Process Power Manager 5

Library for 800xA

PMS 5.6.5 Restart & Reacceleration Library Manual

Version 5.6-5

Power and productivity for a better world™

Process Power Manager 5

Library for 800xA

PMS 5.6.5 Restart & Reacceleration Library Manual

Version 5.6-5

NOTICE

This document contains information about one or more ABB products and may include a description of or a reference to one or more standards that may be generally relevant to the ABB products. The presence of any such description of a standard or reference to a standard is not a representation that all of the ABB products referenced in this document support all of the features of the described or referenced standard. In order to determine the specific features supported by a particular ABB product, the reader should consult the product specifications for the particular ABB product.

ABB may have one or more patents or pending patent applications protecting the intellectual property in the ABB products described in this document.

The information in this document is subject to change without notice and should not be construed as a commitment by ABB. ABB assumes no responsibility for any errors that may appear in this document.

In no event shall ABB be liable for direct, indirect, special, incidental or consequential damages of any nature or kind arising from the use of this document, nor shall ABB be liable for incidental or consequential damages arising from use of any software or hardware described in this document.

This document and parts thereof must not be reproduced or copied without written permission from ABB, and the contents thereof must not be imparted to a third party nor used for any unauthorized purpose.

This product is designed to be connected to and to communicate information and data via a network interface. It is the reader's sole responsibility to provide and continuously ensure a secure connection between the product and the reader's network or any other network (as the case may be). The reader shall establish and maintain any appropriate measures (such as but not limited to the installation of firewalls, application of authentication measures, encryption of data, installation of anti-virus programs, etc) to protect the product, the network, its system and the interface against any kind of security breaches, unauthorized access, interference, intrusion, leakage and/or theft of data or information. ABB is not liable for damages and/or losses related to such security breaches, any unauthorized access, interference, intrusion, leakage and/or theft of data or information.

The software or hardware described in this document is furnished under a license and may be used, copied, or disclosed only in accordance with the terms of such license.
This product meets the requirements specified in EMC Directive 2004/108/EC and in Low Voltage Directive 2006/95/EC.

TRADEMARKS

All rights to copyrights, registered trademarks, and trademarks reside with their respective owners.

Release: September 2022
Document number: 3BNP100234-0396 B

Table of Contents

Contents

List of Figures

List of Tables

About This User Manual

General

This User Manual provides the configuration information for the Restart & Reacceleration control modules that are part of the PMS Library. The information in this manual is directed towards the project engineers. The user should be familiar with the Control IT for AC 800M/C and Operate IT environment.

Document Conventions

Microsoft Windows conventions are normally used for the standard presentation of material when entering text, key sequences, prompts, messages, menu items, screen elements, and so on.

The following conventions are used for the presentation of material:

- The words in names of screen elements (for example, the title in the title bar of a window, the label for a field in a dialog box) are initially capitalized.
- Capital letters are used for the name of a keyboard key if it is labelled on the keyboard. For example, press the ENTER key.
- Lowercase letters are used for the name of a keyboard key that is not labelled on the keyboard. For example, the space bar, comma key, and so on.
- Press CTRL+C indicates that you must hold down the CTRL key while pressing the C key (to copy a selected object in this case).
- The names of push and toggle buttons are boldfaced. For example, click **OK**.
- The names of menus and menu items are boldfaced. For example, the **File** menu.
- The following convention is used for menu operations: MenuName > MenuItem > CascadedMenuItem. For example: choose **File** > **New** > **Type**.
- The **Start** menu name always refers to the Start menu on the Windows Task Bar.
- System prompts/messages are shown in the Courier font, and user responses/input in boldfaced Courier font. For example, if you enter a value out of range, the following message is displayed:

  ```
  Entered value is not valid. The value must be 0 to 30.
  ```

- You may be instructed to enter the string TIC132 in a field. The string is shown as follows in the procedure:

  `TIC132`

- Variables are shown in italics:

  IOPar.Govmode8.value

- Faceplate tabs are boldfaced:

  AVR Mode handler

Warning, Caution, Information, and Tip Icons

This publication includes **Warning**, **Caution**, and **Information** if/where appropriate to point out safety related or other important information. It also includes **Tip** to point out useful hints to the reader. The corresponding symbols should be interpreted as follows:

Electrical warning icon indicates the presence of a hazard which could result in *electrical shock*.

Warning icon indicates the presence of a hazard which could result in *personal injury*.

Caution icon indicates important information or warning related to the concept discussed in the text. It might indicate the presence of a hazard which could result in *corruption of software or damage to equipment/property*.

Information icon alerts the reader to pertinent facts and conditions.

Tip icon indicates advice on, for example, how to design your project or how to use a certain function.

Although **Warning** hazards are related to personal injury, and **Caution** hazards are associated with equipment or property damage, it should be understood that operation of damaged equipment could, under certain operational conditions, result in degraded process performance leading to personal injury or death. Therefore, comply fully with all **Warning** and **Caution** notices.

Terminology

Table 1 lists terms used in this document and associated with the Process Power Manager 5. The reader should be familiar with these terms before proceeding further in this user manual.

Table 1 Terminology

Term	Description
AC 800M/C Connect	Name of the connection/integration between Control IT for AC 800M/C and Operate IT. The following software packages are included under this name: - Aspect System for AC 800M/C - Graphics Object Type Library for AC 800M/C
Aspect	A description of some properties of an Aspect Object. Some examples of aspects are name, circuit diagram, process display and control logic.
Aspect Object	A computer representation of a real object, such as a pump, a valve, an order or a virtual object, such as a service or an object type. An Aspect Object is described by its aspects and these aspects are organized in structures.
Aspect System for AC 800M/C	One part of the AC 800M/C Controller Integration product (the other part is the Graphics Type Library for AC 800M/C, see below). Gives access to the controllers AC 800M, AC 800C and Advant Controller 250, by mirroring all functions in the controllers and their I/O to Operate IT Process Portal from Control Builder Professional.
Connectivity Server	Provides the integration between the Operate IT system and a controller or a device capable of sourcing data.

Term	Description
Contingency	An electrical network within the plant consists of at least one load busbar to which loads, generation and the public grid can be connected. Several of these electrical networks can exist at the same time within the plant electrical network. Each combination is called a contingency.
Control Builder M	Name of the Control Builder M software products. Available in three versions: Control Builder Basic, Control Builder Standard and Control Builder Professional. These are fully integrated Windows 2000 Professional / Windows XP applications for efficient configuration and programming of the ABB controllers AC 800M, AC 800C and Advant Controller 250.
Control Builder Professional	The powerful programming version of the Control Builder M software.
Control Builder Project Explorer	The name of the project navigator in the Control Builder M software, which is used to navigate through, create or modify an automation project.
Control IT for AC 800M/C	The name of the collection of ABB hardware and software products for AC 800M/C.
Control Network	Product name of the ABB network between AC 800M/C controllers, tools and Operator workplaces.
Critical breaker	A breaker the position of which determines the electrical network configuration.
Display Element	A graphical element, which illustrates an object (motor, regulator etc.). In general, clicking on the element will show a faceplate for supervision and control of the object.
Electrical network	A combination of components such as load busbars, generators, transformers and cables connected electrically. A network contains at least one load busbar.
Faceplate	A configurable type of graphic interface normally used by operators for process supervision and control.
Graphics Type Library for AC 800M/C	One part of the AC 800M/C Controller Integration product (the other part is the Aspect System for AC 800M/C, see above). Graphic aspects such as display elements, faceplates and dialogs are available for use in Operate IT Workplace. The graphic aspects correspond to the types delivered in the Control Builder library.
HSI	Human System Interface.
Instance	An individual description of the type. Every instance has the characteristics defined by the type, but each instance has its own individual behaviour.
I/O	Input / Output signals.
IT	Information Technologies.
LVS	Low Voltage Switchgear.
MCC	Motor Control Centre.
MMS	Manufacturing Message Specification. Specifies the structure of messages used for industrial communication (manufacturing, process robotics, etc.). This is the application layer used within MAP (Manufacturing Automation Protocol), a specification for open communication based on the OSI model. MMS for AC 800M/C is a protocol used in ABB Control Network communication.

Term	Description
Object	Objects represent the combination of data and associated procedures (operations that can be applied to the data) are represented. Objects represent significant elements or functions in the process control/process automation domain. Combining these objects creates applications.
OPC	OLE for Process Control. The Control IT for AC 800M/C software contains an OPC Server for AC 800M/C.
Operate IT	The name of the collection of ABB products for daily operation and supervision of an automated process. These products provide an environment for different user categories, such as engineers, operators and maintenance personnel.
Plant Explorer	The name of the project or plant navigator in the Operate IT workplace for creating the Aspect Objects that are used for assembling the various components of the plant. Can also be used for browsing and searching the structures of the plant.
PMS	Power Management System.
PPM	Process Power Manager.
Process Panel	The name of the ABB product for local process monitoring and control. The key functionality is presentation of process information (numerical, text or graphical) on local operator or process panels including functional control keys.
Process Portal A	The name of the ABB product for process monitoring and control. The key functionality is presentation of process graphics, usage of faceplates, presentation of trends, and presentation of alarms.
Structure	A hierarchical tree organization of Aspect Objects that describes the dependencies between the real objects. An Aspect Object can exist in multiple structures, e.g. both in a functional structure and in a location structure.
System Extension	A plug-in software package, which provides the Operate IT system with extended functions and properties.
Type	A general description of a unit that defines the behaviour of an individual unit called Instance. See also Instance.
DCS	Distributed Control System.
ENMC	Electrical Network Monitoring & Control.

Related Documentation

Related documentation includes, but is not limited to, the table below. Other ABB 800xA documentation may also be relevant when configuring an 800xA Process Power Manager.

Table 2 Related Documentation

Document ID	Title
3BNP100234-0390	PMS Library 5.6-5 Release Notes
3BNP100234-0391	PMS Library 5.6-5 Circuit Breaker Control
3BNP100234-0392	PMS Library 5.6-5 Generator Control
3BNP100234-0393	PMS Library 5.6-5 Transformer Control
3BNP100234-0394	PMS Library 5.6-5 Loadshedding
3BNP100234-0395	PMS Library 5.6-5 Power Control
3BNP100234-0396	PMS Library 5.6-5 Restart & Reacceleration
3BNP100234-0397	PMS Library 5.6-5 Report Data Collector
3BNP100234-0398	PMS Library 5.6-5 Synchronization
3BSE037410	Administration and Security

Target Audience

This user manual is primarily intended for technical sales personnel, application, system engineers and maintenance personnel within ABB, external users and customers.

This user manual does not contain last-minute product information and updates which might affect functionality and/or performance. For information on last revisions, late changes and restrictions the user shall refer to *Release Notes.*

Some graphics have been carried over from previous loadshedding manuals, hence Windows frames, and aspect names, library versions, etc. might be different than in current 800xA and PMS version. (e.g. PG2 suffix in aspect names is no longer present, etc.)

