcoStruxure Process Expert provides the user links to access documents associated with an asset through the operations client (runtime navigation services). Hyperlinks are defined within the library to provide access to elements configured by the library.

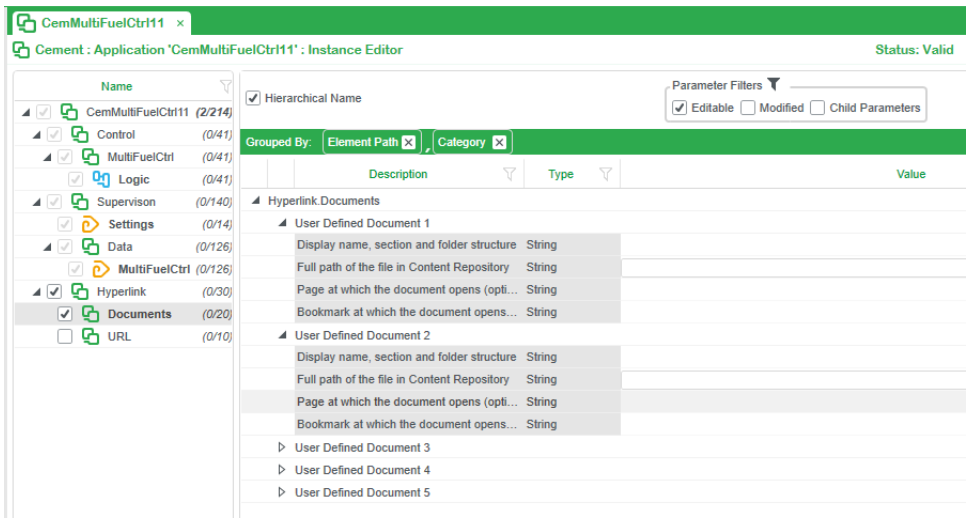
Hyperlinks are also available as a configurable parameters for the user to provide their own links to the documents and web pages such as:

- Equipment Manuals
- Process and Instrumentation Diagrams
- Maintenance Procedures
- Maintenance Requests
- Reports

This figure shows the `HyperLink` configuration to the documentation and web pages:

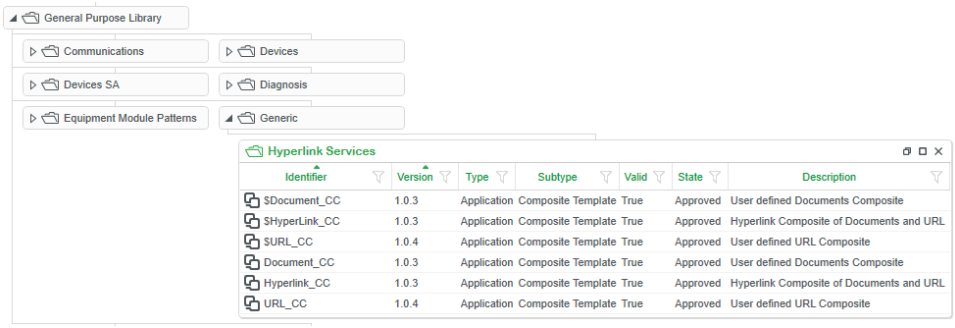


This figure shows the Runtime Navigation to the documentation and web pages:



Hyperlink composites are stored in *Hyperlink* Services folder under *Generic* folder:

This figure shows the Hyperlink services folder:



Control

What's in This Chapter

Overview

The EcoStruxure Process Expert software provides the resources that have been pre-configured and tested by Schneider Electric and that were specifically designed for automating a large variety of processes.

The control resources for process control provide the common required functions, facilitating the development of control systems.

To automate and simplify the implementation process of control systems, you can use these resources together with tools for code generation and for the synchronization of control and monitoring subsystems.

This document describes the basic concepts and details behind each one of the function blocks (DFBs) for implementing the common cross-process and cross-market EcoStruxure control functions.

Modularity

Not Applicable

Function Block Interface

Overview

The function blocks for Process provide an interface that allows them to be configured, monitored, and controlled both from the monitoring subsystem and the control subsystem (continuous and/or sequential control).

The following interfaces are provided:

- Basic Configuration
- Continuous Control
- Sequential Control
- States and Monitoring

Basic Configuration

DFB input pins are usually connected to static data and recognized in engineering time (for example, input channel range).

Continuous Control

DFB input and output pins:

- Allow receiving commands from other blocks.
- Provide block status to other blocks to enable implementing switching operations (for example, remote set-point -`RSP`-), detected alarms (for example the Data from the multiple Dual media filter block is transferred to and from the Queue block to enable queuing of backwash requests and backwashing as per the order in the queue block.

Sequential Control

SC public and structured variable publishes the block status and allows its control from the control sequences (commands not kept).

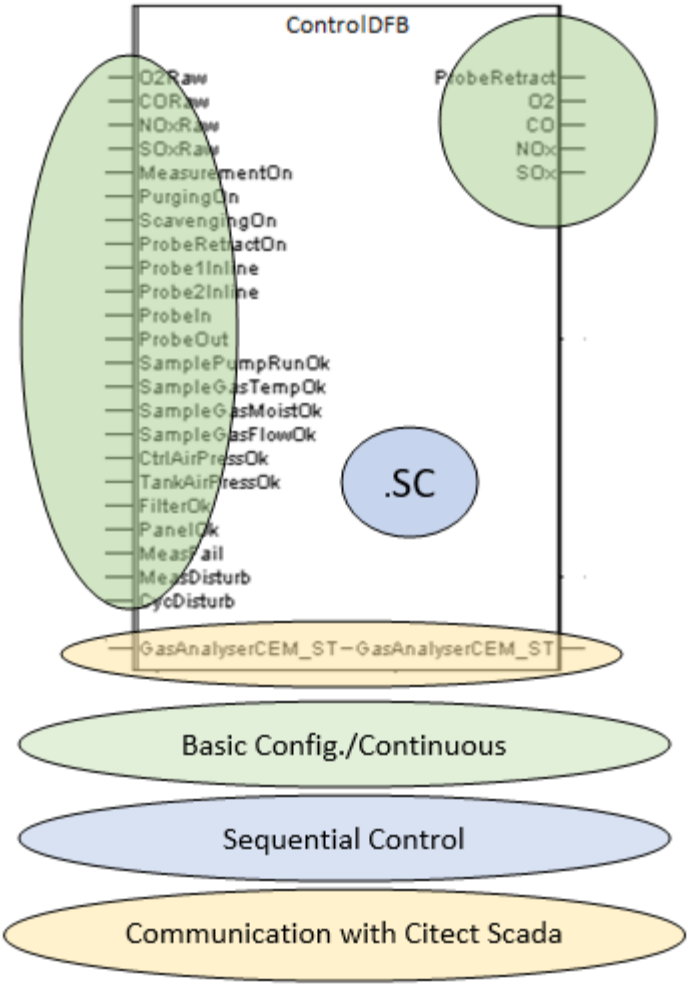
States and Monitoring

Depending on their types, the blocks feature up to 2 input/output pins that need to be connected to variables used to maintain the pin states of the blocks. In addition, these variables allow the commands and parameters received from the monitoring subsystem to be managed.

The frequently considered variables are:

- State (identified with the _ST suffix): These variables hold the state and control characteristics used from the first monitoring subsystem level (dynamic symbols in the flowcharts).
- Configuration (identified with the _CFG suffix): These variables hold second-level information for the parameter configuration of the block. This information is accessible from the monitoring subsystem faceplates.

Illustration



Supervision

What's in This Chapter

This chapter describes the overview of the supervision services.

Supervision Resource Structure

Types of Resources

The GeneralPurposeLibrary project contains the following Supervision resources grouped in libraries:

Symbols	Static graphical elements are supplied as symbol libraries so that you can easily redefine the appearance of Supervision components.
Genies	Dynamic objects are supplied as Supervision genies (in turn, these can be made up of other genies) to represent the dynamic elements that are commonly used in monitoring flowcharts.
Faceplates	Genies are linked to faceplates that allow you to access parameters such as operation, configuration, or maintenance parameters, depending on the function. Faceplates are implemented by using Supervision pages featuring supergenie syntax (refer to the Supervision Participant help for details).
Templates	Template types with various resolutions featuring the standard EcoStruxure Process Expert format.
Cicode	The functions that allow genies and their faceplates to work properly are implemented by using the proprietary Supervision Participant programming language.

Deploying Supervision Resources

When you edit pages, EcoStruxure Process Expert adds the GeneralPurposeLibrary project to the Supervision Participant project so that the resources it contains are deployed with the master project.

Access Control

Privilege Levels

Overview

Supervision components feature access control so that only users having access to the configured area and the required privilege level execute actions on dynamic objects and/or faceplate during operation.

You need to assign privilege levels and areas to users to configure their access permissions depending on the specific requirements of each application.

An Area is configured for each instance. Based on the roles, privilege is assigned to the area. User assigned with the privilege to an area within a role has the view

access to that area. However, the user can only operate the system elements in that area who has appropriate privilege. For more detail, refer to Citect SCADA help.

The privilege levels are mentioned in the custom *8* fields of the item. It is either a single digit or a two digit number. If it is a single digit number, then the privilege is equivalent to the Citect philosophy, refer to Citect SCADA help.

The double-digit privilege level is applicable for write command confirmation.

Default Privilege Levels

By default, the following general criteria are defined to assign privilege levels in Supervision components:

Privilege	Role
0	Viewer
1	Operator
2	Supervisor
3	Maintenance
4	Not used
5	Engineer
6	Not used
8	Not used
{X}	Administrator
NOTE: *X* ranges from 1 to 8.	

Confirmation

Confirmation operation is applicable to write the command. Write operation is confirmed by adding the second digit to the privilege.

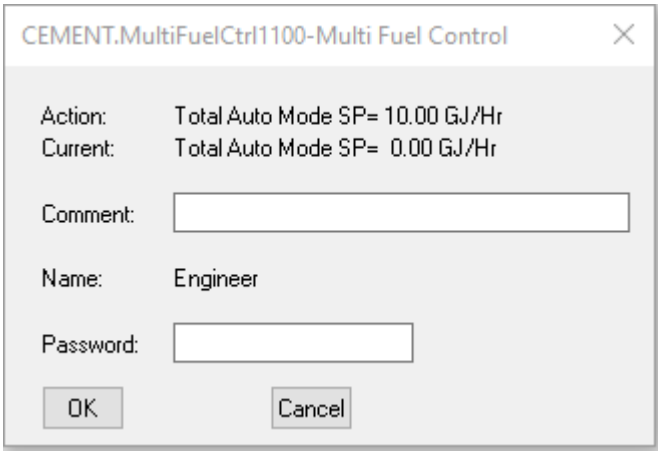
Following are two types of confirmation based on the privilege number:

- Single confirmation
- Double confirmation

Single Confirmation

Confirmation from the currently logged-in user is required. The user has to perform the confirmation operation if the second digit of privilege level is *0*. For example, if the privilege level is set to *50* for the item, then the engineer confirmation *(5 + 0)* are required to perform an operation. This is termed as `Engineer (Confirmed)`, `Operator (Confirmed)`, and so on, based on the privilege level.

Example:

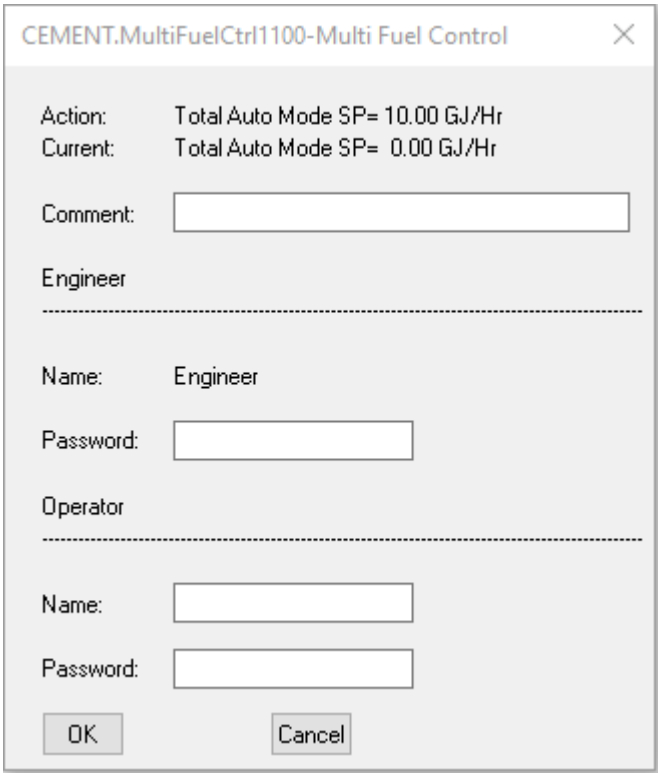


In earlier example, a login user is `Engineer`. when the `Engineer` changes the deviation limit from *50 to 60* of silo weight, a security popup appears and the `Engineer` must have to enter the password again to confirm the operation.

Double Confirmation

Confirmation from currently logged-in user and verifier user who has privilege equal to the second digit is required. The user has to perform the confirmation and verify the operation if the second digit of privilege level is not *0*. If the privilege level is set to *15* for the item, then the `Operator` and `Engineer` *(1 + 5)* are required to perform operation. This is termed as `Operator` + `Engineer`, `Operator` + `Supervisor`, and so on, based on the privilege level.

Example:



In earlier example, a login user is `Operator`. When the `Operator` changes the deviation limit from *50 to 60* of silo weight, a security popup appears and the `Operator` enters the password again to confirm the operation and `Verifier` needs to enter the password to verify the operation.

Calculated Variable

Overview

A calculated variable allows you to generate an item value at runtime that is the result of a Cicode address and call an internal Cicode function as a variable item.

Operating Owner

Following are the operating owner:

- **Operator**: Control module is operated by the supervision operator.
- **Cascade**: Control module is operated by continuous control.
- **Program**: Control module is operated by the sequential control/equipment module/InBatch.

Owner symbol shows the abnormal operating owner on the genie. Owner symbol displays, if the current owner is abnormal.

The table describes the current owner calculation:

Owner Enumeration	Owner Symbol	Owner Value	Program Owner = 1	Cascade Owner = 1	Operator Owner = 1
Unknown	**–**	*0*	No	No	No
Operator	**O**	*1*	–	–	Yes
Program	**P**	*2*	Yes	Yes	–
Cascade	**C**	*3*	–	–	–

Owner Function

The table describes the Cicode function to be used based on the available owner (represented by x) in the control module:

Function	Program	Operator	Cascade	Local Panel	Zero Mode	External
GPL_Owner	X	X	X	–	–	–
NOTE: This function is generated based on the instance configuration level.						

Commonly Used Items

Failure Variable Items

The table describes the variable items that are used by Supervision components:

Item Name	Description	Source
Fail{x}Condition{y}	Condition {y} status of detected failure group{x}.	`Fail{x}ConditionWord.Bit{y-1}`
NOTE: {x} represents the group, range from 1 to 4. {y} represents the condition, range from 1 to 15.		
Fail{x}ConditionResult	Result of the conditions of detected failure group{x}.	`Fail{x}ConditionWord.Bit15`
NOTE: {x} represents the group, range from 1 to 4.		
Fail{x}BypassCondition{y}	Condition {y} bypass status of detected failure group{x}.	`Fail{x}ConditionWord.Bit{y-1}`

Item Name	Description	Source
NOTE: {x} represents the group, range from 1 to 4. {y} represents the condition, range from 1 to 15.		
Fail{x}RearmCondition{y}	Condition {y} rearm required status of detected failure group {x}.	`Fail{x}RearmWord.Bit{y-1}`
NOTE: {x} represents the group, range from 1 to 4. {y} represents the condition, range from 1 to 15.		
Fail{x}Disable	Indicates that detected failure group{x} is disabled.	`Fail{x}BypassWord.Bit15`
NOTE: {x} represents the group range from 1 to 4.		
Fail{x}FirstCondition{y}	Indicates that condition {y} is the first condition which made equipment inoperable.	`Fail{x}FirstWord.Bit{y-1}`
NOTE: {x} represents the group, range from 1 to 4. {y} represents the condition, range from 1 to 15.		
IO items		
Fail{x}ConditionWord	Detected failure condition word	`CONDSUMGP_ST.COND`
Fail{x}BypassWord	Detected failure bypass word	`CONDSUMGP_ST.BYPASS`
Fail{x}RearmWord	Detected failure rearm word	`CONDSUMGP_ST.REARM`
Fail{x}FirstWord	Detected failure first word	`CONDSUMGP_ST.FIRST`
NOTE: {x} represents the group, range from 1 to 4.		

Genies

Overview

This section provides general information about the representation of genies that are part of the GeneralPurposeLibrary project.

Using Genies

Genie Availability

The Cement Library resource contains a number of genies grouped in libraries that correspond to Supervision functions. For example, the `gpl_motor` library contains nine genies for on/off motor management.

These genies allow monitoring and interact with control modules during operation.

Genies become visible in runtime once you assign them to a Supervision page.

Assigning Genies

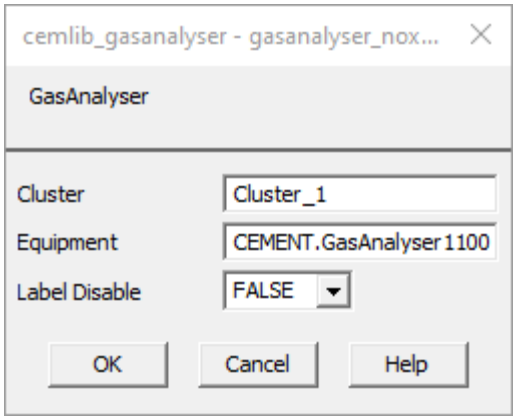
You can assign the genies that are referenced a template to a Supervision page by editing the page, using the **Edit** command.

Genie Properties

Once you have assigned a genie to a page, double-click it to display its properties.

Genies of the Cement Library project have the **Cluster** and **Equipment**, **Label Disable**, properties.

The figure shows an example of the properties dialog box of the `CementLibrary` genie:

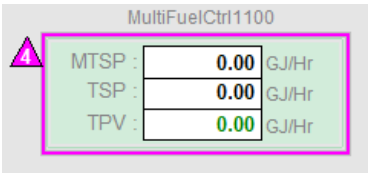
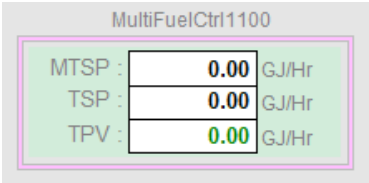
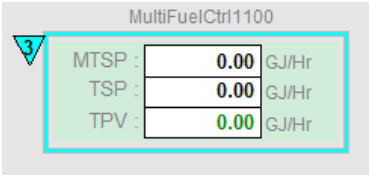


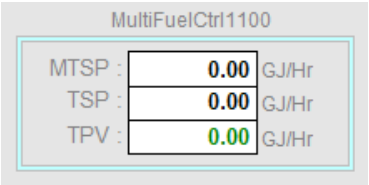
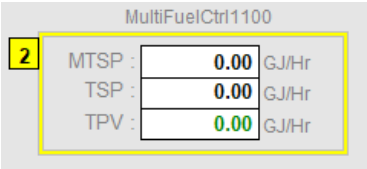
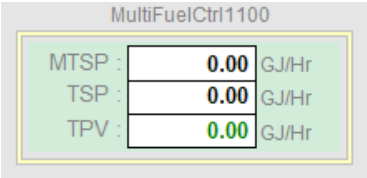
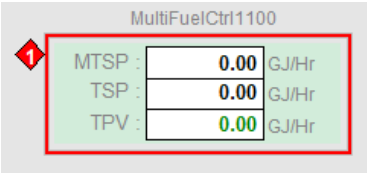
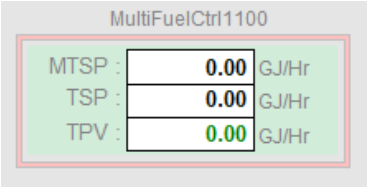
The table describes the properties of the `CementLibrary` genie:

Genie properties	Description
Cluster	Cluster name of the equipment.
Equipment	Equipment name.
Label Disable	By default, Label Disable is False, enter True if you need to hide the label of an object.