Compatibility

For compatibility with previous versions of the product, refer to the *Release Notes 3BNP100234-0390*.

System Security

The supplier of automation systems, based on PMS libraries, is responsible for the system integrity and security. We strongly recommend that strict password policies are applied.

Reference is made to document *Administration and Security*. The whole manual must be carefully consulted, with special attention given to *Security Planning* and *Security Configurations* for guidelines regarding system security, user authentication, and password policies and setting up audit trails.

Section 1 General

When an industrial electrical network experiences severe voltage sags or short duration power interruptions, a certain number of motors (MV or LV) can be tripped by their electrical and mechanical protections, resulting in a plant upset. To minimize process disturbance or even avoid complete plant shutdown, motors used for critical services must be re-accelerated quickly, automatically and sequentially, assuming that they were running before occurrence of the voltage sag and were tripped during it.

The voltage-controlled time-staggered restart of electrical motor groups following a trip due to voltage sags is labeled as “reacceleration schedule”.

If voltage is recovered in a short time after a fault, the shafts of the motors that got tripped during the undervoltage stage might still rotate (depending on their driven load, duration of voltage sag, etc.) and they can be reaccelerated in a certain sequence, in order to minimize disturbance to the plants process flow.

Not all the motors are part of the reacceleration sequence. The need for automatic reacceleration is specified by process designers using the following pattern:

- necessary – motor driven equipment required to keep the unit operating without equipment damage and without safety equipment operating, but not necessary on product specification
- desirable – additional motor driven equipment required to keep units products on specification.
- unnecessary – operator restart is sufficient without adverse effect on unit or product specification.

Only the necessary and desirable motors should be part of the reacceleration sequence.

Section 2 Function

The reacceleration function is an automatic function which does not require operator intervention in order to operate. For a better understanding of its operation, the reacceleration function can be divided into 3 states/sequences:

- **idle** state, where it normally operates.
- **arming** state, that recognizes all inputs which would cause an initiation of the reacceleration sequence
- **operational** state, which covers the actual reconnection of the loads

2.1 Idle state

The idle state represents the most usual state for the reacceleration function, as it is corresponding to a normal voltage level within the electrical network. In idle state the function monitor:

- undervoltage stage activation
- running status of the motors

2.2 Arming state

The arming state starts from detection of an undervoltage stage (that is expected to determine the loads tripped) and it lasts until the voltage has recovered above a certain limit or cancelled. From that point onward, the function could enter into operational state or turn back to idle state.

Detection of an undervoltage stage can happen while the function is in idle state or even when it is in operational state, if a 2nd undervoltage stage occurs during operational state due to motors' reacceleration. In arming state the function monitors:

- undervoltage stage activation
- voltage recovery stage activation
- running status of the motors for trip indication

If voltage has recovered within the allowed voltage recovery time, then arming state is over and the function enters the operational state. Otherwise, reacceleration sequence is cancelled and function enters idle state. An overview of the functional flow from detection of undervoltage stage and voltage recovery to transition between the three states is shown in Figure 1.

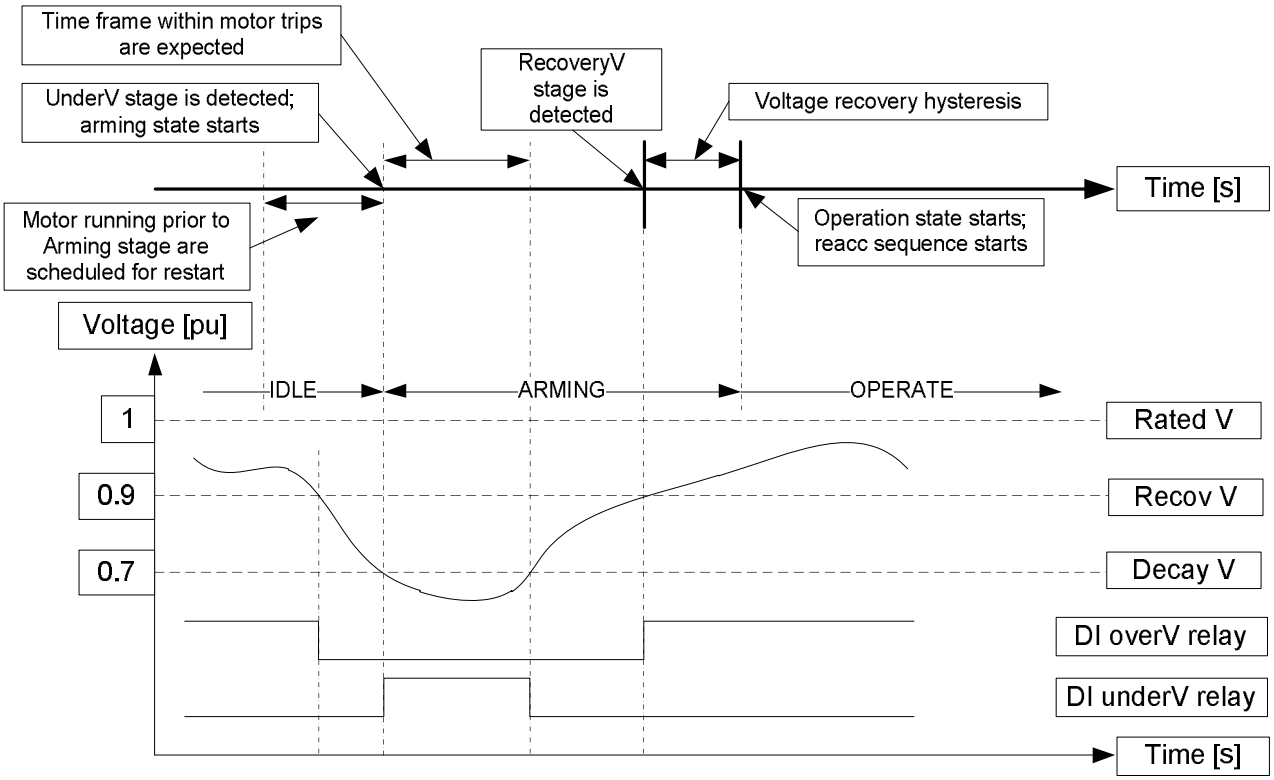


Figure 1 Overview of function flow

2.3 Operational state

Operational state fulfills the task of reconnecting the motors which have tripped during arming state and were running in idle state. As the operational state of the function is activated, the function will compute a priority table for the respective busbar(s). This table represents the reconnection sequence of the motors, according to its priorities. A motor is part of this table if:

- has a reacceleration priority
- not manual or system inhibited
- motor was running before arming state

The reconnection sequence is based solely on the priority number of each motor. Therefore, one reconnection step implies reconnection of one reacceleration group, which is given by all the motors with same reacceleration priority. Based on the reacceleration table, the function will address each priority from the lowest to the highest one by one. Additional, the operation state monitor the following:

- undervoltage stage activation
- voltage recovery stage activation
- reconnected\disconnected load that was scheduled in the arming stage

Once all the loads are reconnected, has failed to reconnect or timed out, the sequence is complete. The function returns to idle state and the historical data is transferred to a report.

2.4 System constraints

The restart and reacceleration function have several system constraints that should be considered at the beginning of the detailed engineering. The function is limited to the following constraints:

- coordination of 2 single busbars in the substation type
- 50 loads on each single busbar
- 20 configurable priority groups
- the report function can handle 4 voltage sags\recoveries
- non-coordinated outside the substation type

2.4.1 Non-coordinated reacceleration function

The function is implemented as a non-coordinated approach which implies that reconnection sequence is restricted to the substation configuration with 2 single busbars as illustrated in Figure 3. The substation type only looks upon its own voltage level and priority assignments and does not take other substations into consideration. This could lead to an overload state of certain components of the electrical network.

The restart and reacceleration functionality does not pay any respect to availability of active and reactive power within an electrical subnetwork when it decides the reconnection of a priority group.

Section 3 Control Module

The *pmsReAccSub* control module is included in the pmsReAccelLib library. A block presentation of the control module is shown in Figure 2.

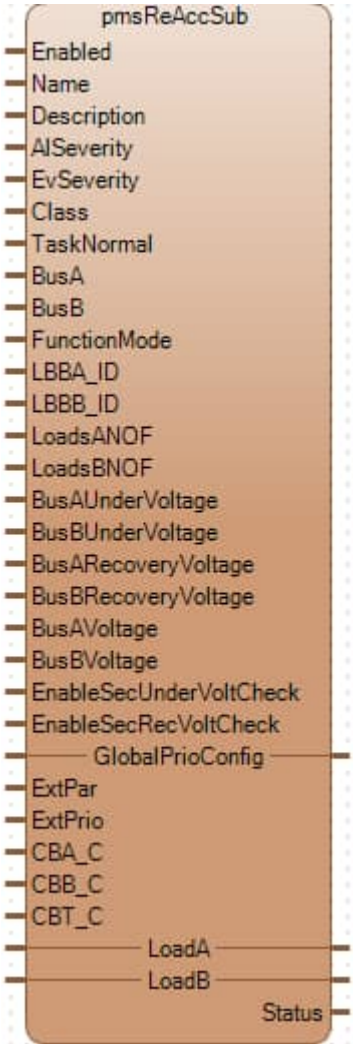


Figure 2 Control module pmsReaccSub

Table 3 Control module parameters

Parameter	Parameter Direction	Significance
Enable	Input	Enabling of the function. If TRUE the control module is executed.
Name	Input	Name of the object. With the name upload tool of the process portal, this property will be used for identifying the object in the 800xA system. The name will be used for grouping the alarms and events and identifying the faceplate of an object. The name must be unique in the project.
Description	Input	Description of the object.

Parameter	Parameter Direction	Significance
		With the name upload tool of the process portal, this property will be used for identifying the object in the 800xA system.
AlSeverity	Input	Alarm severity. The alarm severity can be used for filtering the alarm list. Range 1-1000.
EvSeverity	Input	Event severity. The event severity can be used for filtering the event list. Range 1-1000.
Class	Input	Alarm and Event Class. This parameter can be used for grouping objects to appear in separate alarm/event lists. Range 1-9999.
TaskNormal	Input	Execution cycle for HSI logic
BusA	Input	Busbar text. Default set to Busbar A and displayed in the Main faceplate element.
BusB	Input	Busbar text. Default set to Busbar B and displayed in the Main faceplate element.
FunctionMode	Input	Text configuration of the faceplate for either restart or reacceleration.
LBBA_ID	Input	Index of the load busbar to which the A side of the substation is connected to (for future use).
LBBB_ID	Input	Index of the load busbar to which the B side of the substation is connected to (for future use).
LoadsANOF	Input	Number of loads connected to the A side of the substation (max 50).
LoadsBNOF	Input	Number of loads connected to the B side of the substation (max 50).
BusAUnderVoltage	Input	Undervoltage indication on busbar A as an extended DI datatype with provision for external timestamp of signal (IO from undervoltage relay).
BusBUnderVoltage	Input	Undervoltage indication on busbar B as an extended DI datatype with provision for external timestamp of signal (IO from undervoltage relay).
BusARecoveryVoltage	Input	Recovery voltage indication on busbar A as an extended DI datatype with provision for external timestamp of signal (IO from overvoltage relay).
BusBRecoveryVoltage	Input	Recovery voltage indication on busbar B as an extended DI datatype with provision for external timestamp of signal (IO from overvoltage relay).
BusAVoltage	Input	Voltage measurement on bus A.
BusBVoltage	Input	Voltage measurement on bus B.
EnableSecUnderVoltCheck	Input	Enable undervoltage check if undervoltage DI is in error.
EnableSecRecVoltCheck	Input	Enable recoveryvoltage check if overvoltage DI is in error.
GlobalPrioConfig	Input	Structured parameter for global configuration of the function and priority configuration.
ExtPar	Input	External parameters for application interaction. With this parameter it is possible to disable alarm and events and block the function from executing.