Genie Icons

The table describes the icons that are displayed during an operation:

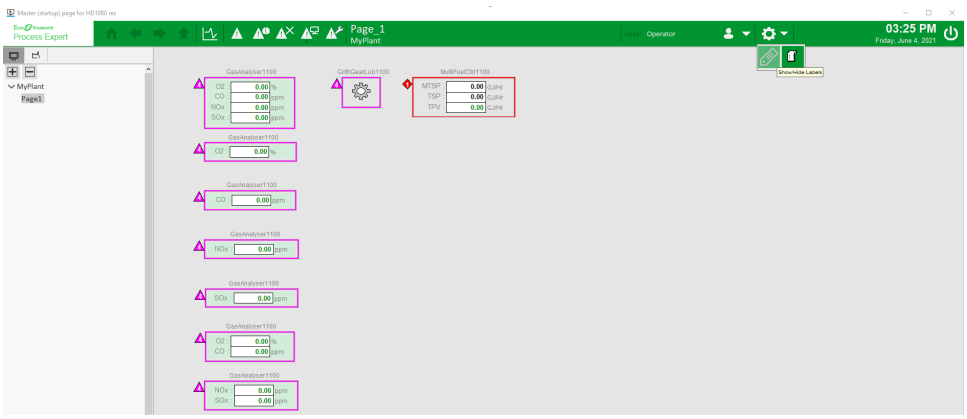
Function	Icon	Meaning	Examples of use	Comment
Displaying owner selection	**P**	Program	Queue in Program Owner mode(The program sets the grant).	–
	C	Cascade		–
	O	Operator	Queue in Operator mode (the operator sets the grant)	The icon is displayed only if its is not in normal mode.
Displaying alarms		Alarm of priority *1*	Differential Pressure very High.	*Critical alarm ON unacknowledge*
		Alarm of priority *1* returns to normal	Filter Run time exceeded set point.	*Critical alarm OFF unacknowledge*
		Alarm of priority *2*	Filter Run time exceeded set point.	*AlarmBorder high unacknowledge*

Function	Icon	Meaning	Examples of use	Comment
		Alarm of priority *2* returns to normal	–	*AlarmBorder high unacknowledge*
		Alarm of priority *3*	Backwash Step 1 feedback not received.	*AlarmBorder medium unacknowledge*
		Alarm of priority *3* returns to normal	–	*AlarmBorder medium unacknowledge*
		Alarm of priority *4*	Short Flush feedback not received.	*AlarmBorder low unacknowledge*
		Alarm of priority *4* returns to normal	–	*AlarmBorder low unacknowledge*
Displaying labels	WI1101	Label of symbols	–	Only if labels are made visible.

Displaying Genies

Overview

The label that identifies each dynamic object inserted in a page can be shown or hidden during runtime.



NOTE: To hide or show the labels, click **Show/Hide Labels** option on the header of the runtime, when the **Label/Disable** is entered as `FALSE` in the property dialog box of the genie.

Faceplate

Overview

This section provides general information about the representation of faceplate linked to genies.

You can access a faceplate by clicking the genie during operation.

Faceplate allow viewing and controlling the corresponding Control block.

Tab Icons

Overview

Faceplate consists of tabs that are grouped according to their functionalities provided by the associated Control block during operation.

Each tab is represented by an icon that you can click to display the information related to it.

Certain tabs are and become available if the control module features the corresponding element and the element is selected.

NOTE: The order of the tabs in a faceplate can be adapted to suit the application without impact on functionality. The tabs order shown in documentation images may differ from that displayed at runtime to the operator.

Description

The table describes the functions that are available in each tab:

Tab	Icon	Functions	Examples of use
Operation		• Current status (present value, setpoint, output value) • Owner change • Operating mode change • Resetting • Configuration of alarms at control level	• Operator/program • Manual/automatic
Input Parameters		Current status of input parameter	Input parameters
Output Parameters/ Queue		Current status of output parameter	Output parameters
State Machine		Current status of the element	`Idle`, `Running`, `Aborted` and so on.
`Failure`		• Status of detected failure conditions. • Bypassing of detected failure conditions. • Navigate to the equipment of the detected failure condition. • Navigate to the initial conditions of pumpset pattern and flow control.	Detected failure condition of the control module.
Initial Conditions		• Status of detected initial conditions • Bypassing of detected initial conditions.	Detected initial condition of the control module.

Tab	Icon	Functions	Examples of use
Alarm		• Acknowledgment of alarms • Configuration of alarms at supervision level. • Navigate to the equipment of the alarm condition.	Acknowledge alarm
Event		Logs the sequence of events	Change of owner

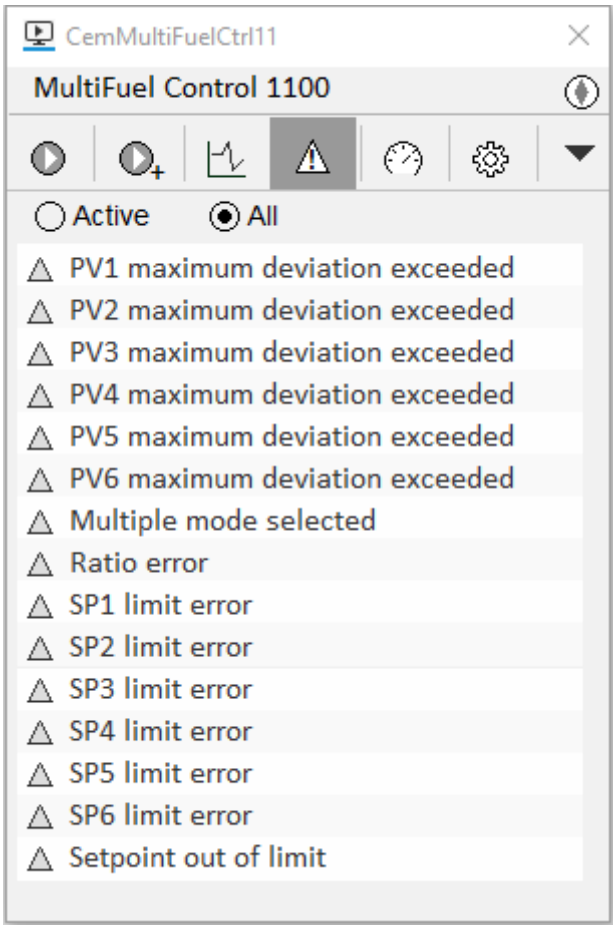
Alarm Tab

Overview

The alarm tab is available in each faceplate. It shows all the configured and active alarms that are associated with the control module and its children.

Alarm Tab Representation

The figure shows an example of an alarm tab:



Tab Icons

Shelved/Disabled Alarm

If any alarm of the control module is in an unacknowledged state, then action required condition is displayed on the alarm tab with flashing orange abnormal indication.

Unacknowledged Alarm

If any alarm of the control module is in an unacknowledged state, then action required condition is displayed on the alarm tab with flashing orange abnormal indication.

Alarm List

Display order

The alarms in the alarm tab are sorted based on the following order:

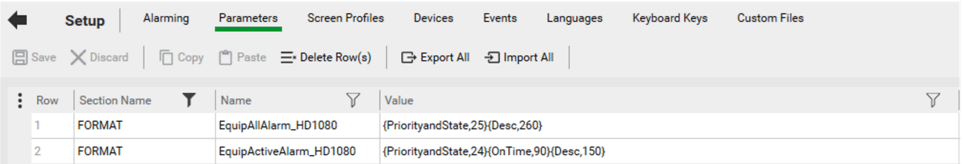
- Alarm Unacknowledged + Active
- Alarm Unacknowledged + Inactive
- Alarm Acknowledged + Active

NOTE:

- The order of displaying an alarm in each state is based on the priority (critical to low).
- The same priority and same state alarms are sorted based on the descending time order.

Display Format

User can manage the display format of an alarm list details by overwriting the **Value** field of `DefaultGPLEquipAllActiveAlarm_HD1080` and `DefaultGPLEquipActiveAlarm_HD1080` available in the **Parameter** tab (Setup > Parameters). Parameter can be overwritten by configuring the same parameter with required value in the Citect project. Refer to the figure.



For example, if user wants to display the category of individual alarm in the third column of an alarm list shown on the faceplate, overwrite the value of the parameter by `{PrioityandState,25}{Desc,260}{category,20}`.

Selection

User can select one of the following alarm list display:

- **Active**: Alarm associated to control module.
- **All**: Configured alarms of the control module.

NOTE: When the *EnableChildSelection* parameter (Setup > Parameters) under section `GPL.Alarm` is configured as `TRUE`, user can select to display the alarms in the child equipment.

Action

Following are the possible actions:

- Acknowledge
- Shelve For
- Shelve Until
- Restore
- Navigate

NOTE: Navigation (sets the context to the selected child) is available for the associated child equipment alarms.

Color Codes

Dynamic Variable Status

Description

The table describes the possible statuses of dynamic variables and the colors that are used to represent them in elements of Supervision components:

Function	Color	Status	Example of use
Discrete module	Grey	Active	Grant Input Set
	White	Passive	Grant Input Reset
	Black	Inconsistency	Open-valve and closed-valve limit switches are active simultaneously
Numeric indicators	400.0	Present value	Current Differential Pressure value
	400.0	Setpoint value	Target temperature value
	100.0	Output value	Control valve position

Optional Services

This part describes the optional services that are available for the cement library.

Failure

This chapter describes the functionality of template, control services, and Supervision functions.

Template

Description

General Description

The `$CONDSUMGP` template is used to evaluate up to 15 signals that can be used, for example, as interlocks or initial conditions. The template allows to evaluate the logic `OR` of these conditions to be evaluated. Each of them can require individual resetting or be ignored according to the template configuration.

You can use this template as a supplement to the rest of the cement library templates to evaluate their interlocking inputs.

Function Description

The main functions of the template are described in the following table:

Function	Description
Summary Condition Evaluation	The template performs the logical `OR` conditions for the given inputs.
Manual Reset Conditions	Conditions configured this way require resetting from the monitoring system.
Bypass	Enables the corresponding condition (one by one) to be ignored when evaluating the expression.

Parameters

The table describes the control and supervision parameters:

Section	Parameter description	Type	Default	Additional remarks
Control Parameter				
Description	Condition{xx}	*String*	–	Description that appears on the `failure` tab of the faceplate when corresponding detected failure condition arises.
	Group description		`@(Failures)`	Group description that appears on the `failure` tab of the faceplate when corresponding detected failure condition arises.
Reset Required	Condition{xx}	*Boolean*	False	Enables reset required condition in `failure` tab.
Disable Bypass	Condition{xx}			Disables bypass condition on `failure` tab.
NOTE: {xx} corresponds to the number of the condition from 1 to 15 that is detected.				
Supervision Parameter				
Historize	Conditions	*Boolean*	False	Enable to historize all activity on `failure` tab.

Available alternate services:

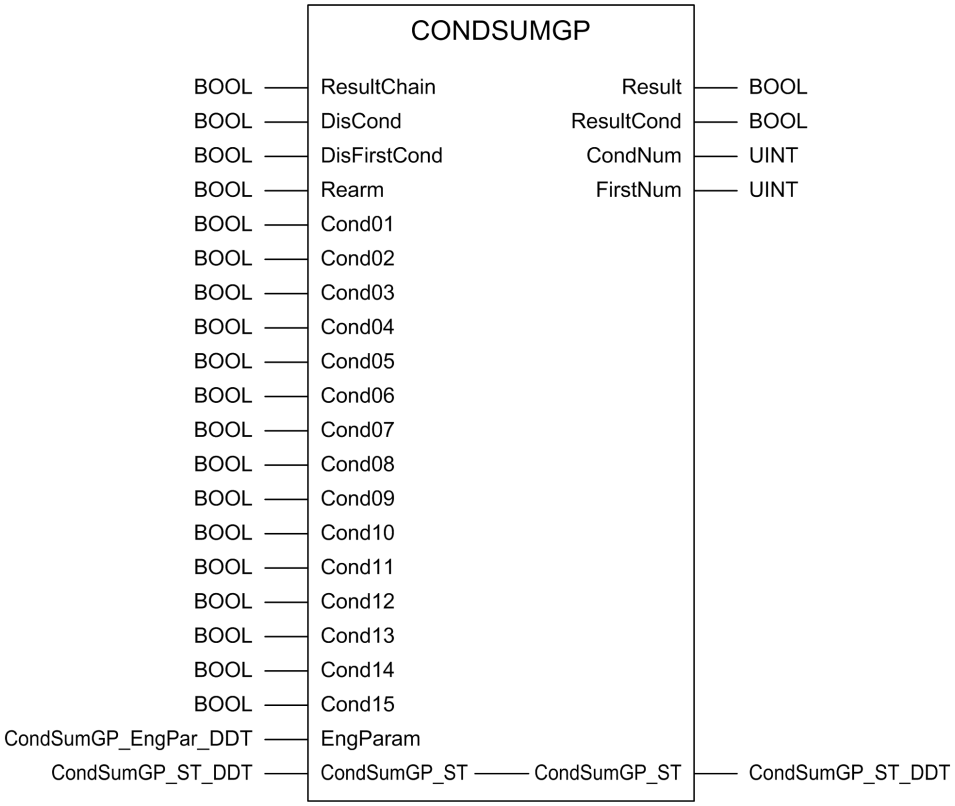
Service	Description
Condsum2On	Detected failure On conditions with 2 groups (15 conditions each).
Condsum1OnAlarm	Detected failure On alarm conditions with 1 group (15 conditions).
Condsum2OnAlarm	Detected failure On alarm conditions with 2 groups (15 conditions each).

These services are available as control and supervision composites (`UC` and `CD`).

Control

DFB Representation

This DFB has been specifically designed for use with the FBD language of the controller.



Inputs

Input Parameter Description

Name	Type	Description
`ResultChain`	`BOOL`	Chain condition status (1 = Condition active).
`DisCond`	`BOOL`	Disable all conditions (1 = Disable).
`DisFirstCond`	`BOOL`	Disable first up condition detected (1 = Disable).
`Rearm`	`BOOL`	Enables the device to reset (1). The DFB sets the signal to 0 after each execution and sets the signal indicating that resetting is required (`CFGW.Rearm`) to 0.
`Cond{x}`	`BOOL`	1 = Condition is activated.
NOTE: `{x}` represents from 01 to 15.		

`CondSumGP_EngPar_DDT` Type

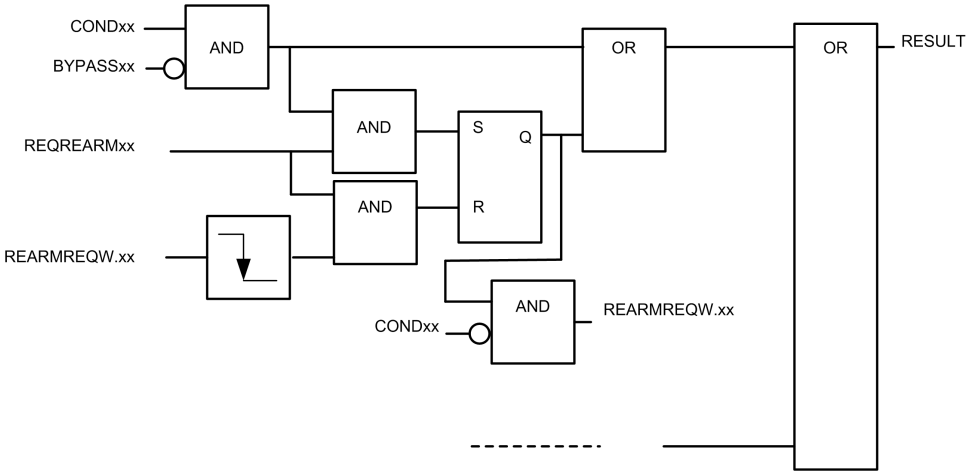
Name	Bit Name	Bit	Type	Description
`RearmReqCW`	–	–	`BYTE`	1 = Enables to configure for manual resetting to be required when the corresponding input condition (`Cond{x}`) is activated.
	`RearmReqC{x}`	`{y}`	`BOOL`	1 = Condition 01 requires rearm once condition returns to normal.
`BypassDisCW`	–	–	`BYTE`	1 = Bypassing Disabled. Makes it possible to disable bypassing for the corresponding interlock conditions (`Cond{x}`).

Name	Bit Name	Bit	Type	Description
	BypassDis{x}	{y}	BOOL	Disable bypass for condition 01 (1 = Bypass disable).
NOTE: {x} represents from 01 to 15 and {y} represents from 0 to 14.				

Outputs

Output Parameter Description

Name	Type	Description
Result	BOOL	1 = Summarizes the result of ORing the conditions according to the following logic diagram.
Result-Cond	BOOL	Result of all the conditions in this DFB, excluding chain condition (1 = Atleast one condition in this block is active or waiting for rearm).
CondNum	UINT	Number of the current active condition, which considered on acending order (0 when no condition are active).
First-Num	UINT	Number of the current active first condition detected (0 when no active first condition detected).



Inputs/Outputs

Input/Output Parameter Description

Name	Type	Description
CondSumGP_ST	CondSumGP_ST_DDT	Data structure that is used as an interface with the monitoring subsystem.

CondSumGP_ST_DDT Type

Name	Bit Name	Bit	Type	Description
CondW	–	–	WORD	Read-only access Bits word with input conditions (refer to the Cond{x} inputs) and their evaluation summary (refer to the Result output pin.
	Cond{x}	{y}	BOOL	1 = Condition 01 is active.
	Result	15	BOOL	Refer to the Result output pin.
FirstCondW	–	–	WORD	First condition
	FirstCond{x}	{y}	BOOL	1 = Condition 01 is the first condition.
BypassW	–	–	WORD	Read/write access

Name	Bit Name	Bit	Type	Description
				Bits word with the bypass signals set from the monitoring system.
	Bypass{x}	{y}	BOOL	1 = Condition 01 bypassed.
	DisCond	15	BOOL	1 = All conditions disabled.
RearmReqW	–	–	WORD	Read/write access Bits word with the required resetting signals (1) set from the DFB and forced resetting (0) from the monitoring system.
	RearmReq{x}	{y}	BOOL	1 = Condition 01 require rearm.
NOTE: {x} represents from 01 to 15 and {y} represents from 0 to 14.				

The following table describes the CondSumGP_ST.CondW word:

Bit	Description
0	Cond01
1	Cond02
...	...
14	Cond15
15	Result

The following table describes the CondSumGP_ST.BypassW word:

Bit	Description
0	Bypass01
1	Bypass02
...	...
14	Bypass15

The following table describes the CondSumGP_ST.RearmReqW word:

Bit	Description
0	RearmReq01
1	RearmReq02
...	...
14	RearmReq15

Public Variables

Public Variable Description

Variable	Type	Description
SC	CondSumGP_SC_DDT	Provides the frequently needed data to monitor the DFB status from the sequential control.

CondSumGP_SC_DDT Type

Name	Type	Description
Result	BOOL	Read-only access Refer to the Result output pin.

Cement Process

What's in This Part

The following chapters describes the Cement process objects.

GasAnalyser - Gas Analyser

What's in This Chapter

This chapter describes the functionality of Template, Control services, and Supervision functions of the $GasAnalyserCEM block.

Template

This section describes the functionality, parameters and interfaces of the $GasAnalyserCEM template.

Description

General Description

The $GasAnalyserCEM control module template supervises and controls the gas analyser operation. The $GasAnalyserCEM control module template supports four inputs (CO, O2, NOx, SOx) and eight types of analyser.

Functional Description

The $GasAnalyserCEM control module template is configured for reading the gas parameters (CO, O2, NOx, SOx) based on the type of analyser configured through the input/output pin (ST.AnalyserType) AnalyserType.

This $GasAnalyserCEM control module template reads the following status of the gas analyser operation:

- Measurement operation on (MeasOn)
- Purging operation on (PurgingOn)
- Probe retraction (ProbeRetractOn)
- Scavenging cycle operation on (ScavengingOn)
- Probe inline status (Probe1Inline, Probe2Inline)
- Probe in status (ProbeIn)
- Probe out status (ProbeOut)

The analysed values are read from the analyser through input pins, and are held when the gas analyser is in Purging mode. This processed value is available at the output pin and displayed in the faceplate.

Parameters

The $GasAnalyserCEM template provides different control and supervision parameters to the user to control the functions as per the requirement.

Control

The table describes the control parameters:

Section	Parameter description	Type	Default	Additional remarks
Configuration	Analyser Type	Enum	O2, CO, NOx & SOx (8)	Allows to change the gas analyser type.
	Time delay for retraction *error* (in sec)	Unsigned-Short	2	Allows to change the time delay for retraction error detected.
	Negate Signal	Bool	False	True = Negates the input signal.
	Custom Signal Name	String	—	Name used for the generated EFB/DFB and variables.

NOTE: The following points are applicable for Negate signal and Custom Signal Name.

- MeasurementOn is selected by default.
- Similar $DISignal_UL is required for
  - PurgingOn
  - ScavengingOn
  - ProbeRetractOn
  - Probe1InLine
  - Probe2InLine
  - ProbeIn
  - ProbeOut
  - SamplePumpRunOk
  - SampleGasTempOk
  - SampleGasMoistOk
  - SampleGasFlowOk
  - CtrlAirPressOk
  - TankAirPressOk
  - FilterOk
  - PanelOk
  - MeasFail
  - MeasDisturb
  - CycDisturb

Supervision

Settings

The table describes the settings parameters:

Section	Parameter description	Type	Default	Additional remarks
Operation	Set Reset	Enum	Operator (confirmed)	To reset alarm
Security	Set Parameters	Enum	Engineer	To change the Probe Retract Time.
	Alarm Acknowledge	Enum	Operator	Privilege to set alarm acknowledge. This parameter is used in alarm tab.

Asset, page 18

Section	Parameter description	Type	Default	Additional remarks
Display, page 18				
Trend	Sample Period	Duration	00:00:05	Defines the rate at which the trend data is displayed on the trend page. Example: Log data after 5 seconds if sample period 00:00:05.
	History Length (Weeks)	Integer	5	The duration of historical data. Example: Historized data of 5 weeks in one file with a configured sampling period.
	History Rollover Time	Duration	12:00:00	The time taken to generate the historized file based on the rollover day configuration.
	History Rollover Day	String	Tuesday	The day at which the new file generates for historising.
	Storage Type	String	TRN_EVENT	A trend that is sampled continuously at a specified period.
	Data Location (Optional)	String	[Data]:	User configurable file location.

Data

The table describes the data parameters:

Section	Parameter description	Type	Default	Additional remarks
Configuration	O2 – Engineering Zero	Float	0.0	Allows to change Engineering zero .
	O2 – Engineering Full	Float	100.0	Allows to change Engineering Full.
	O2 – Engineering Unit	String	%	Allows to change Engineering Unit.
	O2 – Format	String	####.##EU	Allows to change Format
	CO – Engineering Zero	Float	0.0	—
	CO – Engineering Full	Float	100.0	—
	CO – Engineering Unit	String	ppm	—
	CO – Format	String	####.##EU	—
	NOx – Engineering Zero	Float	0.0	—
	NOx – Engineering Full	Float	100.0	—
	NOx – Engineering Unit	String	ppm	—
	NOx – Format	String	####.#EU	—
	SOx – Engineering Zero	Float	0.0	—
	SOx – Engineering Full	Float	100.0	—
	SOx – Engineering Unit	String	ppm	—

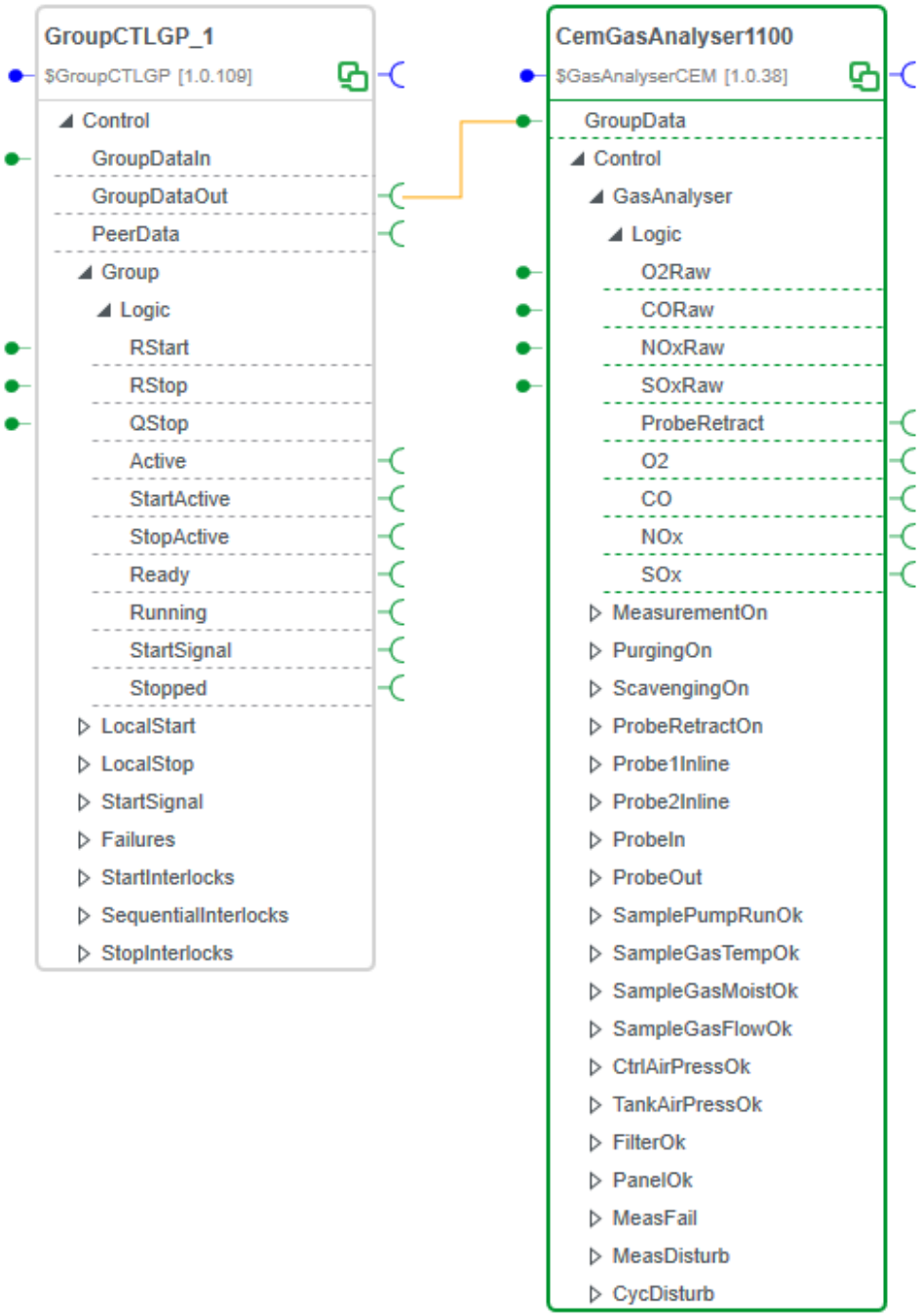
Section	Parameter description	Type	Default	Additional remarks
	SOx – Format	String	####.##EU	—
	Time Format	String	#####	—
Alarm	Probe Retraction Command *Failed*Description	String	@(Probe Retraction command *failed*)	—
	Probe Retraction Command *Failed*Severity	Enum	Low	—
	Probe Retraction Command *Failed* Group	String	*Failure*	—
	Sample Pump Run *failed* Description	String	@(Sample Pump Run *failed*)	—
	Sample Pump Run *Failed* Severity	Enum	Low	—
	Sample Pump Run *Failed* Group	String	*Failure*	—
	Sample Gas Temperature *Failed* Description	String	@(Sample Gas Temperature *failed*)	—
	Sample Gas Temperature *Failed* Description Severity	Enum	Low	—
	Sample Gas Temperature *Failed* Group	String	*Failure*	—
	Sample Gas Moisture *Failed*Description	String	@(Sample Gas Moisture *failed*)	—
	Sample Gas Moisture *Failed* Severity	Enum	Low	—
	Sample Gas Moisture *Failed* Group	String	*Failure*	—
	Sample Gas Flow *Failed* Description	String	@(Sample Gas Flow *failed*)	—
	Sample Gas Flow *Failed* Severity	Enum	Low	—
	Sample Gas Flow *Failed* Group	String	*Failure*	—
	Control Air Pressure *Failed* Description	String	@(Control Air Pressure *Failed*)	—
	Control Air Pressure *Failed* Severity	Enum	Low	—
	Control Air Pressure *Failed* Group	String	*Failure*	—
	Tank Air Pressure *Failed* Description	String	@(Tank Air Pressure *Failed*)	—
	Tank Air Pressure *Failed* Severity	Enum	Low	—

Section	Parameter description	Type	Default	Additional remarks
	Tank Air Pressure *Failed* Group	String	*Failure*	—
	Filter *Failed* Description	String	@(Filter *Failed*)	—
	Filter *Failed* Severity	Enum	Low	—
	Filter *Failed* Group	String	*Failure*	—
	Panel *Failed* Description	String	@(Panel *Failed*)	—
	Panel *Failed* Severity	Enum	Low	—
	Panel *Failed* Group	String	*Failure*	—
	Measurement *Failed* Description	String	@ (Measure-ment *Failed*)	—
	Measurement *Failed* Severity	Enum	Low	—
	Measurement *Failed* Group	String	*Failure*	—
	Measurement Disturbance Description	String	@ (Measure-ment Disturbance)	—
	Measurement Disturbance Severity	Enum	Low	—
	Measurement Disturbance Group	String	*Failure*	—
	Cycle Disturbance Description	String	@(Cycle Disturbance)	—
	Cycle Disturbance Severity	Enum	Low	—
	Cycle Disturbance Group	String	*Failure*	—
Deadband	Oxygen	Un-sign-ed-Short	0	—
	Carbon Monoxide	Un-sign-ed-Short	0	—
	Nitrous Oxide	Un-sign-ed-Short	0	—
	Sulphur Oxide	Un-sign-ed-Short	0	—
Events	Enable Log	Boo-lean	False	—
Trends	Disable O2	Boo-lean	False	—
	Disable CO	Boo-lean	False	—
	Disable NOx	Boo-lean	False	—

Section	Parameter description	Type	Default	Additional remarks
	Disable SOx	Boolean	False	—
Historize	Oxygen	Boolean	False	—
	Carbon Monoxide	Boolean	False	—
	Nitrous Oxide	Boolean	False	—
	Sulphur Oxide	Boolean	False	—
	Analyser Type	Boolean	False	—
	Retraction Time delay	Boolean	False	—

Interfaces

This figure shows the $GasAnalyserCEM template as it appears in Links Editor and an example of other templates connected to it:

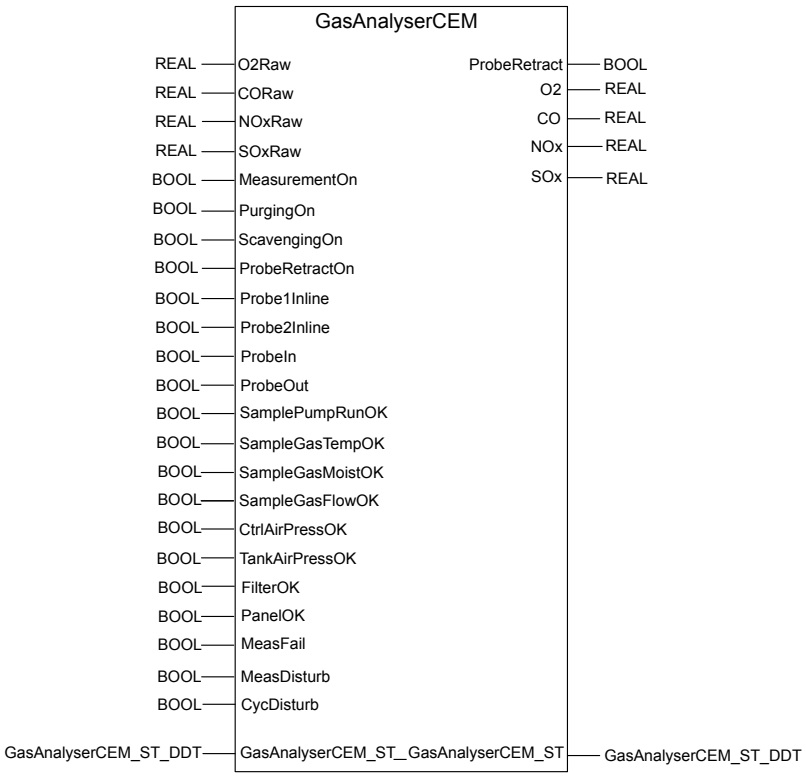


Control

This section describes the `$GasAnalyserCEM` DFB.

DFB Representation

This DFB has been specifically designed for use with the FBD language of the controller.



Inputs

Input Parameter Description

Name	Type	Description
O2Raw	REAL	Raw value of O2 from analyser.
CORaw	REAL	Raw value of CO from analyser.
NOxRaw	REAL	Raw value of NOx from analyser.
SOxRaw	REAL	Raw value of SOx from analyser.
MeasurementOn	BOOL	1 = Analyser is in Measurement mode.
PurgingOn	BOOL	1 = Analyser is in Purging mode.
ScavengingOn	BOOL	1 = Analyser is in Scavenging mode.
ProbeRetractOn	BOOL	1 = Probe is retracted.
Probe1Inline	BOOL	1 = Probe 1 is inline.
Probe2Inline	BOOL	1 = Probe 2 is inline.
ProbeIn	BOOL	1 = Probe in status is active.
ProbeOut	BOOL	1 = Probe out status is active.
SamplePumpRunOk	BOOL	1 = Sample pump run condition is active.
SampleGasTempOk	BOOL	1 = Sample gas temperature is within normal operating range.
SampleGasMoistOk	BOOL	1 = Sample gas moisture is within normal operating range.
SampleGasFlowOk	BOOL	1 = Sample gas flow is within normal operating range.

Name	Type	Description
CtrlAirPressOk	BOOL	1 = Control air pressure is within normal operating range.
TankAirPressOk	BOOL	1 = Tank air pressure is within normal operating range.
FilterOk	BOOL	1 = Filter status is active.
PanelOk	BOOL	1 = Panel status is active.
MeasFail	BOOL	1 = Measurement is operable.
MeasDisturb	BOOL	1 = Measurement disturbance condition is operable.
CycDisturb	BOOL	1 = Cycle disturbance condition is operable.

Outputs

Output Parameter Description

Name	Type	Description
ProbeRetract	BOOL	Probe retraction command
O2	REAL	Processed value of O2
CO	REAL	Processed value of CO
NOx	REAL	Processed value of NOx
SOx	REAL	Processed value of SOx

Inputs/Outputs

Input/Output Parameter Description

Name	Type	Description
GasAnalyserCEM_ST	GasAnalyserCEM_ST_DDT	Structure for gas analyser

GasAnalyserCEM_ST_DDT

The following table describes the structure of the derived data type GasAnalyserCEM_ST_DDT for the $GasAnalyserCEM DFB:

Name	Bit Name	Bit	Type	Description
O2	—	—	REAL	Processed value of O2
CO	—	—	REAL	Processed value of CO
NOx	—	—	REAL	Processed value of NOx
SOx	—	—	REAL	Processed value of SOx
Analyser-Type	—	—	INT	Analyser type
TRetract	—	—	UINT	Time delay for retraction error detected.
CFGW	—	—	WORD	Read/Write access. Configuration word
	ProbeRetract	0	BOOL	Probe retraction command
	AlmRst	7	BOOL	Alarm reset command

Name	Bit Name	Bit	Type	Description
STW	—	—	WORD	Read access. Status word 1
	CtrlAirPressOk	0	BOOL	1 = Control air pressure is within normal operating range and an alarm is off.
	TankAirPressOk	1	BOOL	1 = Tank air pressure is within normal operating range and an alarm is off.
	FilterOk	2	BOOL	1 = Filter status is active and an alarm is off.
	PanelOk	3	BOOL	1 = Panel status is active and an alarm is off.
	MeasFailOk	4	BOOL	1 = Measurement is operable and an alarm is off.
	MeasDisturb	5	BOOL	1 = Measurement disturbance condition is operable and an alert is not generated.
	CycDisturb	6	BOOL	1 = Cycle disturbance condition is operable and an alert is not generated.
	RequiredRearm	7	BOOL	Rearm required.
STAT	—	—	WORD	Read access. Status word
	Wake	0	BOOL	1 = Analyser is in Wake-up state.
	MeasurementOn	1	BOOL	1 = Analyser is in Measurement mode.
	PurgingOn	2	BOOL	1 = Analyser is in Purging mode.
	ScavengingOn	3	BOOL	1 = Analyser is in Scavenging mode.
	ProbeRetractOn	4	BOOL	1 = Probe is retracted.
	Probe1Inline	5	BOOL	1 = Probe 1 is inline.
	Probe2Inline	6	BOOL	1 = Probe 2 is inline.
	ProbeIn	7	BOOL	1 = Probe in status is active.
	ProbeOut	8	BOOL	1 = Probe out status is active.
	SamplePumpRunOk	9	BOOL	1 = Sample pump run condition is active and an alarm is off.
	SampleGasTempOk	10	BOOL	1 = Sample gas temperature is within normal operating range and an alarm is off.
	SampleGasMoistOk	11	BOOL	1 = Sample gas moisture is within normal operating range and an alarm is off.
	SampleGasFlowOk	12	BOOL	1 = Sample gas flow is within normal operating range and an alarm is off.
	Alarminhibit	13	BOOL	1 = All alarms are inhibited. No alarms are generated by the block.
	WarningConditions	14	BOOL	1 = Alert is off.
	AlarmConditions	15	BOOL	1 = Alarm is off.