Parameter	Parameter Direction	Significance
ExtPrio	Input	Structured parameter for external priority commands (for future use).
CBA_C	Input	Checked closed position of the A incomer of the substation.
CBB_C	Input	Checked closed position of the B incomer of the substation.
CBT_C	Input	Checked closed position of the tie breaker of the substation.
LoadA	Input & Output	Structured parameter which contains all the data for load objects connected to panel A of the substation.
LoadB	Input & Output	Structured parameter which contains all the data for load objects connected to panel B of the substation.
Status	Output	Structured parameter for status indications.

The control module pmsReAccSub is suitable for handling the type of substation configurations, as shown in Figure 3.

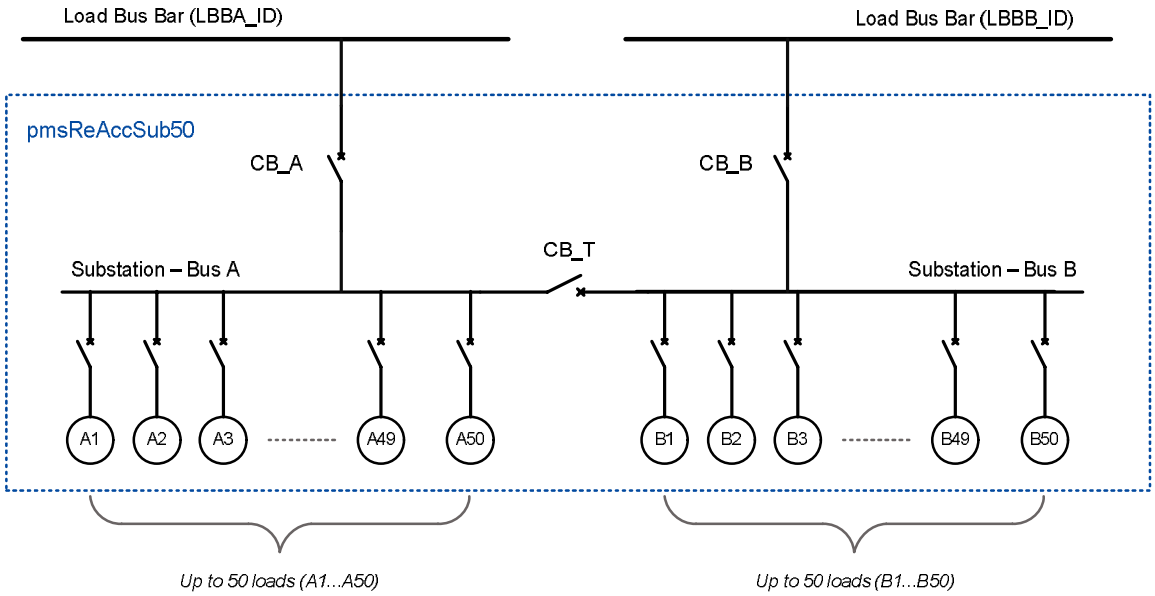


Figure 3. Substation configuration for pmsReAccSub

3.1 Detailed Engineering

The restart & reacceleration function in the PMS Library is based on the control module type pmsReAccSub. The control module monitors voltage on each bus section and running loads in a substation configuration. The following features are supported:

3.1.1 Voltage monitoring

The pmsReAccSub control module can monitor voltage on both bus sections in a substation configuration. The following features are supported:

- undervoltage monitoring from undervoltage relay.
- recovery voltage monitoring from overvoltage relay.
- monitor analogue voltage value

3.1.2 Load monitoring

Up to 50 loads on each bus section can be interfaced with one pmsReAccSub control module. The following interactions are provided to the operator:

- priority configuration
- operator Inhibit
- individual load configuration

3.1.3 Blocking

There are three ways to block the restart and reacceleration function:

- operator blocking
- external blocking
- internal condition

3.1.4 Configuration

Configuration may be done on three levels, individual load configuration supersedes the other:

- global configuration
- priority configuration
- load configuration

3.1.5 Operator interface

The operator interface is provided through a dedicated faceplate for the *pmsReaccSub* module.

3.1.6 History reporting

When reacceleration function has completed or cancelled its sequence, the pmsReAccSub control module can generate a report with historical data about the sequence. The following features are supported:

- up to 4 voltage sags and\or recoveries.
- failed to restart
- inhibited loads
- successful restart
- loads not running

3.1.7 Alarm & Event handling

Critical operational changes that affect the reacceleration sequence are recorded and presented in Alarm and Event lists.

3.1.8 Voltage monitoring

The control module pmsReAccSub provides input for detection of voltage stages both for voltage sags or deeper disruptions and normalized voltage levels on both bus sections of the substation.

3.1.8.1 Undervoltage

To detect an undervoltage stage the pmsReAccSub provides two extended BoolIO data types of *pmsReAccBoolIOExt* that provide the option to external timestamp event via DI830 digital input card. The data type *pmsReAccBoolIOExt* allows monitoring of the I/O channel status, which is important for the signal error handling inside the control module logic. In cases where the measurement is not BoolIO, logic outside the control module must be made for the signal quality status.

Table 4 Parameters configuration for undervoltage

Parameter configuration	Description
BusAUnderVoltage.Value	Detection of undervoltage on bus A.
BusAUnderVoltage.ExtTimeStamp	External timestamp configuration.
BusAUnderVoltage.SignalID	Signal mapping to IO module.
BusBUnderVoltage.Value	Detection of undervoltage on bus B.
BusBUnderVoltage.ExtTimeStamp	External timestamp configuration.
BusBUnderVoltage.SignalID	Signal mapping to IO module.

In all function states the activation of undervoltage stage will always alter the function state from idle to arming or from operational to arming. When the undervoltage stage is activated the voltage below decay level LED transitions from grey to red, the no. of voltage decay count increases, and an alarm is raised and the active stage changes to arming. See Figure 4 for details.

Figure 4 faceplate with busbar status of undervoltage state

The function can be configured for maximum number of voltage sags that could take place before cancelling the sequence. This can be accessed via parameter GlobalPrioConfig.Global.MaxNrVoltageDecays or from Max no. of voltage decay setting in **Priority settings** tab in the extended faceplate.

3.1.8.2 Recovery voltage

To detect a normalized voltage level the pmsReAccSub provides two extended BoolIO data types of *pmsReAccBoolIOExt* that provide the option to external timestamp event via DI830 digital input card. The data type *pmsReAccBoolIOExt* allows monitoring of the I/O channel status, which is important for the signal error handling inside the control module logic. In cases where the measurement is not BoolIO, logic outside the control module must be made for the signal quality status.

Table 5 Parameter configuration for recovery voltage

Parameter configuration	Description
BusARecoveryVoltage.Value	Detection of recovery voltage or normalized voltage on bus A.
BusARecoveryVoltage.ExtTimeStamp	External timestamp configuration.
BusARecoveryVoltage.SignalID	Signal mapping to IO module.
BusBRecoveryVoltage.Value	Detection of recovery voltage or normalized voltage on bus B.
BusBRecoveryVoltage.ExtTimeStamp	External timestamp configuration.
BusBRecoveryVoltage.SignalID	Signal mapping to IO module.

In both arming and operational states of the function an activation of the recovery voltage stage will always alter the function state from arming to operational and a loss of recovery voltage signals will alter the function state from operational to arming. When recovery voltage stage is activated the voltage above recovery level LED transitions from grey to green and the active stage change to operational.

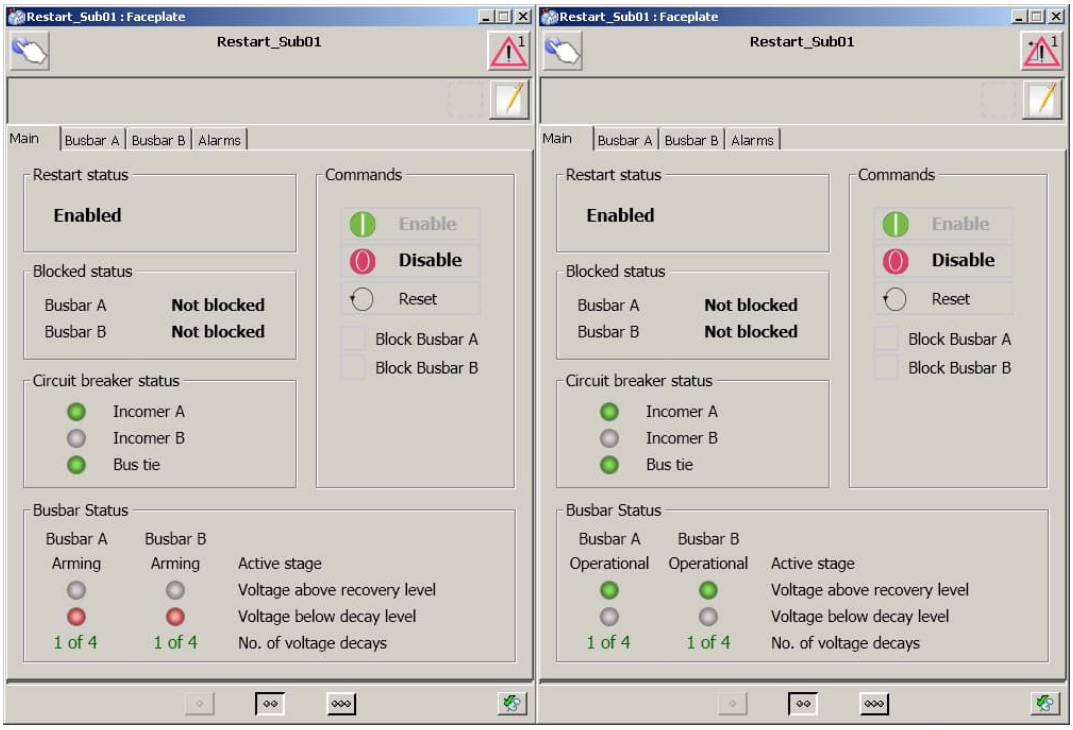


Figure 5 Faceplate with busbar status of voltage recovery state

The function can be configured for maximum allowed voltage recovery time. This can be accessed via parameter *GlobalPrioConfig.Global.AllowedVoltRecoveryTime* or from Allowed voltage recovery time setting from **Priority settings** tab in the extended faceplate.