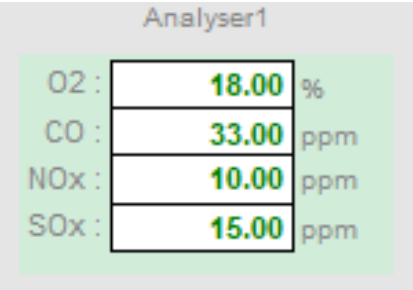
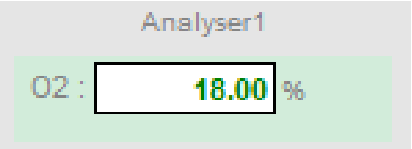
Name	Bit Name	Bit	Type	Description
ALW1	—	—	WORD	Read access. Alarm word
	ProbeRe-tractFail	0	BOOL	Probes retract inoperable.
	SamplePum-pRunFail	1	BOOL	Sample pump runs inoperable.
	SampleGas-TempFail	2	BOOL	Sample gas temperature inoperable.
	SampleGas-MoistFail	3	BOOL	Sample gas moisture inoperable.
	SampleGas-FlowFail	4	BOOL	Sample gas flow inoperable.
	CtrlAir-PressFail	5	BOOL	Control air pressure inoperable.
	TankAir-PressFail	6	BOOL	Tank air pressure inoperable.
	FilterFail	7	BOOL	Filter inoperable.
	PanelFail	8	BOOL	Panel inoperable.
	MeasFail	9	BOOL	Measurement inoperable.
	MeasDisturb-Fail	10	BOOL	Measurement disturbance
	CycDisturb-Fail	11	BOOL	Cycle disturbance

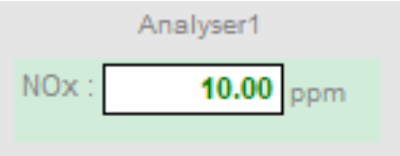
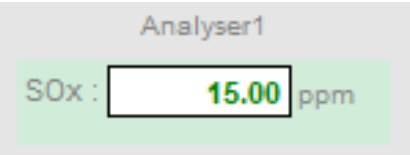
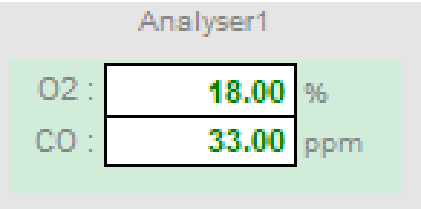
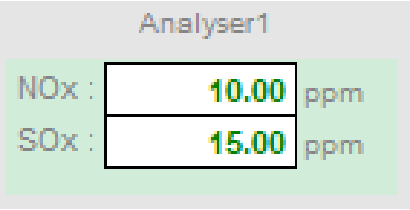
Supervision

This section describes the supervision resources and runtime services that are available for the $GasAnalyserCEM block.

Genies

Representation

Graphic symbol	Genie name
	gasanalyser_all_m
	gasanalyser_o2_m
	gasanalyser_co_m

Graphic symbol	Genie name
	gasanalyser_nox_m
	gasanalyser_sox_m
	gasanalyser_o2_co_m
	gasanalyser_nox_sox_m
	gasanalyser_o2_co_nox_m

Faceplates

Representation of Supervision Data

At the beginning of this document, you can find a general description of the graphic elements that are used in the faceplate.

Available Tabs

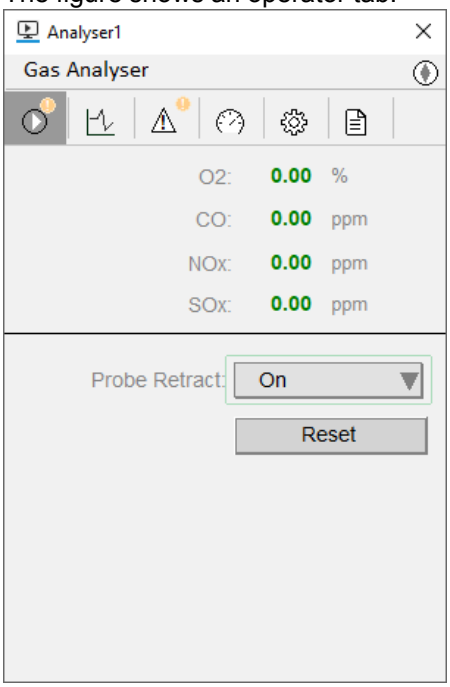
During operation, clicking a gas analyser genie opens a faceplate with the following tabs:

Tabs for core functions:

- Operator
- Trend
- Alarm, page 31
- Measures
- Engineering
- Event

Operator Tab

The figure shows an operator tab:

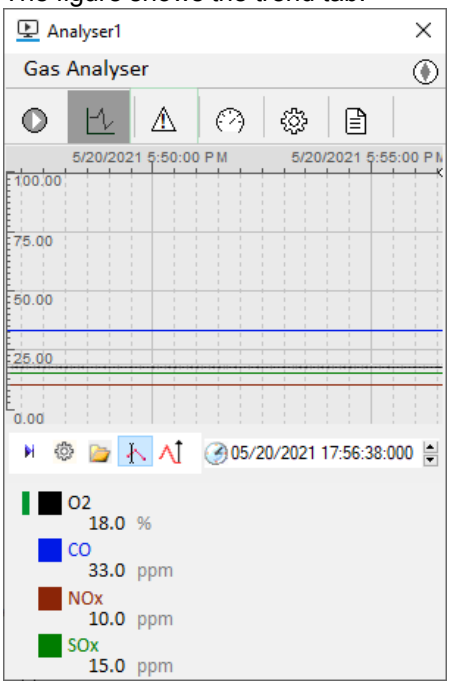


The table describes the control function of an operator tab:

Label	Item Name	Description	Security parameter, page 40
Probe Retract	ProbeRetract	It allows to give command for Emergency Probe Retract.	Operator

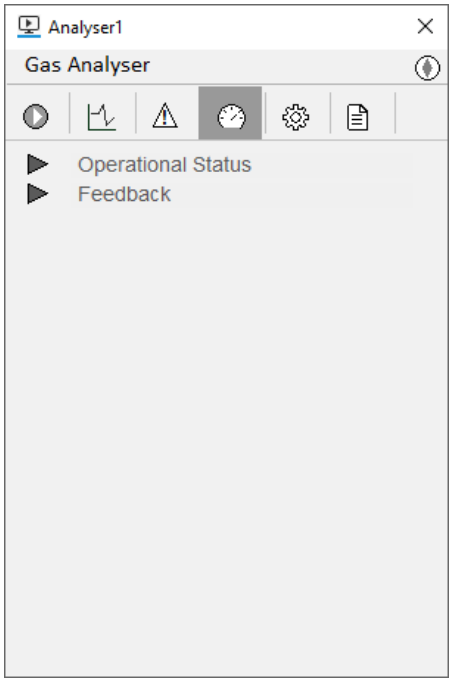
Trend Tab

The figure shows the trend tab:

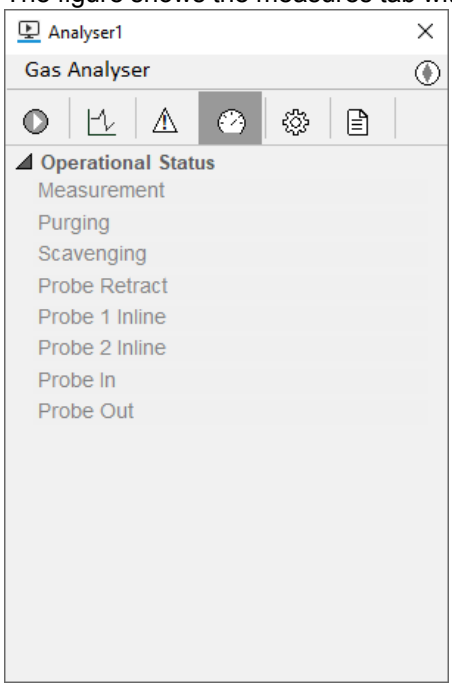


Measures Tab

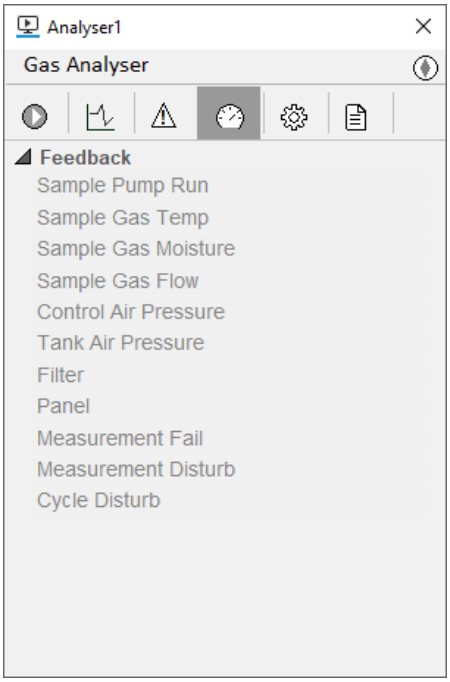
The figure shows the measures tab:



The figure shows the measures tab with operational status:

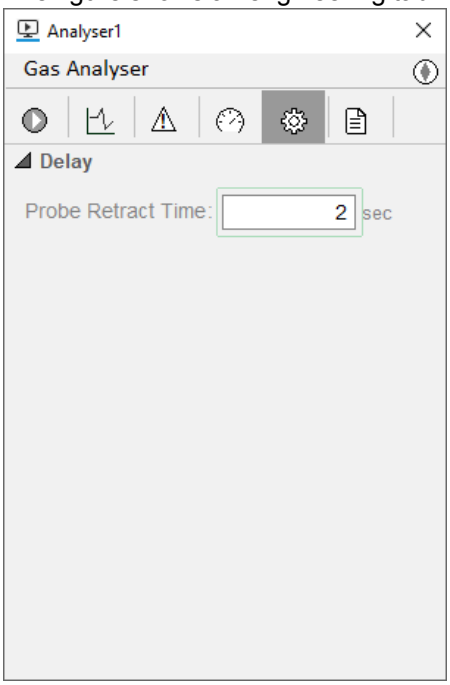


The figure shows the measures tab with feedback:



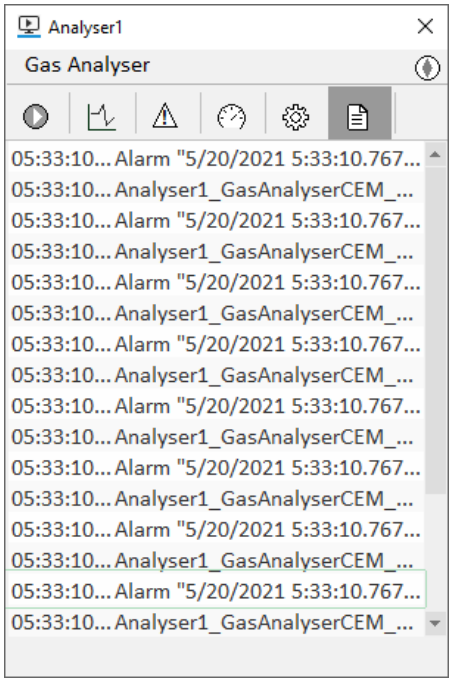
Engineering Tab

The figure shows an engineering tab:



Event Tab

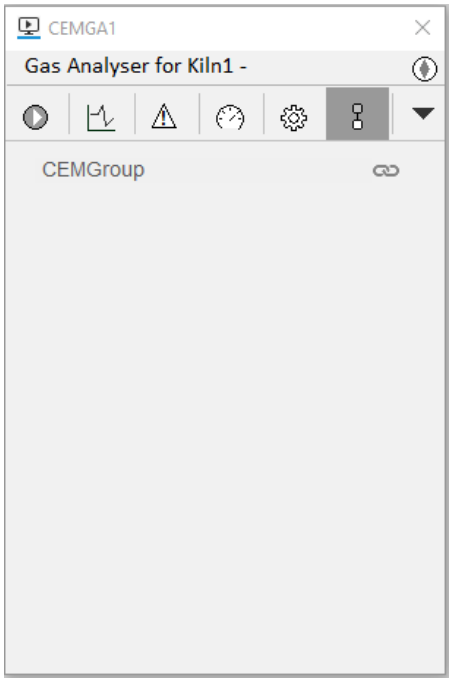
The figure shows an event tab:



```
05:33:10... Alarm "5/20/2021 5:33:10.767...
05:33:10... Analyser1_GasAnalyserCEM_...
05:33:10... Alarm "5/20/2021 5:33:10.767...
05:33:10... Analyser1_GasAnalyserCEM_...
05:33:10... Alarm "5/20/2021 5:33:10.767...
05:33:10... Analyser1_GasAnalyserCEM_...
05:33:10... Alarm "5/20/2021 5:33:10.767...
05:33:10... Analyser1_GasAnalyserCEM_...
05:33:10... Alarm "5/20/2021 5:33:10.767...
05:33:10... Analyser1_GasAnalyserCEM_...
05:33:10... Alarm "5/20/2021 5:33:10.767...
05:33:10... Analyser1_GasAnalyserCEM_...
05:33:10... Alarm "5/20/2021 5:33:10.767...
05:33:10... Analyser1_GasAnalyserCEM_...
05:33:10... Alarm "5/20/2021 5:33:10.767...
05:33:10... Analyser1_GasAnalyserCEM_...
```

Generic Group Tab

The figure shows a generic group tab:



NOTE: This faceplate appears and group data interface is connected.

Items

Overview

This section describes the variables, abnormal conditions, and events items of gas analyser.

Variables

The table describes the variable items that are used by supervision components:

Item name	Description	Enumeration	Address
Bit items derived from Status Word.			
ControlAirPres-sure	Control air pressure feedback	–	Bit0
TankAirPressure	Tank air pressure feedback	–	Bit1
Filter	Filter feedback	–	Bit2
Panel	Panel feedback	–	Bit3
MeasurementFail	Measurement fail detected feedback	–	Bit4
MeasurementDis-turb	Measurement disturb feedback	–	Bit5
CycleDisturb	Cycle disturb feedback	–	Bit6
Bit items derived from Alarm Word			
ProbRetractFail	Probes retract inoperable	–	Bit0
SamplePumpRun-Fail	Sample pump runs inoperable	–	Bit1
SampleGasTemp-Fail	Sample gas temperature inoperable	–	Bit2
SampleGasMoist-Fail	Sample gas moisture inoperable	–	Bit3
SampleGasFlow-Fail	Sample gas flow inoperable	–	Bit4
CtrlAirPressFail	Control air pressure inoperable	–	Bit5
TankAirPresFail	Tank air pressure inoperable	–	Bit6
FilterFail	Filter inoperable	–	Bit7
PanelFail	Panel inoperable	–	Bit8
MeasFail	Measurement inoperable	–	Bit9
MeasDisturb	Measurement disturbance	–	Bit10
CycDisturb	Cycle disturbance	–	Bit11
ControlAirPres-sure	Run feedback	–	Bit0
Bit items derived from STAT Word			
Measurement	Measurement Operational Status	–	Bit1
Purging	Purging Operational Status	–	Bit2
Scavenging	Scavenging Operational Status	–	Bit3
ProbeRetract	Probe Retract Operational Status	–	Bit4
Probe1Inline	Probe 1 Inline Operational Status	–	Bit5
Probe2Inline	Probe 2 Inline Operational Status	–	Bit6
ProbeIn	Probe In Operational Status	–	Bit7
ProbeOut	Probe Out Operational Status	–	Bit8

Item name	Description	Enumeration	Address
SamplePumpRun	Sample Pump Run feedback	–	Bit9
SampleGasTemp	Sample Gas Temp feedback	–	Bit10
SampleGasMoist	Sample Gas Moist feedback	–	Bit11
SampleGasFlow	Sample Gas Flow feedback	–	Bit12
Bit items derived from CFGW			
CommandSetpoint	Feedback.	–	Bit0
Configuration-Word	Feedback.	–	–
Rearm	Feedback.	–	Bit7
RequireRearm	Feedback.	–	Bit7
Bit items derived from Output			
O2	Present value	–	–
CO	Present value	–	–
SOx	Present value	–	–
NOx	Present value	–	–
TRetract	Probe Retract Time	–	–
AnalyserType	Analyser Type	–	–

GirthGearLubCtrl - Girth Gear Lubrication Control

What's in This Chapter

Template

This section describes the functionality, parameters and interfaces of the $GirthGearLubCtrlCEM template.

Description

General Description

The $GirthGearLubCEM control module template supervises and controls the Girth Gear Lubrication operation. It depends on the configuration of the associated devices and the system requirements.

Functional Description

The $GirthGearLubCEM control module template is customized for supervising the girth gear lubrication operation. This block supports 8 spray nozzles and monitors the inoperable conditions.

Auto Mode Operation

The GirthGearLubCtrl block takes the open and closed feedbacks of valves (Valve1 and Valve2), grease spray and grease fill feedback of nozzles (nozzle 1A, 2A, 3A, 4A, 1B, 2B, 3B, and 4B). Valve 1 serves nozzles 1A, 1B, 2A, 2B and Valve 2 serves Nozzles 3A, 3B, 4A, and 4B. The block checks for the synchronization of nozzle spray feedback with valve open feedback and nozzle fill feedback with valve close feedback, and generates the status as operable. If all the nozzles are operable, then it generates the status of the cycle to be ok.

If the synchronization of nozzle feedback with valve position is lost for more than a pre-defined time (Stx.TFbSpray, Stx.TFbFill), an alarm is generated. The block also supports bypass of the nozzles in case of temporary site requirements. Alarms are not generated for the bypassed nozzles.

The block also checks for the inoperable conditions (tank level low, control air pressure low, grease flow low, line differential pressure low) of the girth gear lubrication system and generates relevant alarms. The block also supports bypass of the inoperable conditions in case of temporary site requirements. Alarms are not generated for the relevant inoperable conditions.

The alerts and alarms are indicated in the SCADA. The details of the active alarms are shown in the **Alarm** tab of the faceplate. The trip alarms can be reset by AlmRst pin.

Parameters

The $GirthGearLubCtrlCEM template provides different control and supervision parameters to the user to control the functions as per the requirement.

Control

The table describes the control parameters:

Section	Parameter Description	Type	Default	Additional Remarks
Configuration	Spray feedback time (in Min)	UnsignedShort	Five (5)	Allows to change the time delay for Spray feedback time
	Fill feedback time (in Min)	UnsignedShort	Five (5)	Allows to change the Time delay for Fill feedback time
	Negate Signal	Bool	False	True = Negates the input signal.
	Custom Signal Name	String	-	Name used for the generated EFB/DFB and variables.

NOTE: The following points are applicable for Negate signal and Custom Signal Name.