3.1.8.3 Analogue voltage measurement

Two parameters of data type RealIO are provided for analogue measurement. The data type RealIO allows monitoring of the I/O channel status, which is important for the signal error handling inside the control module logic. In cases where the measurement is not RealIO, logic outside the control module must be made for the signal quality status.

Both measurements are used as a backup solution for detection of both undervoltage and recovery voltage.

Table 6 Parameter configuration for voltage measurement

Parameter configuration	Description
BusAVoltage.Value	Analogue voltage measurement for bus A
BusBVoltage.Value	Analogue voltage measurement for bus B

If primary undervoltage detection parameter is faulty and *EnableSecUnderVoltCheck* is set to `True` the function considers an undervoltage stage to be active on the bus when the voltage measurement is less or equal to Voltage decay level setting. This setting can be configured via parameter *GlobalPrioConfig.Global.VoltageDecayLevel* or from Voltage decay level setting in **Priority settings** tab in the extended faceplate.

If primary recovery voltage detection parameter is faulty and *EnableSecRecVoltCheck* is set to `True` the function considers a recovery voltage stage to be active on the bus when the voltage measurement is higher or equal to Voltage recovery level setting. This setting can be configured via parameter *GlobalPrioConfig.Global.VoltageRecoveryLevel* or from Voltage recovery level setting in **Priority settings** tab in the extended faceplate.

3.1.8.4 Signal transition

In the event of a signal transition period where both undervoltage and recovery voltage signals are high, the module will act as following;

- Block of the function and cancelation of the sequence if *GlobalPrioConfig.Global.SignalTransitionT* is set to 0s(disable)
- For a *GlobalPrioConfig.Global.SignalTransitionT* of more than 0s; it is assumed that the latest signal are up-to-date and the “older” signal suppress (i.e. In Idle state where recovery voltage signal is high and for some cycles later also the undervoltage is high, the function will suppress the voltage recovery signal and enter into arming stage). The signal is only suppressed while both are high and SignalTransitionT has not elapsed. If both signals remain high after SignalTransitionT has elapsed, the function is blocked and the sequence cancelled. See Table 7 for prioritized signal.

Table 7 Prioritized voltage signal related to function stage

Function stage	Priority signal in transition period
Idle	Undervoltage
Arming	Recovery Voltage
Operational	Undervoltage

3.1.8.5 Built-in Error handling

The function has built-in signal error handling. Both undervoltage and recovery voltage parameters are of data types that allow monitoring of the I/O channel status, which is important for the signals handling. In a substation configuration when the tie breaker is closed, the healthy signal will be transferred to the bus with faulty measurement. If both are faulty and a backup solution with secondary undervoltcheck\recoveryvoltcheck is disabled, the function will go into blocking condition. For more details about blocking, see chapter 3.1.3. The project can specify the healthy condition signal from the **project constants** -> **pmsConstants**->**cStatusOK** and **cStatusMask** for masking of signal before comparison check with **cStatusOK**.

3.1.9 Load monitoring

The control module handles 50 loads connected to each half-panel of a substation configuration. The supported functions for each load are:

3.1.9.1 Running status monitoring

The pmsReAccSub module monitors the running feedback for determining which load was running prior to an undervoltage. The running feedback signal can be based on a simple “breaker closed feedback” or be based on operation conditions of the motor (speed, torque, current, etc.).

3.1.9.2 Individual configuration timers

Each load has provision for individual configuration within a priority group. For the ease of configuration each load can also be assigned to inherit the configuration settings for the priority group settings.

3.1.9.3 Operator interaction and status

The pmsReAccSub control module provides the following features for operator interface:

- definition of priority for a load
- inhibiting of close commands and thus preventing the load from being restarted.
- monitoring of load value and automatic inhibited loads

3.1.9.4 Reconnect, output command

If a load was connected to the busbar prior to an undervoltage, the load is scheduled for restart. When the operational stage executes the scheduled loads priority group, the control module shall activate a pulsed digital output for reconnecting the assumed tripped load.

3.1.9.5 Alarms and Events

Critical operational changes that affect the loads are recorded and presented in Alarm and Event lists.

3.1.9.6 Feedback monitoring

The structured *LoadPar* parameter of the control module pmsReAccSub must be configured to receive the running feedback signal that confirms that the load is connected to a load busbar and the circuit breaker is in remote point of control (i.e. PMS control) to restart the load. These parameters are structured data types and the components applicable are shown in Table 8.

Table 8 Parameter configuration for feedback monitoring

Parameter configuration	Description
Load.LoadPar.RunningStatus	Feedback signal for Load running
Load.LoadPar.POCLoc	Circuit breaker point of control

Both *LoadPar.RunningStatus* and *LoadPar.POCLoc* are of an extended BoolIO type – *pmsReAccBoolIOex* that provide the option to external timestamp event via DI830 digital input card. The data type allows monitoring of the I/O channel status, which is important for the signal handling inside the control module logic. If one of the feedback signals is faulty the load is automatically inhibited and will not be restarted.

The project will have to choose functionality behind the signal Load.LoadPar.RunningStatus, as follows:

- Based on a simple breaker closed feedback signal.
- Based on operation conditions of the motor (speed, torque, current, etc.). The signal will then contain the information that motor reached a stable operation state and its transient period has ended.

The functionality behind the running status will affect how the group delay timers are configured, see chapter 3.1.15.2 for more details.

3.1.9.7 Individual configuration timers

Each load can be configured individually whilst belonging to a priority group. Configuration settings can be completed from the faceplate extended view tab **Load settings**. A user with *Configure* permission can configure all settings except for priority which requires *Tune* permission.

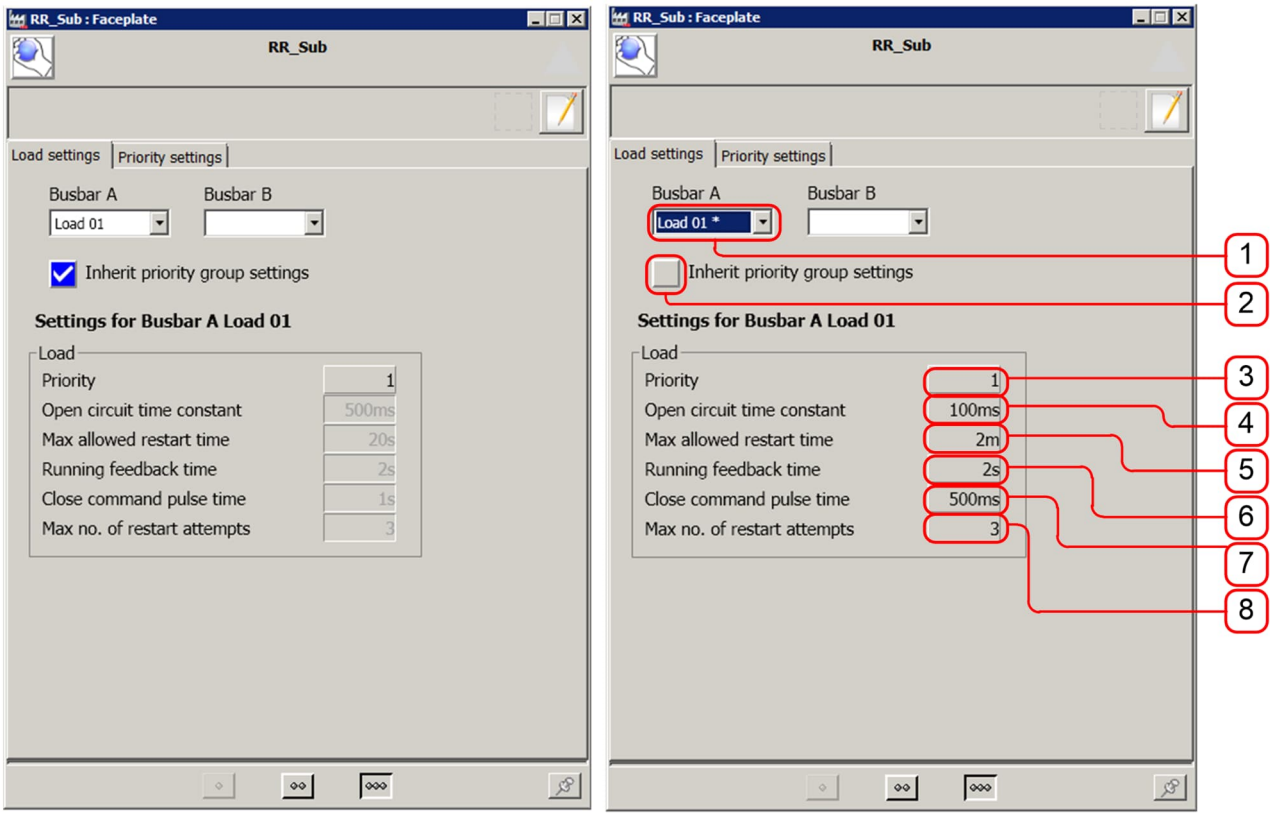


Figure 6 Faceplate configuration of Load settings

Table 9 faceplate configuration of load configuration timers

Field	Indication	Item	Description
1	Load selection	Drop down menu	Selection of load configuration
2	Inherited settings	Check box	Checkbox to inherit settings for selected priority group
3	Priority	Input field	Configuration of the priority for the restarting function
4	Open circuit time constant	Input field	Time from detection of undervoltage to reconnection of motor to electrical network is restricted.
5	Max allowed restart time	Input field	Time from undervoltage to reacceleration should succeed
6	Running feedback time	Input field	Time from close command is given to running feedback is expected
7	Close command pulse time	Input field	Pulse length for close command
8	Max no. of restart attempts	Input field	Max number of restart attempts

It is also possible to configure the load settings from an application code via the input parameter *Load.LoadPar* listed in table below. It is only possible to utilize application logic to set the load settings when *GlobalPrioConfig.Global.OperatorSettable* is set to False.

Table 10 Parameter configuration of load configuration timers

Parameter configuration	Description
Load.LoadParXx.OpenCircuit_time	Time from detection of undervoltage to reconnection of motor to electrical network is restricted.
Load.LoadParXx.AllowedReaccel_time	Time from undervoltageto reacceleration should succeed
Load.LoadParXx.CloseCMD_time	Pulse length for close command
Load.LoadParXx.RunningFb_timeout	Time from close command is given to running feedback is expected
Load.LoadParXx.MaxNrOfAttempts	Max number of restart attempts
Load.LoadParXx.Grouped	Individual settings to be decided by group settings

3.1.10 Operator commands

3.1.10.1 Priority definitions

The priority definition can be configured via the **Busbar A and\or B** faceplate tab, as shown in Figure 7. A user with *Tune* permission can configure the property.