- Nozzle1AFbFill is selectable and by default will be selected.
- Similar $DISignal_UL required for
  - NozzlexAFbFill
  - NozzlexBFbFill
  - NozzlexAFbSpray
  - NozzlexBFbSpray
  - Val1FbPosX
  - Val2FbPosX
  - PumpRun
  - TankLvlOk
  - CtrlAirPressOk
  - GreaseFlowOk
  - LineDPOk

x denotes 1 – 4, X denotes 1 – 2

Supervision

Section	Parameter Description	Type	Default	Additional Remarks
Security	Set Reset	Enum	Operator Confirmed(10)	To Reset Alarm
	Set Parameters	Enum	Engineer(5)	To change the Spray and fill time delay
	Alarm Acknowledge	Enum	Operator(1)	Alarm Acknowledge-ment
	Set Bypass	Enum	Supervisor(2)	To Normal/ Bypass Alarms and Inoperable conditions
Asset, page 18				
Display, page 18				
Configuration	Time Format	String	#####	Allows to change the time format
Alarm	Tank Level Low Description	String	@(Tank level is low)	
	Tank Level Low Severity	Enum	Low	
	Tank Level Low Group	String	Failure	
	Control Air Pressure Low Description	String	@(Control air pressure low)	

Section	Parameter Description	Type	Default	Additional Remarks
	Control Air Pressure Low Severity	Enum	Low	
	Control Air Pressure Low Group	String	Failure	
	Grease Flow Low Description	String	@(Grease flow is low)	
	Grease Flow Low Description Severity	Enum	Low	
	Grease Flow Low Group	String	Failure	
	Line Differential Pressure Low Description	String	@(Line differential pressure is low)	
	Line Differential Pressure Low Severity	Enum	Low	
	Line Differential Pressure Low Group	String	Failure	
	Nozzle 1A Grease Fill Inoperable Description	String	@(Nozzle 1A for grease fill inoperable)	
	Nozzle 1A Grease Fill Inoperable Severity	Enum	Low	
	Nozzle 1A Grease Fill Inoperable Group	String	Failure	
	Nozzle 2A Grease Fill Inoperable Description	String	@(Nozzle 2A for grease fill inoperable)	
	Nozzle 2A Grease Fill Inoperable Severity	Enum	Low	
	Nozzle 2A Grease Fill Inoperable Group	String	Failure	
	Nozzle 3A Grease Fill Inoperable Description	String	@(Nozzle 3A for grease fill inoperable)	
	Nozzle 3A Grease Fill Inoperable Severity	Enum	Low	
	Nozzle 3A Grease Fill Inoperable Group	String	Failure	
	Nozzle 4A Grease Fill Inoperable Description	String	@(Nozzle 4A for grease fill inoperable)	
	Nozzle 4A Grease Fill Inoperable Severity	Enum	Low	

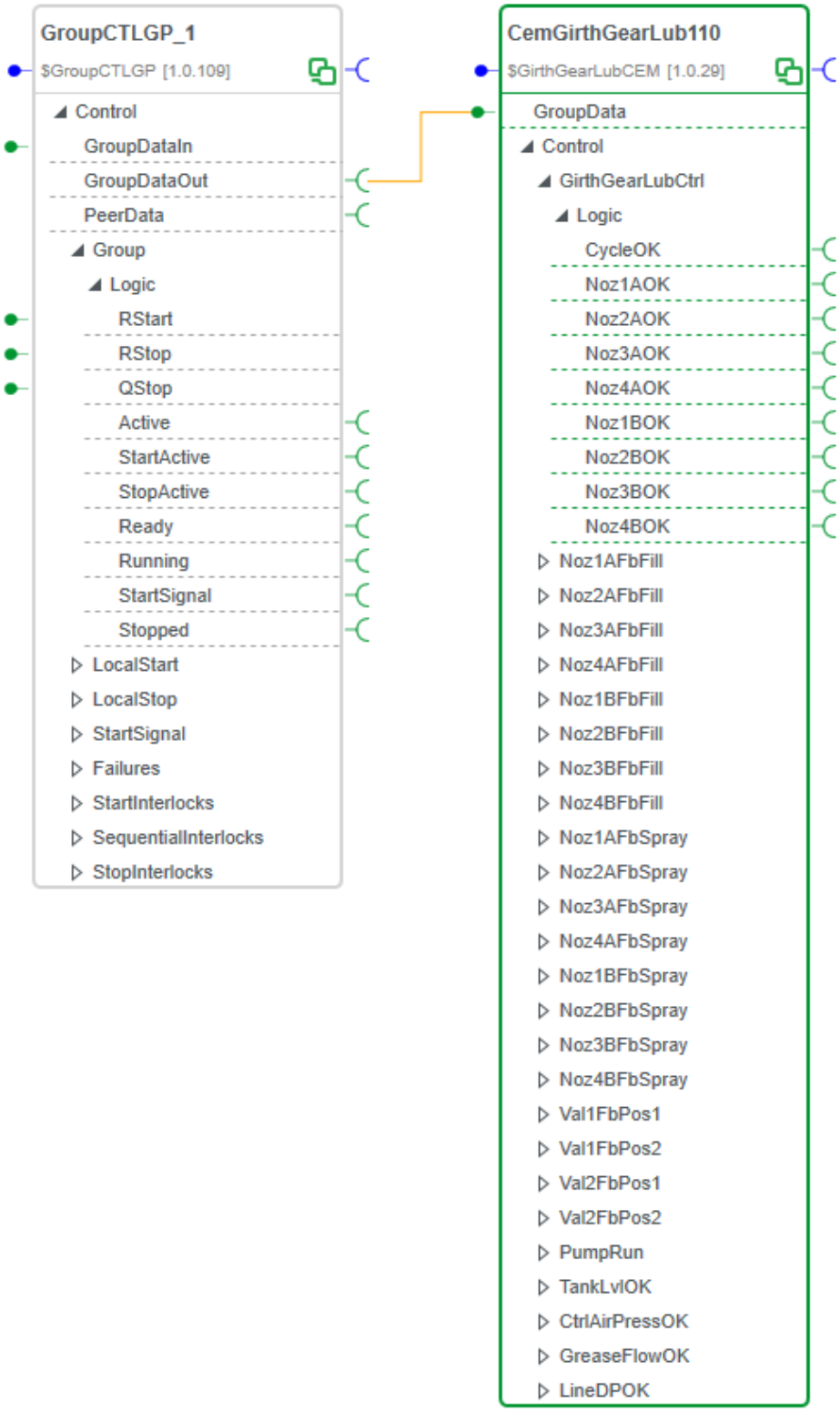
Section	Parameter Description	Type	Default	Additional Remarks
	Nozzle 4A Grease Fill Inoperable Group	String	Failure	
	Nozzle 1B Grease Fill Inoperable Description	String	@(Nozzle 1B for grease fill inoperable)	
	Nozzle 1B Grease Fill Inoperable Severity	Enum	Low	
	Nozzle 1B Grease Fill Inoperable Group	String	Failure	
	Nozzle 2B Grease Fill Inoperable Description	String	@(Nozzle 2B for grease fill inoperable)	
	Nozzle 2B Grease Fill Inoperable Severity	Enum	Low	
	Nozzle 3B Grease Fill Inoperable Description	String	@(Nozzle 3B for grease fill inoperable)	
	Nozzle 3B Grease Fill Inoperable Severity	Enum	Low	
	Nozzle 3B Grease Fill Inoperable Group	String	Failure	
	Nozzle 4B Grease Fill Inoperable Description	String	@(Nozzle 4B for grease fill inoperable)	
	Nozzle 4B Grease Fill Inoperable Severity	Enum	Low	
	Nozzle 4B Grease Fill Inoperable Group	String	Failure	
	Nozzle 1A Grease Spray Inoperable Description	String	@(Nozzle 1A for grease spray inoperable)	
	Nozzle 1A Grease Spray Inoperable Severity	Enum	Low	
	Nozzle 1A Grease Spray Inoperable Group	String	Failure	
	Nozzle 2A Grease Spray Inoperable Description	String	@(Nozzle 2A for grease spray inoperable)	
	Nozzle 2A Grease Spray Inoperable Severity	Enum	Low	

Section	Parameter Description	Type	Default	Additional Remarks
	Nozzle 2A Grease Spray Inoperable Group	String	Failure	
	Nozzle 3A Grease Spray Inoperable Description	String	@(Nozzle 3A for grease spray inoperable)	
	Nozzle 3A Grease Spray Inoperable Severity	Enum	Low	
	Nozzle 3A Grease Spray Inoperable Group	String	Failure	
	Nozzle 4A Grease Spray Inoperable Description	String	@(Nozzle 4A for grease spray inoperable)	
	Nozzle 4A Grease Spray Inoperable Severity	Enum	Low	
	Nozzle 4A Grease Spray Inoperable Group	String	Failure	
	Nozzle 1B Grease Spray Inoperable Description	String	@(Nozzle 1B for grease spray inoperable)	
	Nozzle 1B Grease Spray Inoperable Severity	Enum	Low	
	Nozzle 1B Grease Spray Inoperable Group	String	Failure	
	Nozzle 2B Grease Spray Inoperable Description	String	@(Nozzle 2B for grease spray inoperable)	
	Nozzle 2B Grease Spray Inoperable Severity	Enum	Low	
	Nozzle 2B Grease Spray Inoperable Group	String	Failure	
	Nozzle 3B Grease Spray Inoperable Description	String	@(Nozzle 3B for grease spray inoperable)	
	Nozzle 3B Grease Spray Inoperable Severity	Enum	Low	
	Nozzle 3B Grease Spray Inoperable Group	String	Failure	
	Nozzle 4B Grease Spray Inoperable Description	String	@(Nozzle 4B for grease spray inoperable)	
	Nozzle 4B Grease Spray	Enum	Low	

Section	Parameter Description	Type	Default	Additional Remarks
	Inoperable Severity			
	Nozzle 4B Grease Spray Inoperable Group	String	Failure	
Events	Enable Log	Boolean	False	
Historize	Spray Feedback Time Delay	Boolean	False	
	Fill Feedback Time Delay	Boolean	False	
Alarm	Nozzle 3B Grease Fill Inoperable Description	String	@(Nozzle 3B for grease fill inoperable)	
	Nozzle 3B Grease Fill Inoperable Severity	Enum	Low	
	Nozzle 3B Grease Fill Inoperable Group	String	Failure	
	Nozzle 4B Grease Fill Inoperable Description	String	@(Nozzle 4B for grease fill inoperable)	
	Nozzle 4B Grease Fill Inoperable Severity	Enum	Low	
	Nozzle 4B Grease Spray Inoperable Group	String	Failure	
Events	Enable Log	Boolean	False	
Historize	Spray Feedback Time Delay	Boolean	False	
	Fill Feedback Time Delay	Boolean	False	

Interfaces

The following figure shows an instance of the `$GirthGearLubCtrl` template with the default configuration as it appears in the `Links Editor` of Process Expert:

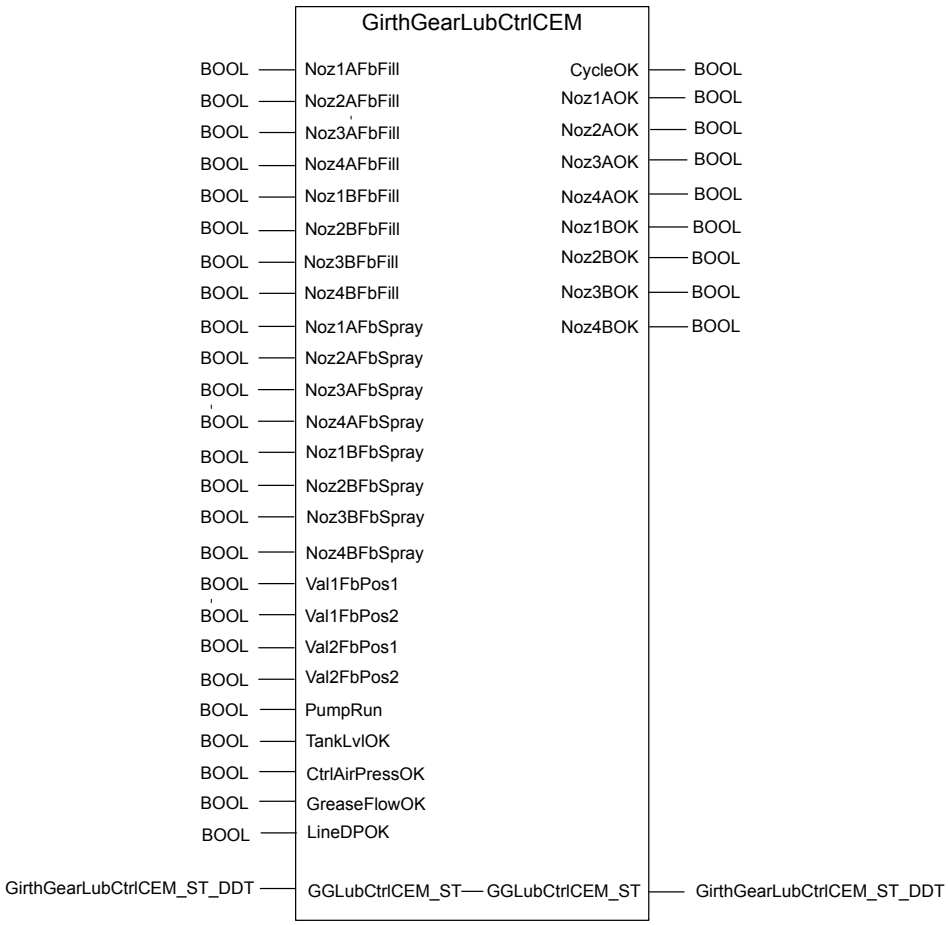


Control

This section describes the `GirthGearLubCtrl` DFB.

DFB Representation

This DFB has been specifically designed for use with the FBD language of the controller.



Inputs

Input Parameter Description

Parameter	Type	Description
`NozxAFbFill` (x: 1...4)	`BOOL`	Fill feedback of nozzle xA (x: 1...4) 1= It is active.
`NozxBFbFill` (x: 1...4)	`BOOL`	Fill feedback of nozzle xB (x: 1...4) 1= It is active.
`NozxAFbSpray` (x: 1...4)	`BOOL`	Spray feedback of nozzle xA (x: 1...4) 1= It is active.
`NozxBFbSpray` (x: 1...4)	`BOOL`	Spray feedback of nozzle xB (x: 1...4) 1= It is active.
`Val1FbPos1`	`BOOL`	Position 1 feedback of valve 1 1= It is active.
`Val2FbPos2`	`BOOL`	Position 2 feedback of valve 1 1= It is active.
`Val2FbPos1`	`BOOL`	Position 1 feedback of valve 2 1= It is active.
`Val2FbPos2`	`BOOL`	Position 2 feedback of valve 2 1= It is active.
`PumpRun`	`BOOL`	Pump run 1= Pump run status is active.
`TankLvlOK`	`BOOL`	Tank level ok

Parameter	Type	Description
		1= Tank level condition is operable.
CtrlAirPressOK	BOOL	Control air pressure ok 1= Control air pressure condition is operable and an alarm is not generated.
GreaseFlowOK	BOOL	Grease flow ok 1= Grease flow condition is operable.
LineDPOK	BOOL	Line differential pressure ok 1= Line differential pressure condition is operable.

Outputs

Output Parameter Description

Parameter	Type	Description
CycleOK	BOOL	Cycle ok status 1 = Girth gear lubrication cycle is ok.
NozxAOK (x: 1...4)	BOOL	Noz xA (x: 1...4) ok status 1 = Nozzle xA (x: 1...4) is ok.
NozxBOK (x: 1...4)	BOOL	Noz xB (x: 1...4) ok status 1 = Nozzle xB (x:1...4) is ok.

Inputs/Outputs

Input/Output Parameter Description

Parameter	Type	Description
GGLubCtrlCEM_ST	GirthGearLubCtrl-CEM_ST_DDT	Structure for Girth Gear Lubrication Control.

GirthGearLubCtrl_ST_DDT Type

The following table describes the structure of the derived data type GirthGearLubCtrl_St for the GirthGearLubCtrl block:

Element	Type	Description
TFbSpray	UINT	Spray feedback time
TFbFill	UINT	Fill feedback time
CFGW	WORD	Configuration word
STW1	WORD	Status word 1
STW2	WORD	Status word 2
STAT	WORD	Status
ALW1	WORD	Alarm word 1
ALW2	WORD	Alarm word 2

Supervision

This section describes the supervision resources and runtime services that are available for the GirthGearLubCtrl block.

Genies

Representation

Graphic symbol	Genie Name
	girthgearlubctrlcem

Faceplates

Representation of Supervision Data

At the beginning of this document, you can find a general description of the graphic elements that are used in the faceplate.

Available Tabs

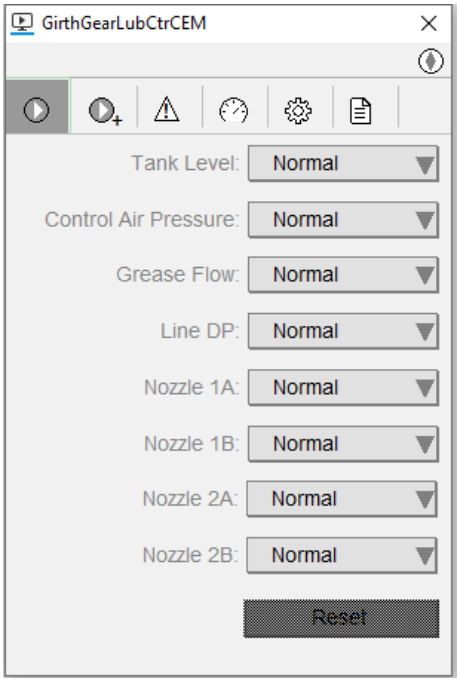
During operation, clicking a gas analyser genie opens a faceplate with the following tabs:

Tabs for core functions:

- Operator Tab
- Extended Operator Tab
- Alarm Tab, page 31
- Measures Tab
- Engineering Tab
- Event Tab

Operator Tab

The figure shows an operator tab:

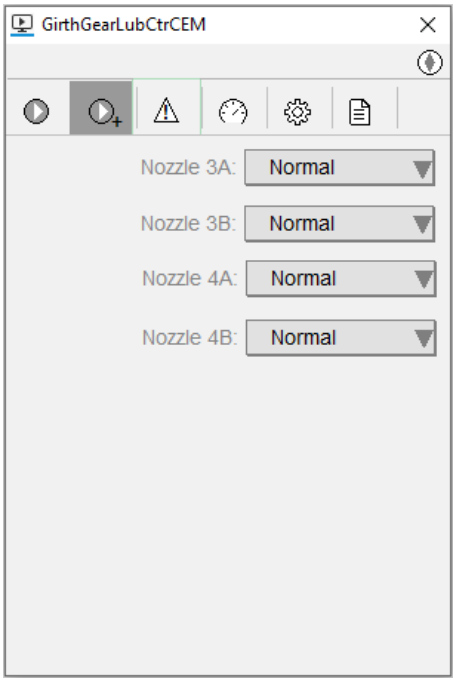


The table describes the control function of the operator tab:

Label	Item Name	Description	Security Parameter
Tank Level	Tanklvlbypass	Bypass the Tank level	Supervisor
Control Air Pressure	CtrlAirPressBypass	Bypass the Control air pressure	Supervisor
Grease Flow	GreaseFlowBypass	Bypass the Grease flow	Supervisor
Line DP	LineDPBypass	Bypass the Line differential pressure	Supervisor
Nozzle 1A	Noz1ABypass	Bypass the Nozzle 1A	Supervisor
Nozzle 1B	Noz1BBypass	Bypass Nozzle 1B	Supervisor
Nozzle 2A	Noz2ABypass	Bypass Nozzle 2A	Supervisor
Nozzle 2B	Noz2BBypass	Bypass Nozzle 2B	Supervisor

Extended Operator Tab

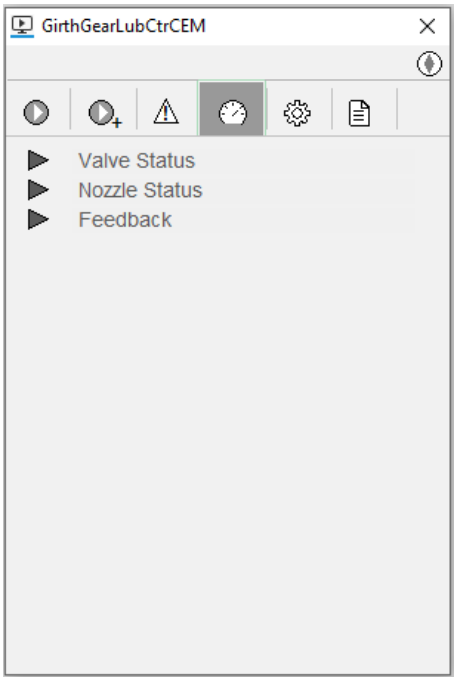
The figure shows the extended operator tab:



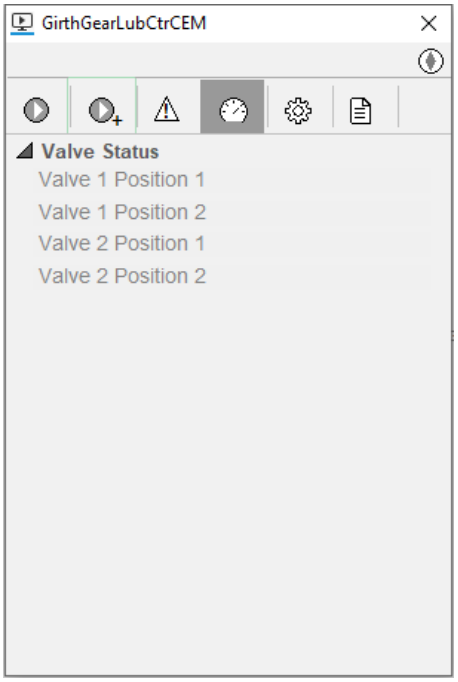
Label	Item Name	Description	Security Parameter
Nozzle 3A	Noz3ABypass	Bypass Nozzle 3A	Supervisor
Nozzle 3B	Noz3BBypass	Bypass Nozzle 3B	Supervisor
Nozzle 4A	Noz4ABypass	Bypass Nozzle 4A	Supervisor
Nozzle 4B	Noz4BBypass	Bypass Nozzle 4B	Supervisor

Measures Tab

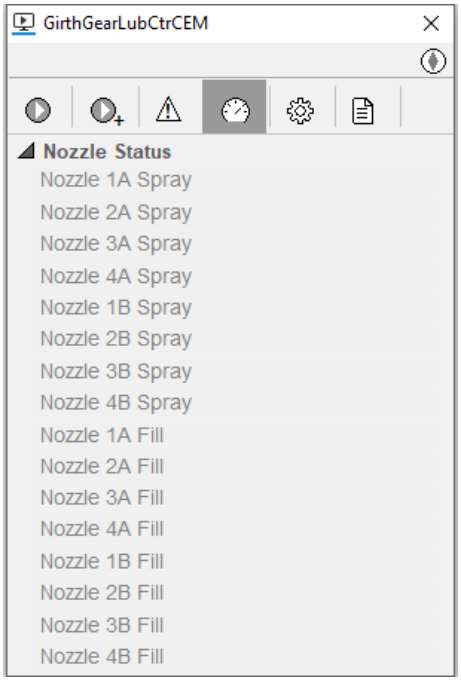
The figure shows the measures tab:



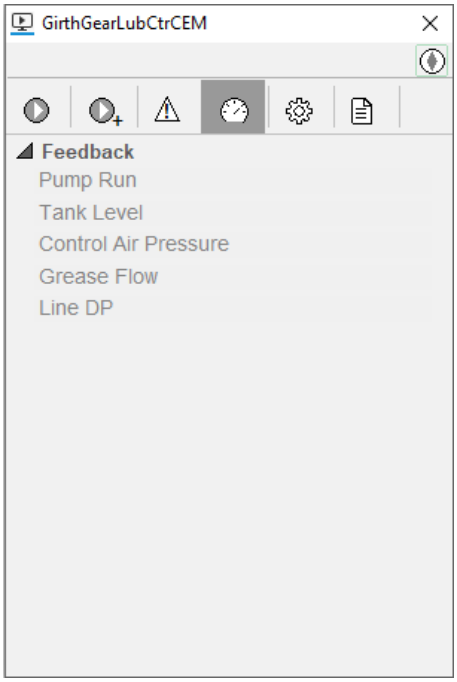
The figure shows the measures tab with valve status:



The figure shows the measures tab with nozzle status:

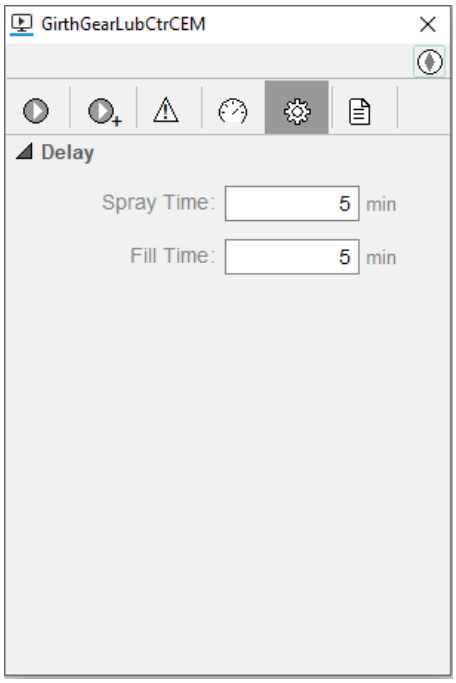


The figure shows the measures tab with feedback:



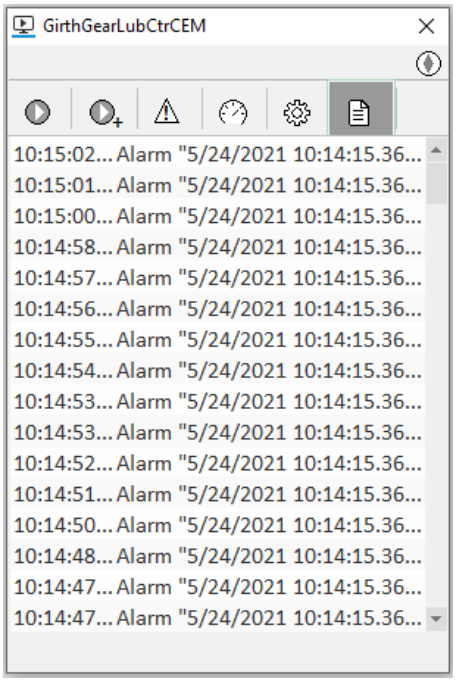
Engineering Tab

The figure shows an engineering tab:



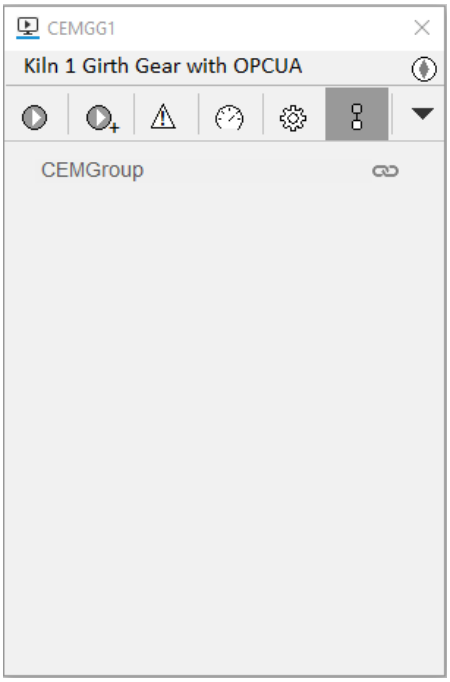
Event Tab

The figure shows an event tab:



Generic Group Tab

The figure shows a generic group tab:



NOTE: This faceplate appears and group data interface is connected.

Items

Overview

This section describes the variables, abnormal conditions, and events items of Queue.

Variables

The table describes the variable items that are used by supervision components:

Item name	Description	Enumeration	Address
Bit items derived from StatusWord.			
Noz1AOk	Tank level low.	–	Bit2
Noz2AOk	Control air pressure is low.	–	Bit3
Noz3AOk	Grease flow is low.	–	Bit4
Noz4AOk	Line differential pressure is low.	–	Bit5
Bit items derived from Alarm word 1			
AlwTankLvlLow	Tank level low.	–	Bit0
AlwCtrlAirPres-sLow	Control air pressure is low.	–	Bit1
AlwGreaseFlowLow	Grease flow is low.	–	Bit2
AlwLineDPLow	Line differential pressure is low.	–	Bit3
Bit items derived from Alarm word 2			
AlwNozXAFillFail	Nozzle XA for grease fill inoperable.	–	Bitx
AlwNozXBFillFail	Nozzle XB for grease fill inoperable.	–	Bitx
AlwNoz1XASpray-Fail	Nozzle XA for grease Spray inoperable.	–	Bitx
AlwNoz1BSpray-Fail	Nozzle XB for grease Spray inoperable.	–	Bitx
NOTE: X ranges from 1 to 4, x ranges from 0 to 15			
Bit items derived from STW1 word			
PumpRun	Feedback.	–	Bit0
TankLevel	Feedback.	–	Bit1
ControlAirPres-sure	Feedback.	–	Bit2
GreaseFlow	Feedback.	–	Bit3
LineDP	Feedback.	–	Bit4
Valve1Position1	Valve Status.	–	Bit5
Valve1Position2	Valve Status.	–	Bit7
Valve2Position1	Valve Status.	–	Bit6
Valve2Position2	Valve Status.	–	Bit8
Bit items derived from STW2 word			
NozzleXAFill	Indicates the nozzle XA Fill status.	–	Bitx
NozzleXBFill	Indicates the nozzle XB Fill status.	–	Bitx
NozzleXASpray	Indicates the nozzle XA Spray status.	–	Bitx
NozzleXBSpray	Indicates the nozzle XB Spray status.	–	Bitx
NOTE: X ranges from 1 to 4, x ranges from 0 to 15			

MultiFuelCtrl - Multi Fuel Control

What's in This Chapter

Template

This section describes the functionality, parameters and interfaces of the $MultiFuelCtrlCEM template.

Description

General Description

The $MultiFuelCtrlCEM control module template is used to control the ratio of different fuels of different types. The main application is managing kiln burner fuel ratio.

The $MultiFuelCtrlCEM control module template can control upto six different fuels. It generates six set points to individual actuators that indirectly maintain a ratio between six different fuels. It depends on the configuration of associated devices and the system requirements.

Functional Description

The $MultiFuelCtrlCEM control module template supports four modes of operation:

- Auto mode 1
- Auto mode 2
- Manual mode
- External mode

Auto Mode 1 Operation

The total set point is fed to the block through input pin (ExtTotalSP) or by the operator from supervision (ST.ManTotalSP) based on the external set point enabled or disabled.

The individual set point ratios AutoRatio(X) are entered in percentage by the operator from the faceplate. When the command update ratio (UpdateRatio) is given and OPEn(X) is enabled, the ratio is updated to the final ratio of the outputs Ratio (X).

Set point ST.SPHeat(X) is calculated by the formulae:

ST.SPHeat = TotalSP*Ratio/100.

When the ratio or the total set point is changed, the heat set points are adjusted to maintain the ratio. The final fuel set point to the respective actuators is calculated by :

SP (X) = SPHeat/CalValx(X).

The generated set points are restricted to the limits ST.SPH and ST.SPL(X) entered by the operator from faceplate. If the total ratio is not equal to 100% for more than pre-defined time (ST.TRatioError), then an alert is generated.

NOTE: X ranges from 1 to 6.

Auto Mode 2 Operation

This is a type of expert mode, where the controller defines the ratio of individual fuels based on the priority, high and low limits of the fuel. The total set point is fed to the block through input pin (ExtTotalSP) or by the operator from supervision (ST.ManTotalSP) based on the external set point enable/disable. The operator enters the priority of the fuel and the set point limits from the faceplate. The controller calculates these inputs with the in-built algorithm and generates the heat set points for each fuel. When the ratio or the total set point is changed, the heat set points are adjusted.

The final fuel set point to the respective actuators is calculated by :

SP (X) = SPHeat/CalVal(X).

The generated set points are restricted to the limits ST.SPH and ST.SPL(X) entered by the operator from faceplate. If the total ratio is not equal to 100% for more than pre-defined time (ST.TRatioError), then an alert is generated.

NOTE: X ranges from 1 to 6.

Manual Mode Operation

When the block is in Manual mode, the manual set point ST.ManSP (X) is entered by the operator from the faceplate, and is generated as the output set point SP(X).

Manual set point entry from faceplate is restricted to the limits ST.SPH and ST.SPL(X).

NOTE: X ranges from 1 to 6.

External Mode Operation

When the block is in External mode, the external set point is given as an input to the block and is generated as the final fuel set point SP(X) to the respective actuators.

NOTE: X ranges from 1 to 6.

Limits and Inoperable Conditions

The block continuously monitors the process feedback value PV(X). The value of the PV(X) and total process values are displayed in the faceplate (PV, TotalPV) of the block.

If the process value PV(X) deviates from SP(X) beyond limit (ST.DevH) for a longer time (TDev), then an alarm is generated.

In the Auto 1 mode or Auto 2 mode, if the calculated set point and output set points are not equal, then an alarm is generated. The alert and alarms are indicated in the supervision. The details of the active alarms are shown in the Alarm tab of the faceplate.

NOTE: X ranges from 1 to 6.

Parameters

The $MultiFuelCtrlCEM template provides different control and supervision parameters to the user to control the functions as per the requirement.

Control

The table describes the control parameters:

Section	Parameter Description	Type	Default	Additional Remarks
Configuration	Default Mode	ENUM	Auto1	Allows to change the modes
	Maximum Deviation	Float	50.0	Allows to change the maximum deviation
	Time Delay for Deviation error (in Min)	Un-signed-Short	2	Allows to change the time delay for

Section	Parameter Description	Type	Default	Additional Remarks
				deviation error detected.
	Time Delay for Ratio error (in Min)	Un-signed-Short	2	Allows to change the time delay for ratio error detected.
	Time Delay for Setpoint error (in Min)	Un-signed-Short	2	Allows to change time delay for setpoint error detected.
Calorific Value	Fuel {X}	Float	0.0	Allows to change calorific value for fuel [X}.
Output Priority	Fuel {X}	ENUM	{Y}({X})	Allows to change output priority for fuel {X}.
Setpoint High Limit	Fuel {X}	Float	100.0	Allows to change the setpoint high limit for fuel {X}
Setpoint Low Limit	Fuel {X}	Float	0.0	Allows to change the setpoint low limit for fuel {X}
Heat Setpoint High Limit	Fuel {X}	Float	100.0	—
Heat Setpoint Low Limit	Fuel {X}	Float	0.0	—

NOTE:

- X ranges from 1 to 6.
- Y ranges from one to six.

NOTE: ENUM values for default mode parameter is Auto1 (1) / Auto2 (2) / Manual (3) / External (4).

Supervision

Settings

The table describes the settings parameters:

Section	Parameter Description	Type	Default	Additional Remarks
Security	Set Reset	Enum	Operator Confirmed (10)	Privilege to reset. This parameter is used in **Operator**tab.
	Set Parameters		Engineer (5)	Privilege to set parameters.
	Alarm Acknowledge		Operator (1)	Privilege to set alarm acknowledge. This parameter is used in **Operator**tab.
	Set Bypass		Supervi-sor (2)	Privilege to bypass conditions.

Asset and Display Parameters

For more details, refer to the topic describing the parameters of asset and display, page 18.

Data

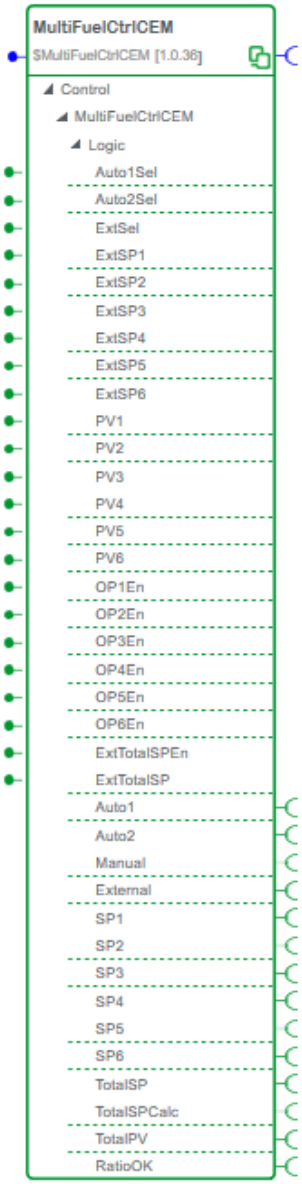
The table describes the data parameters:

Section	Parameter Description	Type	Default	Additional Remarks
Configuration	Total Setpoint – Engineering Zero	Float	0.0	—
	Total Setpoint – Engineering Full	Float	100..0	—
	Total Setpoint – Engineering Unit	String	GJ/Hr	—
	Total Setpoint – Format	String	####. ##EU	—
	Auto Ratio Fuel {X} – Engineering Zero	Float	0.0	—
	Auto Ratio Fuel {X} – Engineering Full	Float	100..0	—
	Auto Ratio Fuel {X} – Engineering Unit	String	%	—
	Auto Ratio Fuel {X} – Format	String	####. ##EU	—
	Setpoint Fuel {X} – Engineering Zero	Float	0.0	—
	Setpoint Fuel {X} – Engineering Full	Float	100..0	—
	Setpoint Fuel {X} – Engineering Unit	String	%	—
	Setpoint Fuel {X} – Format	String	####. ##EU	—
	Time Format	String	#####	—
	Fuel {X} Name	String	Fuel{X}	—
Alarm	SP{X} Limit *Error* Description	String	@(SP{X} limit *error*)	—
	SP{X} Limit *Error* Severity	Enum	Low	—
	SP{X} Limit *Error* Group	String	*Failure*	—
	PV{X} Maximum Deviation Exceeded Description	String	@(PV{X} maximum deviation excee-ded)	—
	PV{X} Maximum Deviation Exceeded Severity	Enum	Low	—
	PV{X} Maximum Deviation Exceeded Group	String	*Failure*	—
	Ratio *Error* Description	String	@(Ratio *error*)	—
	Ratio *Error* Severity	Enum	Low	—
	Ratio *Error* Group	String	*Failure*	—
	Multiple Mode Selected Description	String	@ (Multiple mode selected)	—
	Multiple Mode Selected Severity	Enum	Low	—
	Multiple Mode Selected Group	String	*Failure*	—

Section	Parameter Description	Type	Default	Additional Remarks
	Setpoint Out of Limit Description	String	@ (Setpoint out of limit)	—
	Setpoint Out of Limit Severity	Enum	Low	—
	Setpoint Out of Limit Group	String	*Failure*	—
Deadband	Total Setpoint	Un-signed-Short	0	—
	Ratio Setpoint			
	Calculated Setpoint			
	Total Setpoint			
	Setpoint			
	Process Value Fuel {X}			
Events	Enable Log	Boolean	False	—
Trends	Disable Fuel {X} SP	Boolean	False	—
	Disable Total SP			
	Disable Total PV			
Historize	Setpoints	Boolean	False	—
	Total PVs			
	Ratios			
NOTE: X ranges from 1 to 6.				

Interfaces

This figure shows the $MultiFuelCtrlCEM template as it appears in Links Editor and an example of other templates connected to it:

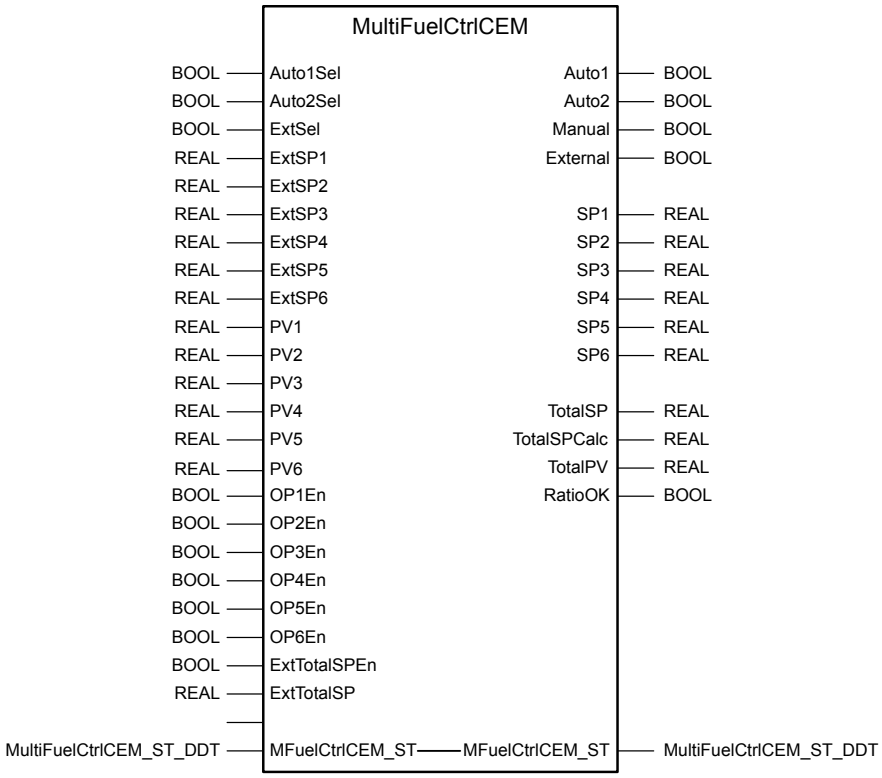


Control

This section describes the `$MultifuelCtrlCEM` DFB.

DFB Representation

This DFB has been specifically designed for use with the FBD language of the controller.



Inputs

Input Parameter Description

Name	Type	Description
`Auto1Sel`	`BOOL`	1 = Auto 1 mode is activated
`Auto2Sel`	`BOOL`	1 = Auto 2 mode is activated
`ExtSel`	`BOOL`	1 = External mode is activated
`ExtSP{X}`	`REAL`	External setpoint {X}
`PV{X}`	`REAL`	Process value of fuel {X}
`OP{X}En`	`BOOL`	1 = Output {X} is controlled by the block.
`ExtTotalSPEn`	`BOOL`	1 = Block considers the input coming from `ExtTotalSP` for total setpoint value in Auto 1/ Auto 2 mode.
`ExtTotalSP`	`REAL`	External total setpoint
NOTE: X ranges from 1 to 6.		

Outputs

Output Parameter Description

Name	Type	Description
`Auto1`	`BOOL`	1 = Auto 1 mode active status
`Auto2`	`BOOL`	1 = Auto 2 mode active status
`Manual`	`BOOL`	1 = Manual mode active status
`External`	`BOOL`	1 = External mode active status
`SP{X}`	`REAL`	Setpoint for fuel {X}

Name	Type	Description
TotalSP	REAL	Total setpoint (in GJ/hr)
TotalSPCalc	REAL	Calculated total setpoint (in GJ/hr)
TotalPV	REAL	Total process value (in GJ/hr)
RatioOK	BOOL	1 = The sum of ratios for enabled outputs is 100%.
NOTE: X ranges from 1 to 6.		

Inputs/Outputs

Input/Output Parameter Description

Name	Type	Description
MFuelCtrlCEM_ST	MultiFuelCtrlCEM_ST_DDT	Structure for multi fuel control

MultiFuelCtrlCEM_ST_DDT

The following table describes the structure of the derived data type MultiFuelCtrlCEM_ST_DDT for the $MultiFuelCtrlCEM DFB:

Name	Bit Name	Bit	Type	Description
ManTotalSP	—	—	REAL	Manual total setpoint (in GJ/hr)
TotalSP	—	—	REAL	Total setpoint (in GJ/hr)
TotalSPCalc	—	—	REAL	Calculated total setpoint (in GJ/hr)
TotalPV	—	—	REAL	Total process value (in GJ/hr)
AutoRatio {X}	—	—	REAL	Auto mode ratio setpoint for fuel {X}
Ratio{X}	—	—	REAL	Final ratio for fuel {X}
CalVal{X}	—	—	REAL	Calorific value of fuel {X}
ManSP{X}	—	—	REAL	Manual mode setpoint for fuel {X}
SP{X}Heat	—	—	REAL	Heat setpoint for fuel {X} (in GJ/hr)
SP{X}HeatH	—	—	REAL	Heat setpoint high limit for fuel {X} (in GJ/hr)
SP{X}HeatL	—	—	REAL	Heat setpoint low limit for fuel {X} (in GJ/hr)
SP{X}Calc	—	—	REAL	Calculated setpoint for fuel {X}
SP{X}	—	—	REAL	Setpoint for fuel {X}
SP{X}H	—	—	REAL	Setpoint high limit for fuel {X}
SP{X}L	—	—	REAL	Setpoint low limit for fuel {X}
PV{X}	—	—	REAL	Process value of fuel {X}
TotalSPH	—	—	REAL	Total setpoint high limit (in GJ/hr)
TotalSPL	—	—	REAL	Total setpoint low limit (in GJ/hr)
DevH	—	—	UINT	Deviation high limit
TDev	—	—	UINT	Time delay for deviation error detection (in min)
TRatioError	—	—	UINT	Time delay for ratio error detection (in min)

Name	Bit Name	Bit	Type	Description
TSPError	—	—	UINT	Time delay for setpoint error detection (in min)
OP{X} Priority	—	—	INT	Priority of usage of output {X}
CFGW	—	—	WORD	Read/Write access. Configuration word
	Auto1	0	BOOL	1 = Auto 1 mode selected
	Auto2	1	BOOL	1 = Auto 2 mode selected
	ManMode	2	BOOL	1 = Manual mode selected
	ExtMode	3	BOOL	1 = External mode selected
	UpdateRatio	4	BOOL	1 = Update ratio command
	ManTotSPTrk-En	6	BOOL	1 = Manual total setpoint enable command
	AlmRst	7	BOOL	1 = Reset alarm
STW1	—	—	WORD	Read access. Status word 1
	Auto1	0	BOOL	1 = Auto 1 mode activated
	Auto2	1	BOOL	1 = Auto 2 mode activated
	Manual	2	BOOL	1 = Manual mode activated
	External	3	BOOL	1 = External mode activated
	Alarminhibit	13	BOOL	1 = All alarms are inhibited. No alarms are generated by the block.
STAT	—	—	WORD	Read access. Status word
	Wake	0	BOOL	1 = Multi fuel control is in Wake-up state.
	Auto12Ext	1	BOOL	1 = Auto 1 or Auto 2 or External mode activated.
	ExtTotalSPEn	2	BOOL	1 = Block considers the inputs coming from ExtTotalSP for total setpoint value in Auto 1/ Auto 2 mode.
	OP{X}En	{Y}	BOOL	1 = Output {X} is controlled by the block.
	Require-dRearm	10	BOOL	1 = Required Rearm
	RatioOK	12	BOOL	1 = The sum of ratios for enabled outputs is 100%.
	UpdateRa-tioEnStat	13	BOOL	1 = Update Ratio enabled
	WarningCon-ditions	14	BOOL	1= Alert is off
	AlarmCondi-tions	15	BOOL	1= Alarm is off
ALW1	—	—	WORD	Read access. Alarm word
	SP{X}LimErr	{Z}	BOOL	SP {X} limit error
	MaxPV{X}Dev	{P}	BOOL	PV {X} maximum deviation exceeded
	RatioErr	12	BOOL	Ratio error detection
	ModeForce-Fail	13	BOOL	Multiple mode selected
	SPOutofLim	14	BOOL	Setpoint out of limit

Name	Bit Name	Bit	Type	Description
Dummy	—	—	WORD	Not used

NOTE:

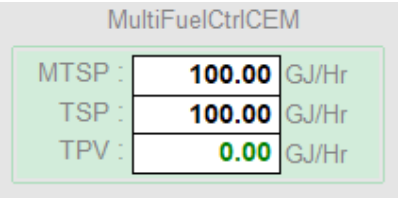
- X ranges from 1 to 6.
- Y ranges from 4 to 9.
- Z ranges from 0 to 5.
- P ranges from 6 to 11.

Supervision

This section describes the supervision resources and runtime services that are available for the $MultiFuelCtrlCEM block.

Genies

Representation

Graphic symbol	Genie name
	multifuelctrlcem_m

Faceplates

Representation of Supervision Data

At the beginning of this document, you can find a general description of the graphic elements that are used in the faceplate.

Available Tabs

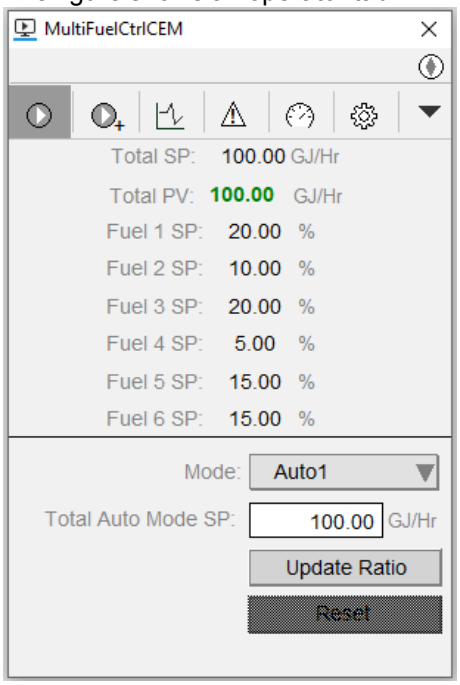
During operation, clicking a multi fuel control genie opens a faceplate with the following tabs:

Tabs for core functions:

- Operator
- Extended Operator
- Trend
- Alarm, page 31
- Measures
- Engineering
- Event

Operator Tab

The figure shows an operator tab:

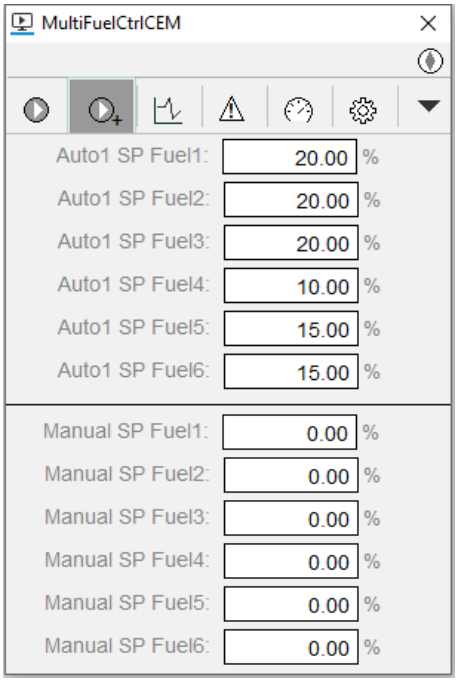


The table describes the control function of an operator tab:

Label	Item Name	Description	Security parameter, page 75
Total SP	TSP	Total Setpoint	—
Total PV	TPV	Total PV	—
Fuel{X} SETPOINT	Fuel{X}SETPOINT	Fuel{X} SETPOINT	—
Mode	MultifuelMode	Mode Selection	Operator
Total Auto Mode SP	TAModeSP	Total Auto Mode Setpoint	Operator
Update Ratio	URatio	Update Ratio updates all modified ratios.	Operator
Reset	AlmRst	To reset the alarms	Operator
Probe Retract	ProbeRetract	It allows to give command for Emergency Probe Retract.	Operator
NOTE: X ranges from 1 to 6.			

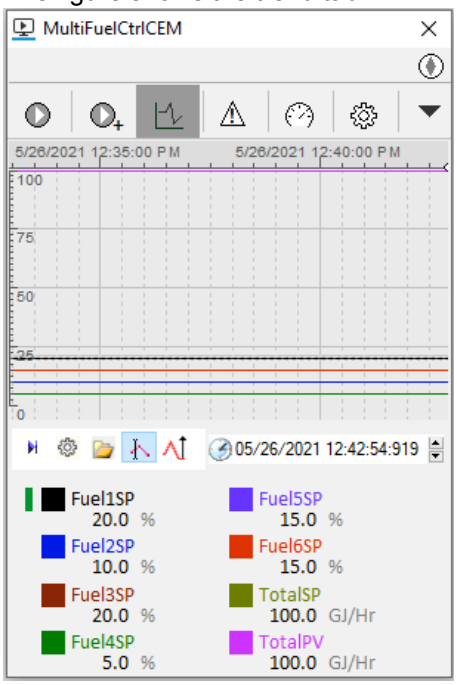
Extended Operator Tab

The figure shows an extended operator tab:



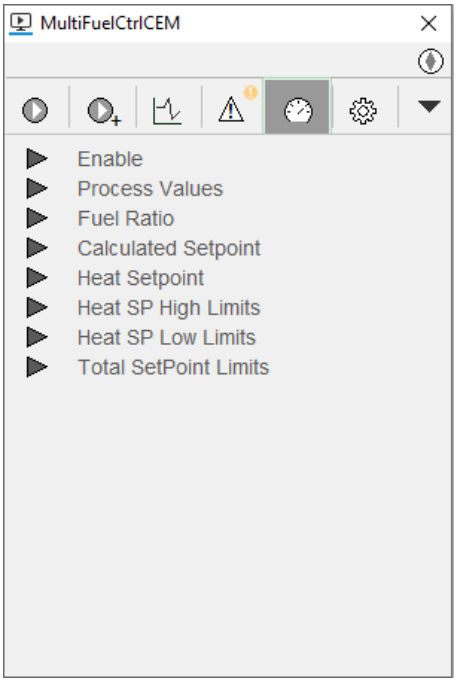
Trend Tab

The figure shows the trend tab:

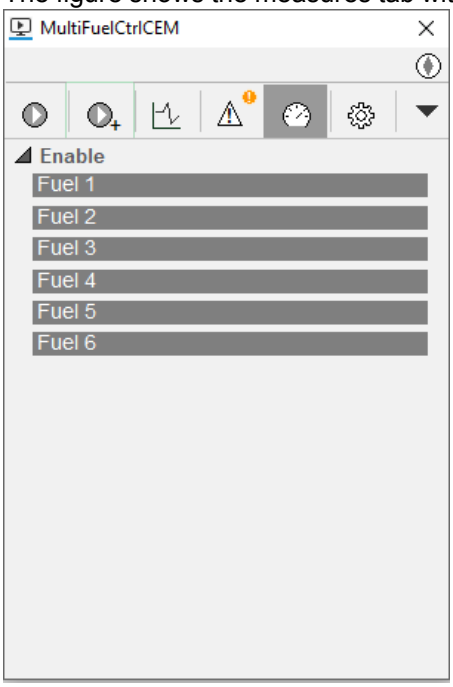


Measures Tab

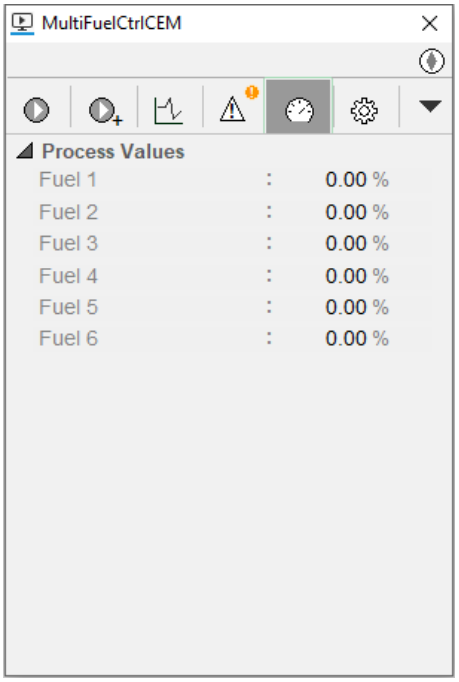
The figure shows the measures tab:



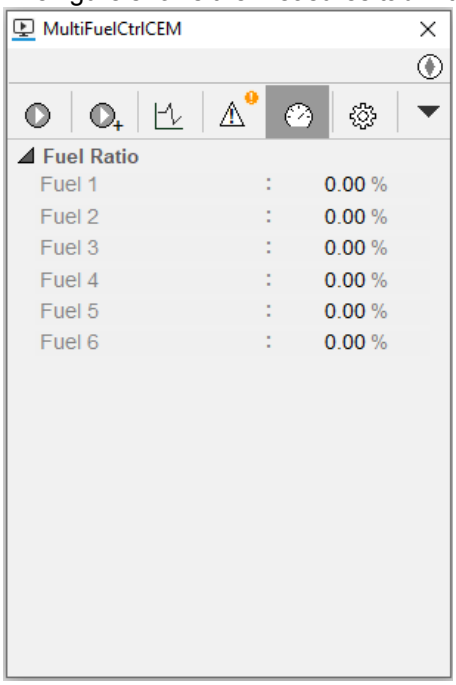
The figure shows the measures tab with enable option:



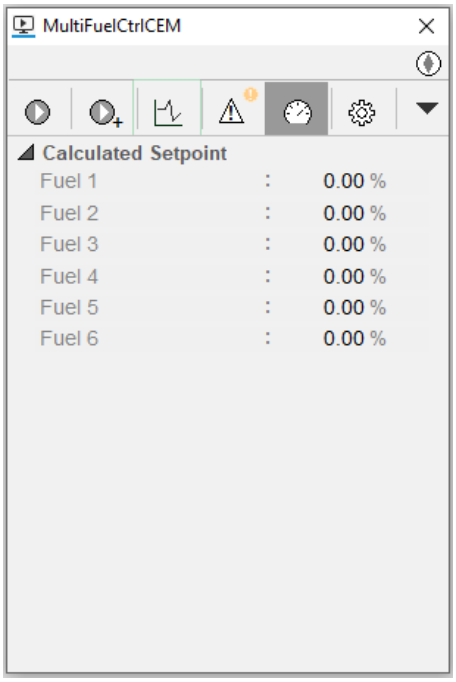
The figure shows the measures tab with process values:



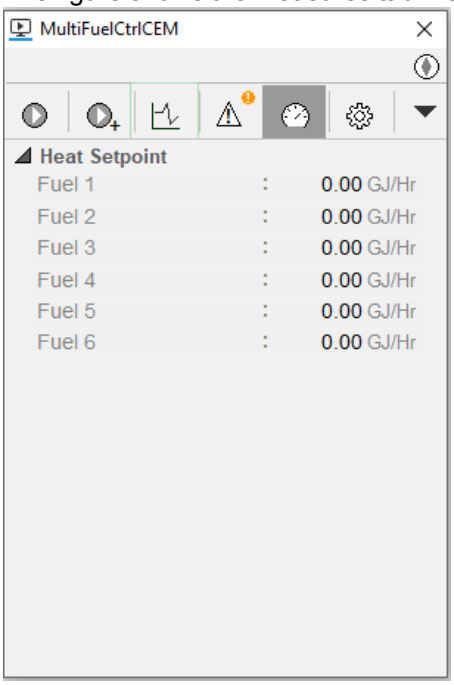
The figure shows the measures tab with fuel ratios:



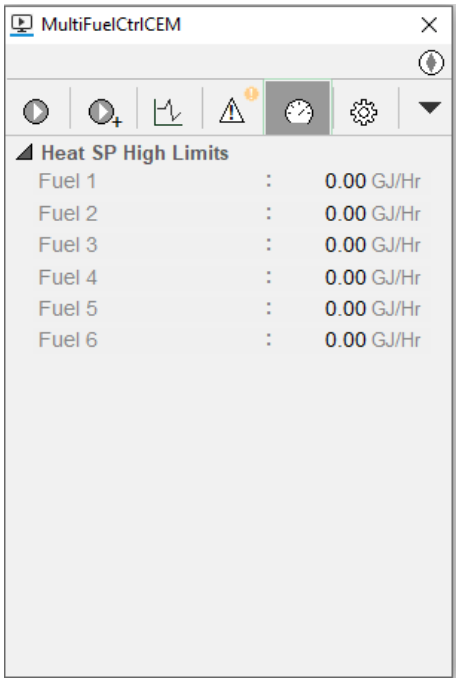
The figure shows the measures tab with calculated setpoint:



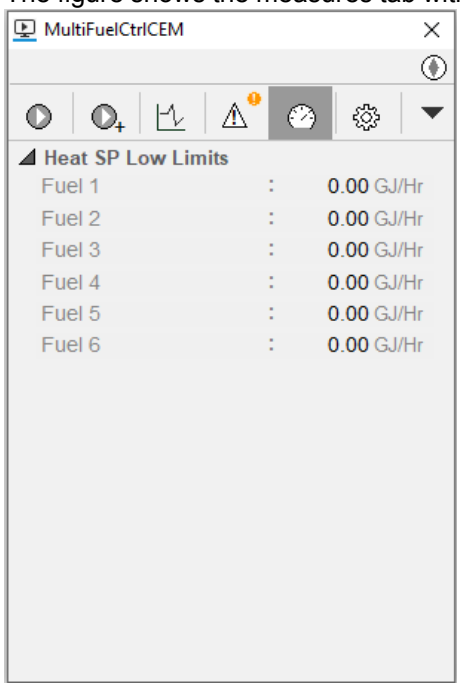
The figure shows the measures tab with heat setpoint:



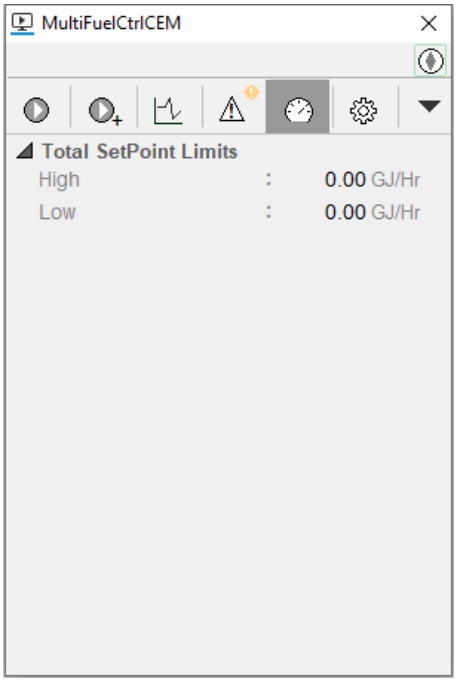
The figure shows the measures tab with heat SP high limits:



The figure shows the measures tab with heat SP low limits:

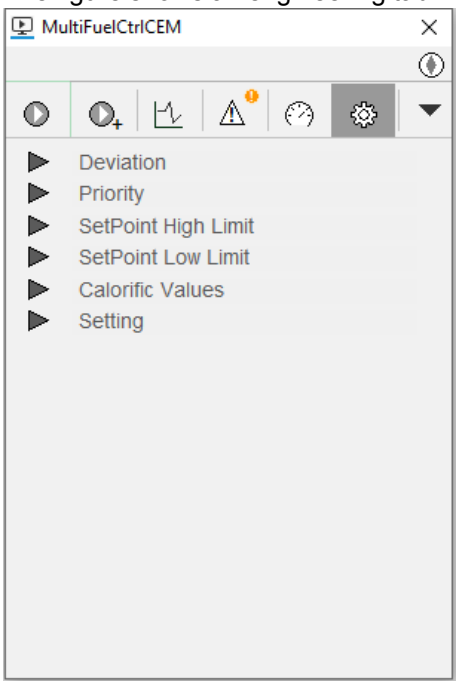


The figure shows the measures tab with total setpoint limits:

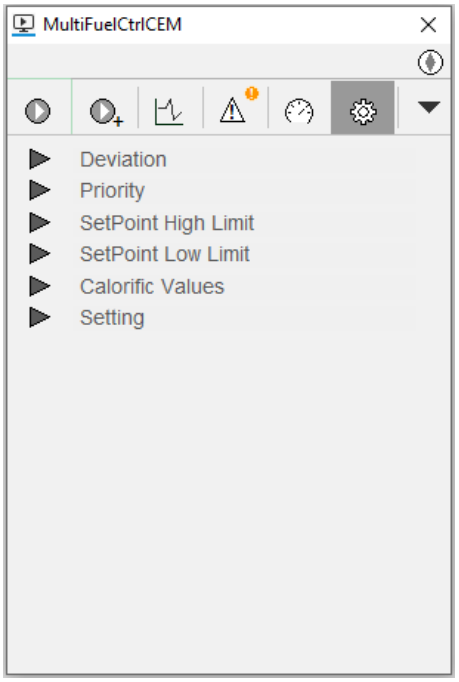


Engineering Tab

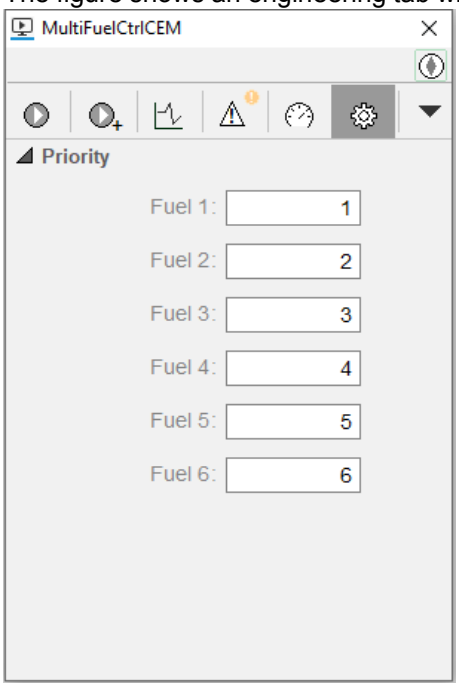
The figure shows an engineering tab:



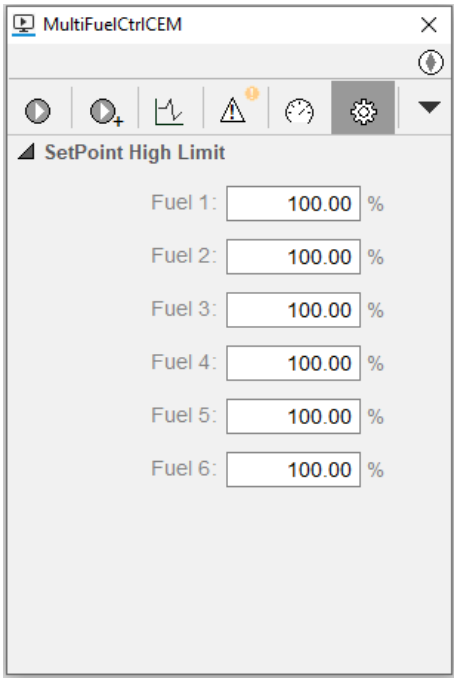
The figure shows an engineering tab with deviation:



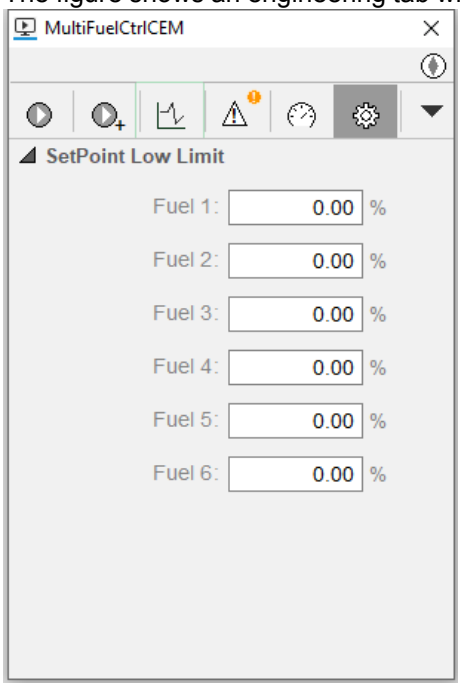
The figure shows an engineering tab with priority:



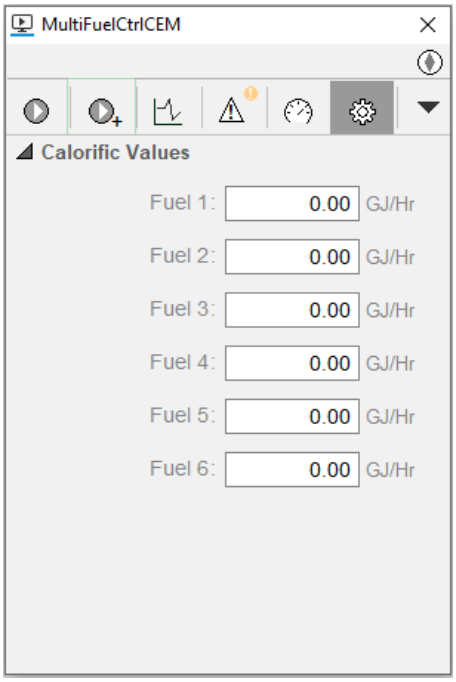
The figure shows an engineering tab with setpoint high limit:



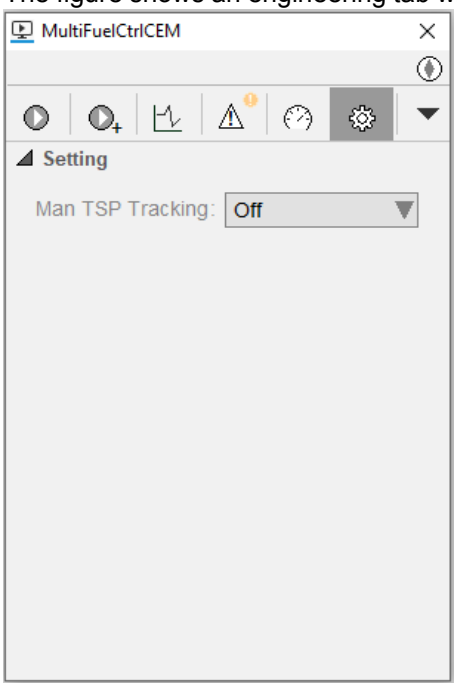
The figure shows an engineering tab with setpoint low limit:



The figure shows an engineering tab with calorific value:

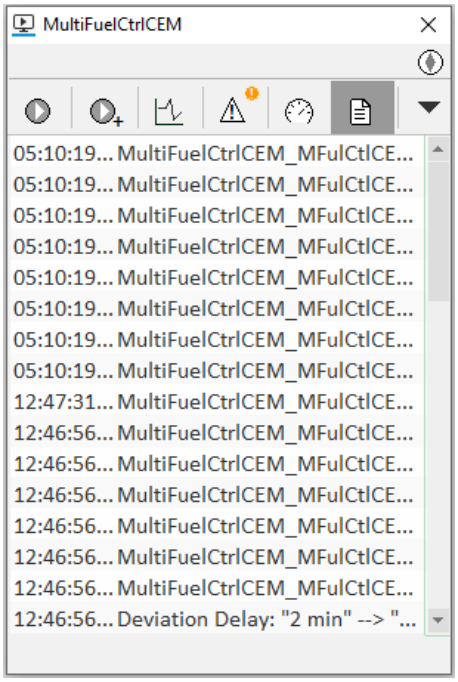


The figure shows an engineering tab with settings:



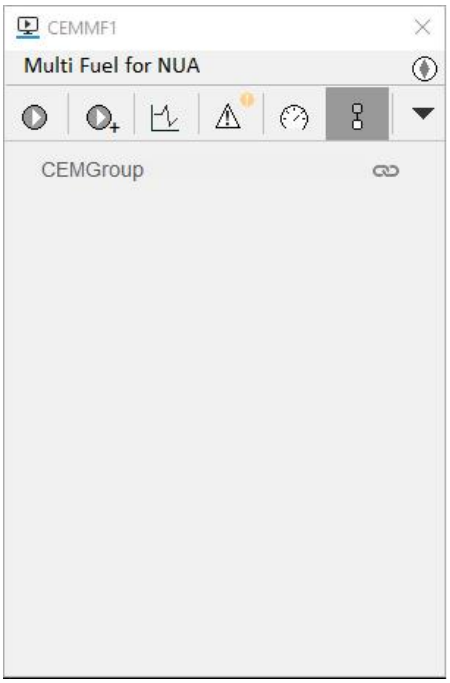
Event Tab

The figure shows an event tab:



Generic Group Tab

The figure shows a generic group tab:



NOTE: This faceplate appears and group data interface is connected.

Items

Overview

This section describes the variables, abnormal conditions, and events items of multi fuel control.

Variables

The table describes the variable items that are used by supervision components:

Item name	Description	Enumeration	Address
Bit items derived from output			
Fuel{X}SP	Fuel {X} SP	–	–
TSP	Total setpoint	–	–
TPV	Total process value	–	–
MTSP	Total auto mode setpoint	–	–
ASPFuel{X}	Auto1 setpoint fuel {X}	–	–
MSPFuel{X}	Manual setpoint fuel {X}	–	–
Bit items derived from Engineer			
MaxDev	Maximum deviation	–	–
DevDelay	Deviation delay	–	–
RErrDelay	Ratio error detection delay	–	–
SPErrDelay	SP error detection delay	–	–
Fuel{X}Priority	Fuel {X} priority	–	–
Fuel{X}SPHL	Setpoint high limit for fuel {X}	–	–
Fuel{X}SPLL	Setpoint low limit for fuel {X}	–	–
Fuel{X}CalVal	Calorific value of fuel {X}	–	–
Bit items derived from Mode			
MultifuelMode	Mode of operation	–	–
Bit items derived from EquipStatus			
EquipStatus	Control module status	–	–
Bit items derived from Equipstate			
Equipstate	Output status	–	–
Bit items derived from STAT Word			
Fuel{X}	Fuel {X} enable	–	{Y}
URatio	–	–	Bit13
Auto12Ext	–	–	Bit1
Bit items derived from Measures			
Fuel{X}Ratio	Fuel {X} ratio	–	–
Fuel{X}HSP	Fuel {X} Heat Setpoint	–	–
Fuel{X}CalSP	Fuel {X} calculated setpoint	–	–
Fuel{X}PV	Fuel {X} process values	–	–
Fuel{X}HL	Fuel {X} heat sp high limits	–	–
Fuel{X}LL	Fuel {X} heat sp low limits	–	–
TSPHL	Total setpoint high limits	–	–
TSPLL	Total setpoint low limits	–	–
Bit items derived from CFGW			

Item name	Description	Enumeration	Address
Configuration-Word	–	–	–
CommandSetpoint	Manual total setpoint enable command Man TSP tracking Setting	–	–
Rearm	Alarm reset	–	Bit7
RequireRearm	Reset required	–	Bit10
Output	Update ratio	–	Bit4
Auto1	Mode	–	Bit0
Auto2	Mode	–	Bit1
Manual	Mode	–	Bit2
External	Mode	–	Bit3
Bit items derived from Abnormal			
AbnormalOperatorTab	–	–	Bit8
Bit items derived from Actual Items			
Configuration-Word	–	–	–
STW1	–	–	–

NOTE:

- X ranges from 1 to 6.
- Y ranges from 4 to 9.

Usage

What's in This Part

Overview

This chapter describes the function objects.

Use Case Examples

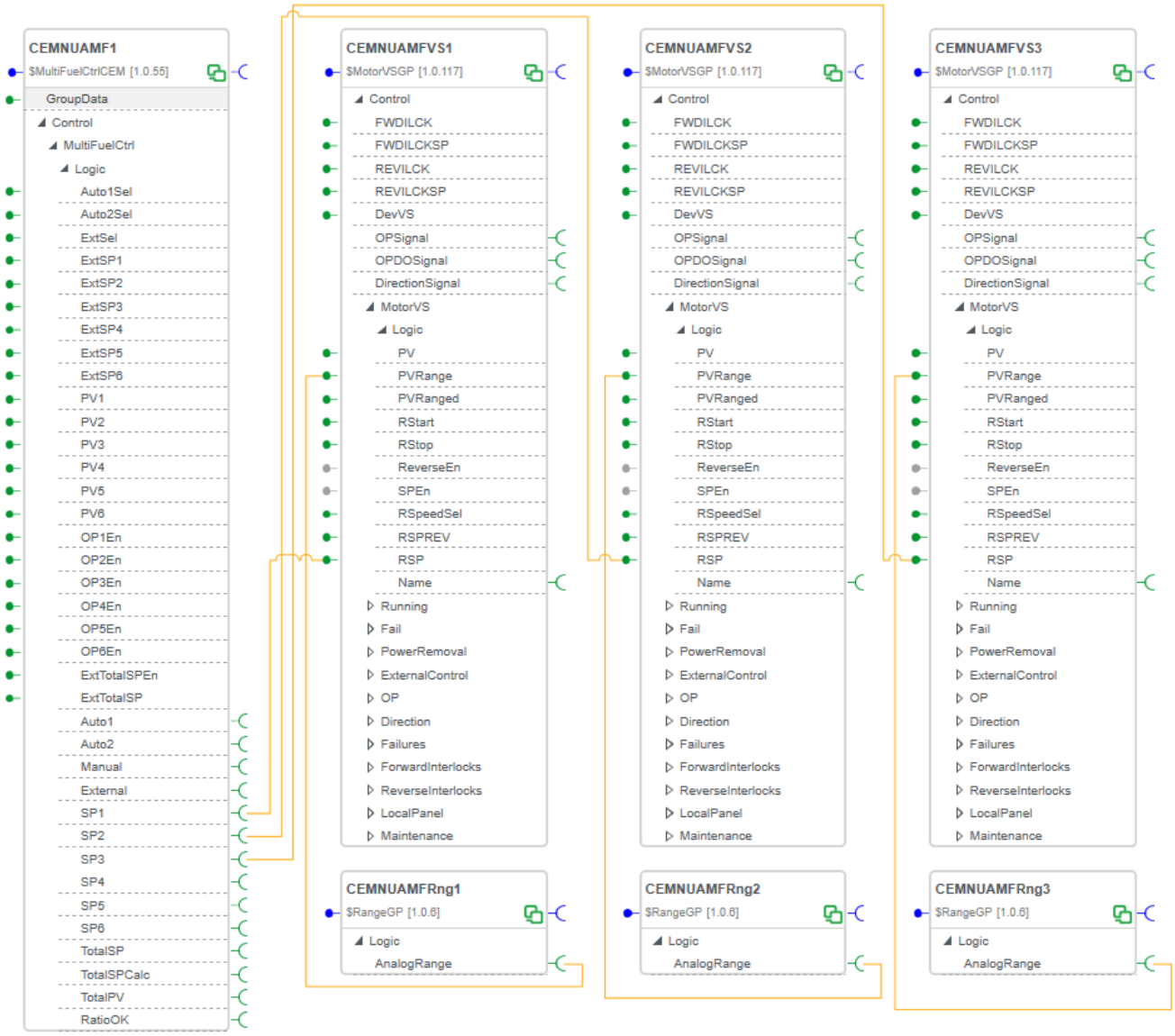
Overview

Examples in this manual are given 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.**

Use Case Example

The following figure describes the minimum configurations required for *MultiFuelCtrlCEM* function block with *GrpCtrl* and *MotorVSGP* function blocks. In this example, the *MultifuelCtrlCEM* function block controls three different types of fuel (represented by *MotorVSGP* block):



Index

A

C

F

G

H

O

P

T

Printed in:
Schneider Electric
35 rue Joseph Monier
92500 Rueil Malmaison - France
+ 33 (0) 1 41 29 70 00

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.

© 2021 – Schneider Electric. All rights reserved.

EIO0000004524.00