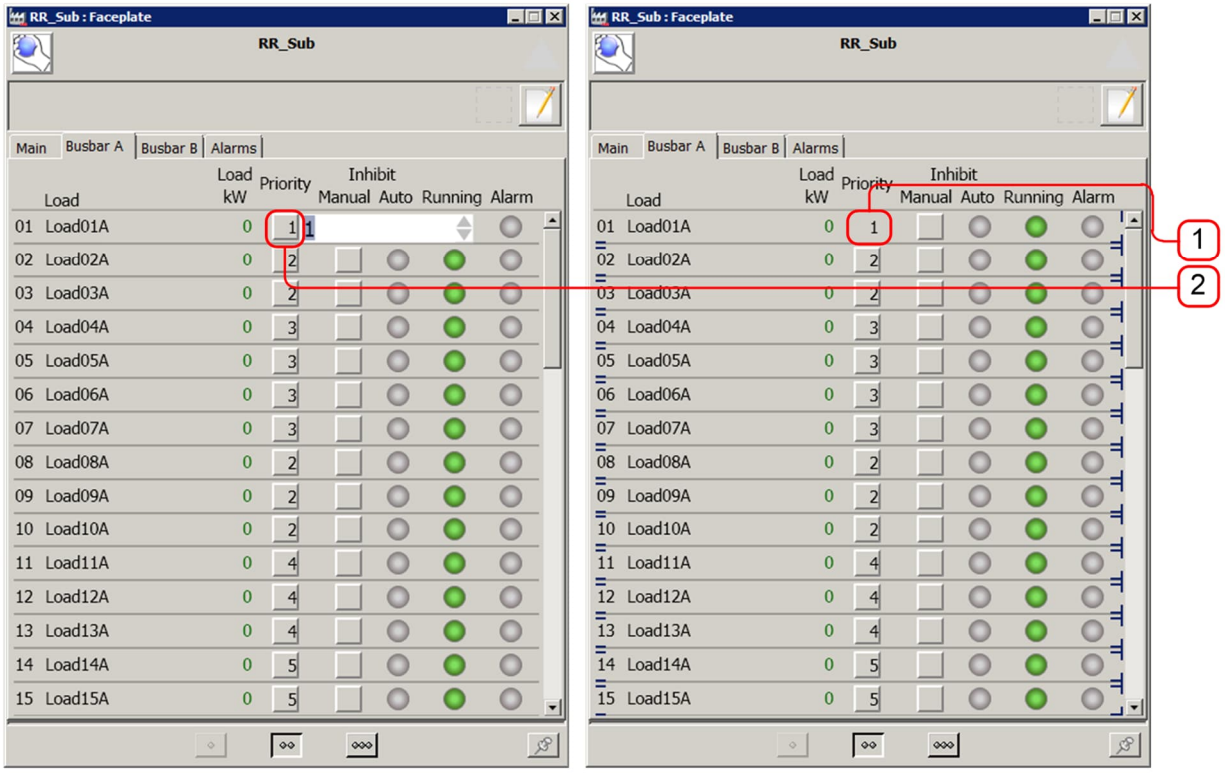


Figure 7 faceplate configuration of priority configuration

Table 11 Faceplate configuration for priority definition

Field	Indication	Item	Description
1	Priority	Input field	Configuration of the priority for the restarting function
2	External priority	Numerical field	Externally set priority from application code.

It is also possible to configure the priorities from an application code via the input parameter *Load.LoadPar*:

- When the input parameter *Load.LoadPar.ExtPrioSel* is true, the priority will follow the values from the input parameters *Load.LoadPar.Prio* respectively and the faceplate interaction will be restricted, as shown in Figure 7.
- After the parameter *Load.LoadPar.ExtPrioSel* returns to false, the priorities will remain to their current values and the faceplate interaction will be enabled.

3.1.10.2 Inhibit restart commands

The load can be manually inhibited from a user, via Manual inhibit checkbox, through faceplate interaction from **Busbar tab**. If signal quality of running feedback, close command or circuit breaker point of control is in error, the load will automatically be inhibited.

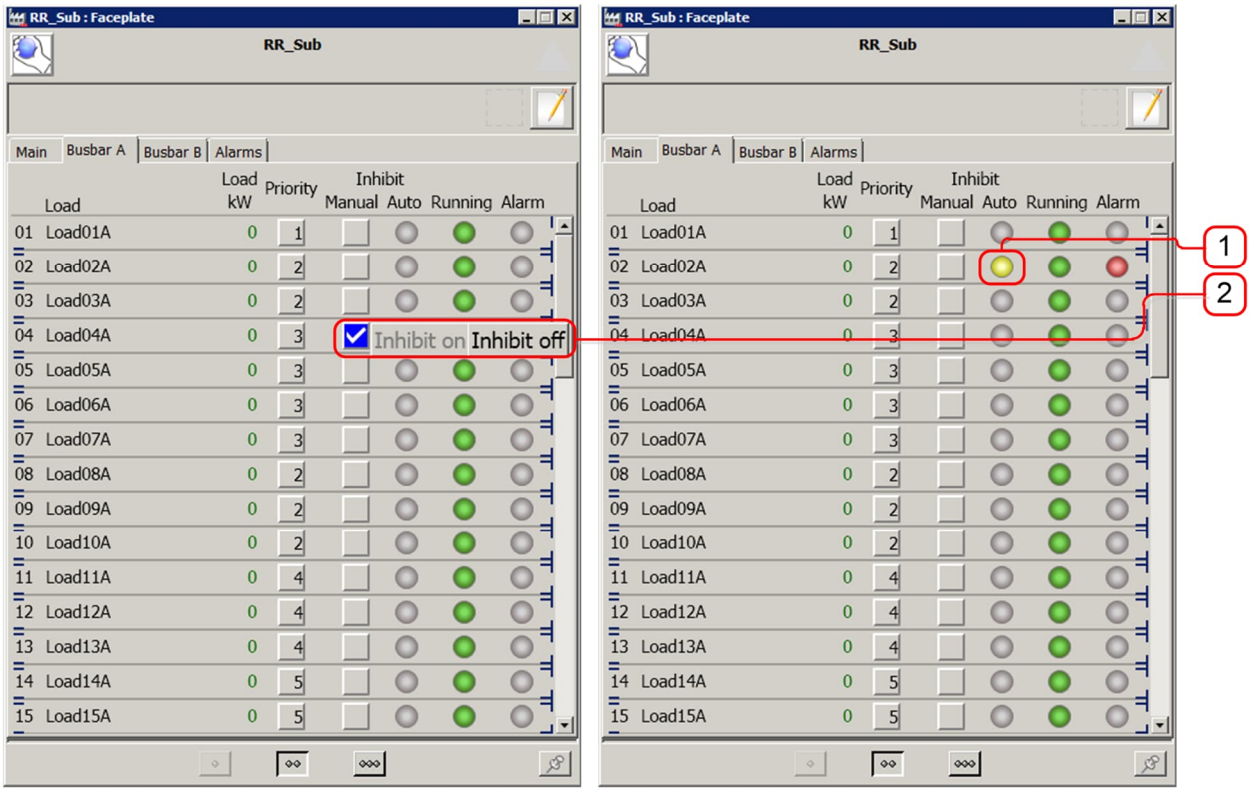


Figure 8 Faceplate configuration for inhibit interaction

Table 12 Faceplate configuration for inhibit interaction

Field	Indication	Item	Description
1	Manual Inhibit	Check box	Inhibit load from restarting sequence.
2	Automatic Inhibit	Indication	Automatic inhibit of load from restarting sequence; 1) I/O Error (RunningStatus or Close_CMD or POCLoc status is in error) 2) External inhibit

A user with *Tune* permission can set this property.

It is also possible to automatically inhibit the load via external application code by setting input parameter *Load.LoadPar.ExternalInhibit* to True.

3.1.11 Reconnect, output command

Based on the reacceleration table, the function will address each priority from the lowest to the highest, one by one and the following conditions will be checked before the load is issued a reconnect command:

- time elapsed from disconnection is greater than the load's open circuit time constant
- time elapsed from disconnection is lower than max allowed reacceleration time
- load is not inhibited (external, operator or I/O error) from participation in the reacceleration sequence. (i.e. the stand-by unit has started in the meantime, DCS process flow restriction)
- number of max reconnection attempts are not reached
- circuit breaker is in Remote point of control (the function will not pay any respect to mode of the breaker within ENMC system: manual or automatic)

If the only condition not fulfilled is the load's open circuit time constant, then the function will wait for that time to elapse and proceed with close command, if the rest are met. If one of the last four conditions is not fulfilled, the load will be removed from reacceleration sequence but the overall reacceleration function will continue. Hence that condition related to open circuit time constant may delay the command for other reacceleration groups in cases where time constants of motors are different. This is due to the delay timer starts after the last executed load in a priority group has completed or failed.

If all the above conditions are fulfilled, the reacceleration function will give a close command to all load breakers within the current reacceleration group. The function can be configured to perform a certain number of close command attempts if the motor running feedback is not received after the first command. At the moment when the first close command is given for the current reacceleration group, a timer is started to:

- monitor the time from close command is given to running status is received (*Running feedback time)*
- monitor the time interval between different acceleration priorities (*Reacceleration groups' timers*)

3.1.11.1 Load Status

Each load has several status indications that can be helpful for a project to build application logic. The parameters can be found in table below.

Table 13 Parameter for load status

Parameter configuration	Description
Load.LoadParXx.Disabled	Load status, load is disabled for restart
Load.LoadParXx.Signal_Err	Load signal error indication
Load.LoadParXx.FailedReaccel	Load status, failed to restart
Load.LoadParXx.InhibitStatus	Load status, Automatic\Operator inhibit active

3.1.11.2 Alarms and Events

For more details see chapter 3.1.18 for alarms and chapter 3.1.19 for events.

3.1.12 Alarm & Event disabling

The control module pmsReAccSub provides the possibility to disable alarms & events from the configuration parameter *ExtPar.* The alarm and events that can be disabled are described in table below.

Table 14 Parameter for external disabling of alarms & events

Parameter configuration	Description
ExtPar.DisUnderVoltAlarm	Disables undervoltage alarm
ExtPar.DisRepVoltSagAlarm	Disables repeating voltage sag alarm
ExtPar.DisIOErrorAlarm	Disables IO error alarm
ExtPar.DisSpuriousAlarm	Disables spurious condition alarm
ExtPar.DisConfigError	Disables configuration error alarm
ExtPar.DisActiveStageEvent	Disables active function stage event
ExtPar.DisEnableEvent	Disables enable\disable event
ExtPar.DisBlockEvent	Disables block event
ExtPar.DisAbortRestartEvent	Disables abort restart sequence event
ExtPar.DisResetEvent	Disables reset event
ExtPar.DisReleasePrioEvent	Disables released priority event
ExtPar.DisLoadEvOpInhibit	Disables load manual inhibit event
ExtPar.DisLoadEvAutoInhibit	Disables load auto inhibit event
ExtPar.DisLoadEvFailedRestart	Disables load failed restart event
ExtPar.DisLoadEvPrioChange	Disables load priority change event
ExtPar.DisLoadEvRunningFb	Disables load running feedback event
ExtPar.DisLoadEvRunningFbLost	Disables load running feedback lost event
ExtPar.DisLoadEvCMD	Disables load close command event
ExtPar.DisLoadIOErrorAlarm	Disables load IO error alarm
ExtPar.DisLoadConfigErrorAlarm	Disables load configuration error alarm

Disabling of events *ExtPar.DisLoadEvCMD*, *ExtPar.DisLoadEvRunningFb* and *ExtPar.DisLoadEvRunningFbLost* have will more than half the execution time of the module and should be considered if fast execution time is required.

3.1.13 Blocking

The restart and reacceleration function is subject to blocking condition and there are three blocking conditions available;

- manual block
- automatic block
- external block

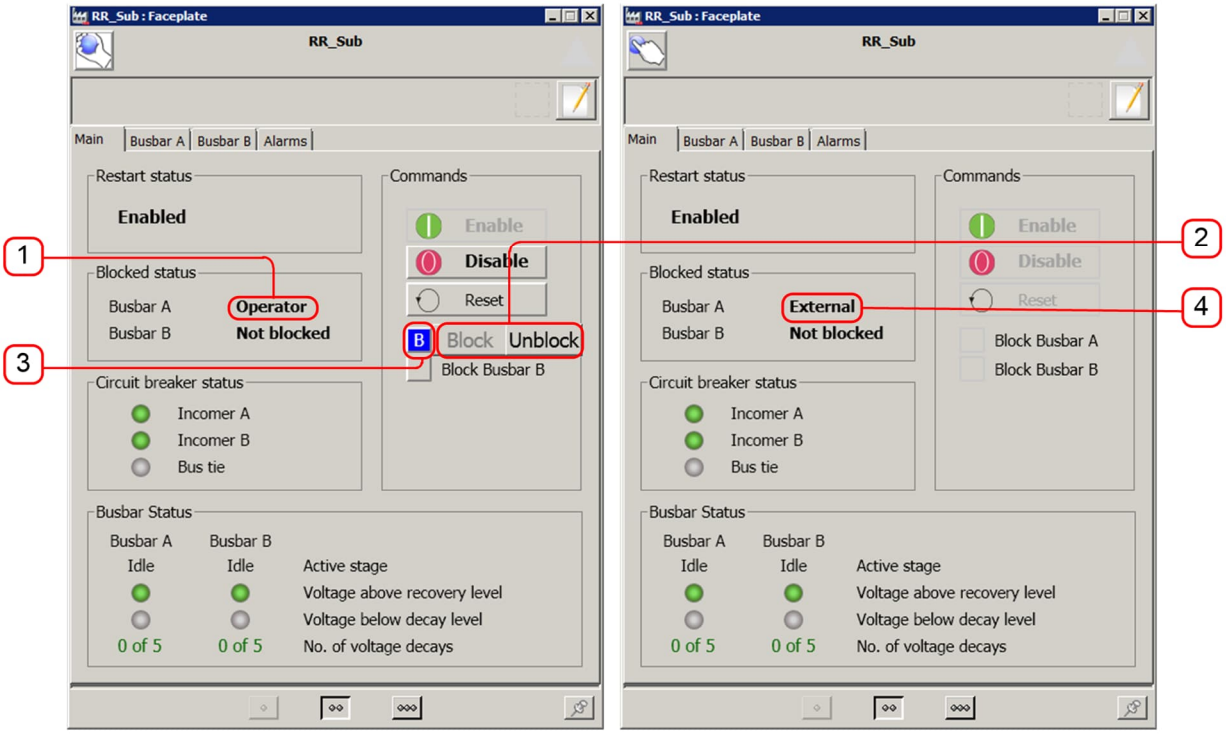


Figure 9 faceplate configuration of block

Table 15 Main faceplate block interaction

Field	Indication	Item	Description
1	Blocking status	Indication field	Busbar blocked status – blocked by operator.
2	Block command	Command button	Command option to Block or Unblock.
3	Block indication	Indication field	Indication of block operator action.
4	Block indication	Indication field	Busbar blocked status – blocked by external logic.

Manual block operation is available to users with *Operate* permission.

It is also possible to block the function from executing via external application code by setting input parameter *ExtPar.ExBlock_BusbarX* to True. The external block text presented in the main faceplate is configured via parameter *ExtPar.ExBlock_BusbarXEvent*.

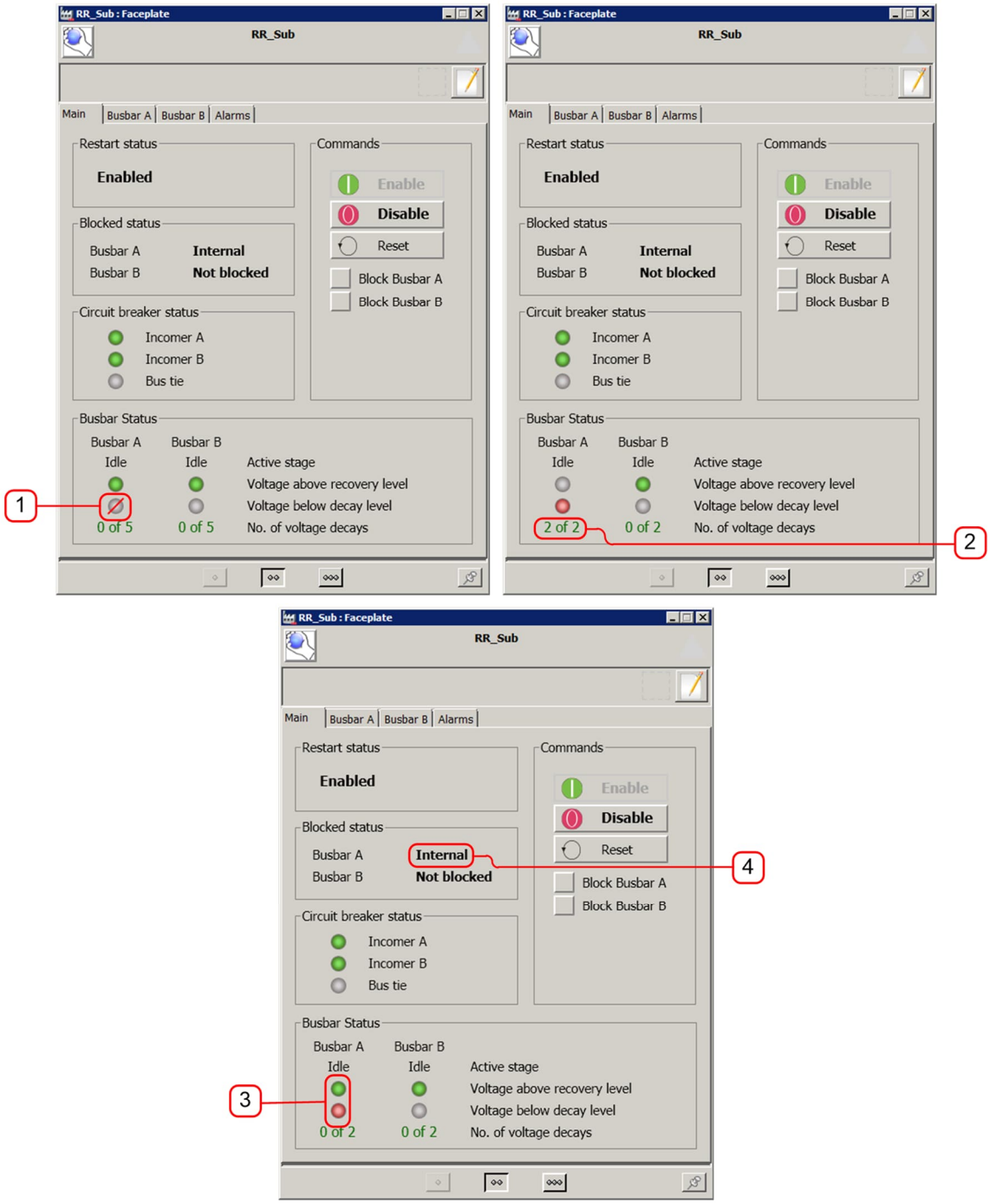


Figure 10 Faceplate indication of automatic block

Table 16 Main faceplate automatic block interaction

Field	Indication	Item	Description
1	I/O error	Indication field	Internal block caused by I\O error
2	No. of voltage decays	Indication field	Internal block caused by Maximum voltage decays
3	Spurious condition	Indication field	Internal block caused by spurious condition
4	Block indication	Indication field	Busbar blocked status – blocked by internal logic; 1) I/O Error (on both busbar if interconnected) 2) Spurious condition (on both busbar if interconnected) 3) No. of voltage decays >= Max no. of voltage decays

3.1.14 External priority

The *ExtPrio* parameter is set aside for future use.

3.1.15 Configuration

3.1.15.1 Global configuration

The control module pmsReAccSub provides two options when configuring the global settings. All settings are configurable via parameter connection *GlobalPrioConfig*. Most settings which could be subject to tuning are available via the faceplate Priority settings.

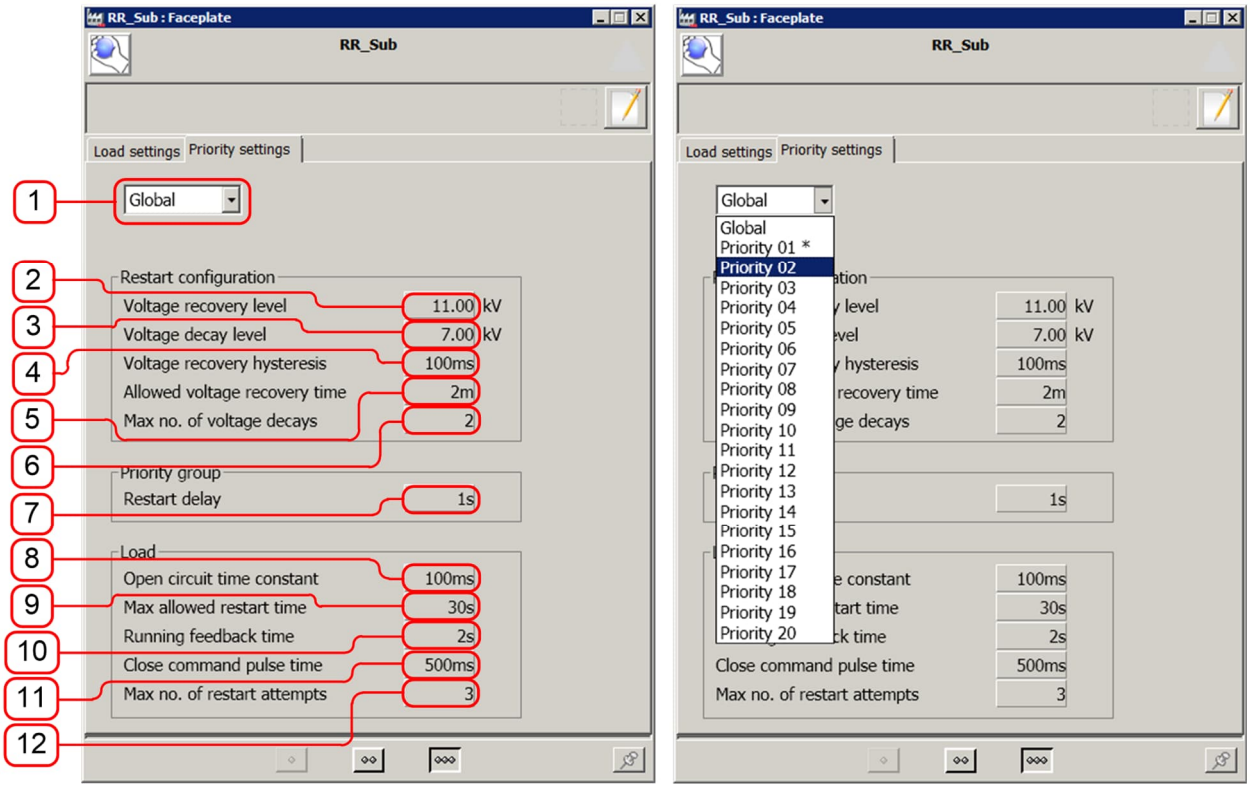


Figure 11 Faceplate configuration of Global settings

An asterisk (*) indication behind a priority group in dropdown menu, indicates that the priority does not inherit the global settings.

Table 17 Faceplate configuration of Global settings

Field	Indication	Item	Description
1	Configuration selection	Dropdown menu	Selection of Global configuration or individual priority.configuration.
2	Voltage recovery level	Input field	Comparator for analogue voltage measurement to determine sufficient voltage recovery.
3	Voltage decay level	Input field	Comparator for analogue voltage measurement to determine voltage decay level is reached.
4	Voltage recovery hysteresis	Input field	The amount of time voltage should keep above voltage recovery level before actual voltage recovery should be considered and continuation of the sequence can commence.
5	Allowed voltage recovery time	Input field	The amount of time that should be considered before the sequence is cancelled and reverted back to idle due to failed voltage recovery.

6	Max no. of voltage decays	Input field	The max number of voltage decays allowed from initial decay (also counted) to the sequence can be completed. After a completed sequence, the number of voltage decays is reset back to 0.
7	Restart delay	Input field	Restart delay is the delay that should be considered from last running feedback signal is received to the next priority group can be started.
8	Open circuit time constant	Input field	Time from detection of undervoltage to reconnection of motor to electrical network is restricted.
9	Max allowed restart time	Input field	Time from undervoltageto reacceleration should succeed.
10	Running feedback time	Input field	Time from close command is given to running feedback is expected.
11	Close command pulse time	Input field	Pulse length for close command.
12	Max no. of restart attempts	Input field	Max number of restart attempts.

The Global settings are only configurable from the faceplate if *GlobalPrioConfig.Global.OperatorSettable* is set to True and for users with Configure permission.

Not all configuration settings are available from the faceplate, but are only configurable from application logic. The *GlobalPrioConfig.Global* has the configuration parameters detailed in the table below.

Table 18 Parameter configuration of GlobalPrioConfigGlobal

Parameter configuration	Description
EnableReporting	Enable reporting feature, see chapter 3.1.17 for more details.
EnableOpReporting	For future use.
BusbarConfiguration	The BusbarConfiguration hides busbar in main faceplate according to this setting.
CBConfigured	The CBConfigured hides circuit breakers in the main faceplate according to this setting.
MaxPriority	Specify max priority group utilized in the project.
HideManualInhibit	This setting hides manual inhibit checkbox in faceplate element Busbar A and B.
HideLoadValue	This setting hides manual inhibit checkbox in faceplate element Busbar A and B.
OperatorSettable	This setting controls if settings should be set from application logic or faceplate.
VoltageDecayLevel	Comparator for analogue voltage measurement to determine voltage decay level is reached.
VoltageRecoveryLevel	Comparator for analogue voltage measurement to determine sufficient voltage recovery.

Parameter configuration	Description
VoltageRecoveryHyst	The amount of time voltage should keep above voltage recovery level before actual voltage recovery should be considered and continuation of the sequence can commence.
AllowedVoltRecoveryTime	The amount of time that should be considered before the sequence is cancelled and reverted back to idle due to failed voltage recovery.
MaxNrVoltageDecays	The max number of voltage decays allowed from initial decay (also counted) to the sequence can be completed. After a completed sequence, the number of voltage decays is reset back to 0.
MaxLimit_IncomerA	For future use.
MaxLimit_IncomerB	For future use.
ReAccGroupDelayTime	Restart delay is the delay that should be considered from last running feedback signal is received to the next priority group can be started.
OpenCircuit_time	Time from detection of undervoltageto reconnection of motor to electrical network is restricted.
AllowedReaccel_time	Time from undervoltageto reacceleration should succeed.
CloseCMD_time	Pulse length for close command.
RunningFb_timeout	Time from close command is given to running feedback is expected.
MaxNrOfAttempts	Max number of restart attempts.
UpdateRunningFbT	Update rate for scanning running feedback in Idle state.
SignalTransitionT	Transition time for undervoltage and recovery voltage signals.

The project should consider the restart delay time setting depending on what the functionality behind the signal “Motor running status” is based on:

- If it is based on a simple “breaker closed feedback”, then the delay between priorities is meant to wait for electrical network response after reconnection of a group of loads and proceed to next group if network stability allows it (voltage above recovery level at the end of group delay time). It is then assumed that response of the network will come within the priority delay time.
- If it is based on operation conditions of the motor (speed, torque, current, etc.), then that signal will contain the information that motor reached a stable operation state and its transient period due to acceleration has ended. Therefore, waiting for group delay to elapse is no longer needed and those timers become obsolete.

The project should consider at which point the trip of a motor was due to undervoltage. The order the signals are received by the control module might be misleading. The motors that have lost the “Motor running status” moments before an undervoltage, should be scheduled and subsequently restarted. The loads are scheduled if Motor running status was high 1-2s (UpdateRunningFbT) before undervoltage occurred.

3.1.15.1 Load configuration

See chapter 3.1.9.7 for details.

3.1.15.2 Priority configuration

Each priority group can be configured separately to specify time settings and required delay time for that specific priority group, otherwise settings may be inherited from global settings.

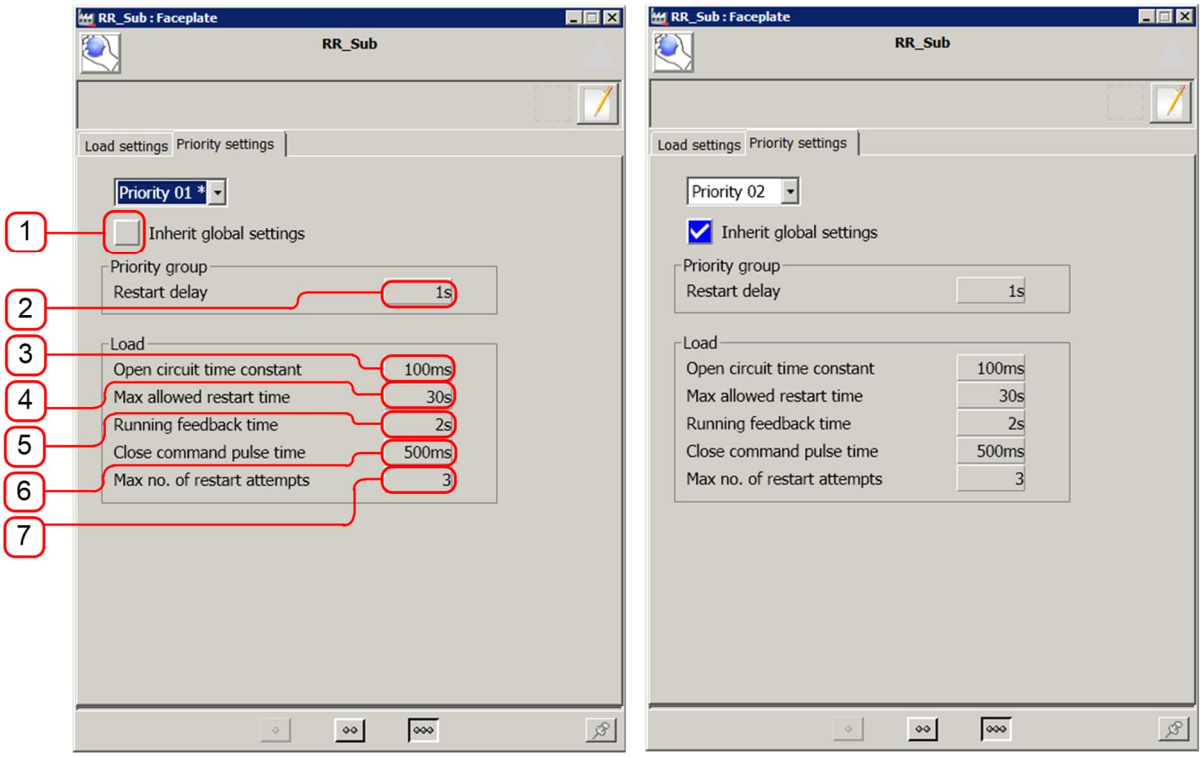


Figure 12 Faceplate configuration of priority group settings

Table 19 Main faceplate automatic block interaction

Field	Indication	Item	Description
1	Inherit global settings	Check box	Enables or disables inheritance of global settings.
2	Restart delay	Input field	Delay from last running feedback signal is received to the next priority group can be started.
3	Open circuit time constant	Input field	Time from detection of undervoltage to reconnection of motor to electrical network is restricted.
4	Max allowed restart time	Input field	Time from undervoltage to reacceleration should succeed.
5	Running feedback time	Input field	Time from close command is given to running feedback is expected.
6	Close command pulse time	Input field	Pulse length for close command.
7	Max no. of restart attempts	Input field	Max number of restart attempts.

The Global settings are only configurable from the faceplate if *GlobalPrioConfig.Global.OperatorSettable* is set to True and for users with Configure permission.

It is also possible to configure each priority via parameter *GlobalPrioConfig.PrioGroup01-20.* See Table 18 for similar description of the parameters.

3.1.16 Operator interface

3.1.16.1 Main view

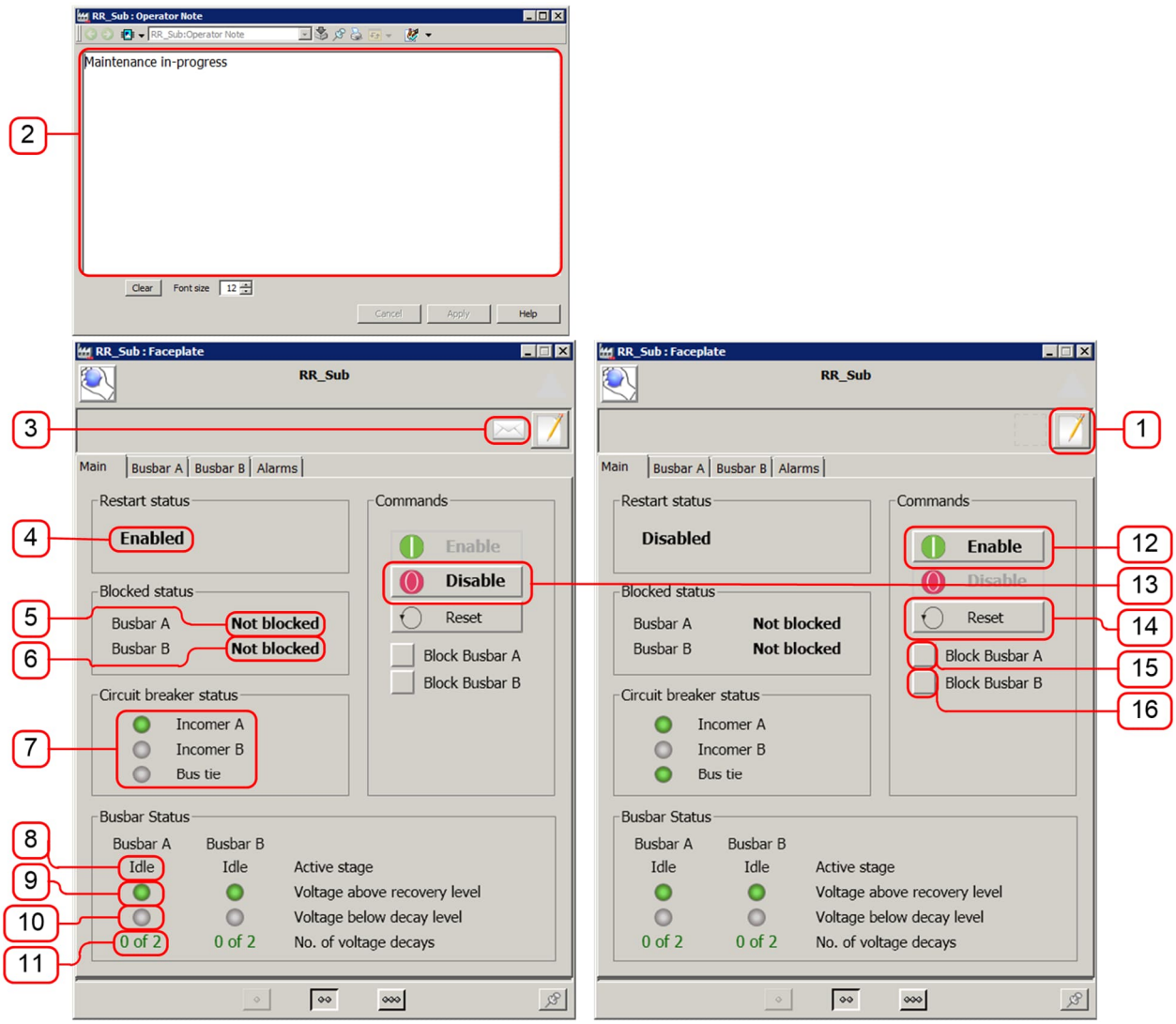


Figure 13 Faceplate element Main view

Table 20 Faceplate element Main View description

Field	Indication	Item	Description
1	Operator note editor link	Push button link	Push button link to the operator note editor
2	Operator note editor	Input area	This is an aspect in the 800xA for operator to make notes. Only users with *Operate* permission can write notes.
3	Operator note holds information	Icon	A letter icon indicates that the operator note holds data. Empty operator notes are indicated as empty field.
4	Function status	Text Indication	Indicates the function statues (Enable or disabled)
5	Blocked status busbar A	Text Indication	If configured in the *GlobalPrioConfig.Global. BusbarConfiguration* parameter, this field indicates the block status of busbar A (the status could be any of these four; Not blocked, Operator, Internal, External)

Field	Indication	Item	Description
6	Blocked status busbar B	Text Indication	If configured in the *GlobalPrioConfig.Global. BusbarConfiguration* parameter, this field indicates the block status of busbar B (the status could be any of these four; Not blocked, Operator, Internal, External)
7	Circuit breaker status	Indication symbols	The configured circuit breakers according to *GlobalPrioConfig.Global. CBConfigured* will indicate closed position with a green filled circle. Grey filled circle indicates the circuit breaker is in open position. The circuit breaker text is configurable via General Property aspect BreakerText.
8	Active stage	Text Indication	The busbars configured in the *GlobalPrioConfig.Global. BusbarConfiguration,* indicates the currently active stage in the sequence. (the status could be any of these three; Idle, Arming, Operational)
9	Voltage above recovery level	Text Indication	The busbars configured in the *GlobalPrioConfig.Global. BusbarConfiguration,* indicates voltage above recovery level with a green filled circle. Grey filled circle indicates voltage is below recovery level.
10	Voltage below decay level	Text Indication	The busbars configured in the *GlobalPrioConfig.Global. BusbarConfiguration,* indicates under voltage with a red filled circle. Grey filled circle indicates voltage is above under voltage level.
11	No. of voltage decays	Text Indication	The busbars configured in the *GlobalPrioConfig.Global. BusbarConfiguration,* indicates how many voltage decays has occurred since last completed sequence. If no. of voltage decays reaches max no. of voltage decays, the function is blocked (Internal) and a reset is required to normalize.
12	Enable	Command button	This command button set the function to enable state. Only users with *Operate* permission can access the button.
13	Disable	Command button	This command button set the function to disabled state. Only users with *Operate* permission can access the button.
14	Reset	Command button	This command button resets the no. of voltage decays. Only users with *Operate* permission can access the button.
15	Block busbar A	Check box	If configured in the *GlobalPrioConfig.Global. BusbarConfiguration,* there is a command option to Block or Unblock the busbar. If both busbars are connected together, a block on one busbar will block both busbars. Only users with *Operate* permission can access the checkbox.
16	Block busbar B	Check box	If configured in the *GlobalPrioConfig.Global. BusbarConfiguration,* there is a command option to Block or Unblock the busbar. If both busbars are connected together, a block on one busbar will block both busbars. Only users with *Operate* permission can access the checkbox.

The texts in the faceplate elements are configurable via the NLS Resource Manager aspect in Plant Explorer under **Library Structure > Preferences & Customizations > pms_NLS > pmsReAccelLib_PG2 \ pmsBasicText_PG2 > NLS_pmsReAccelLib_PG2 \ NLS_BasicText_PG2.**

3.1.16.2 Busbar A&B

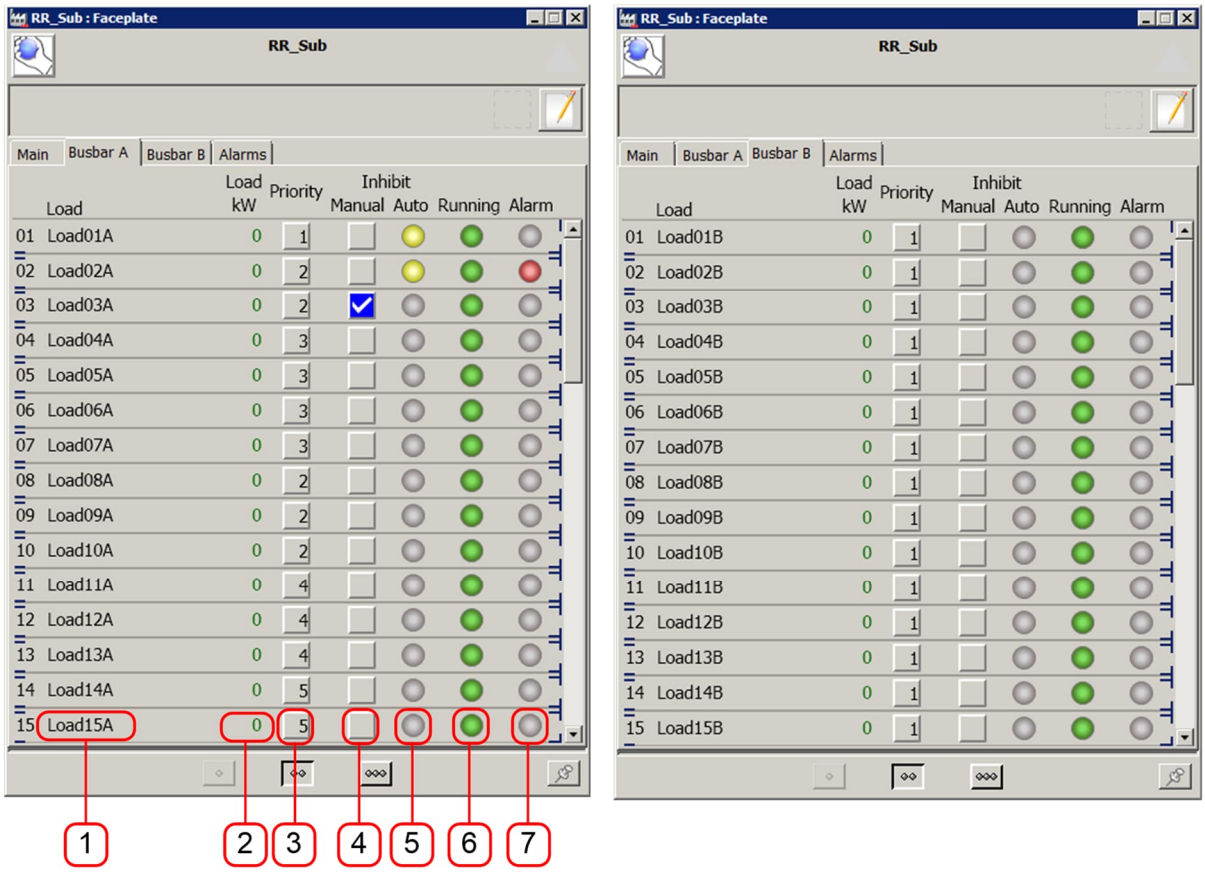


Figure 14 Faceplate element Busbar A&B

Table 21 Faceplate element Busbar A & B description

Field	Indication	Item	Description
1	Load tag	Text Indication	This identifies each load from parameter *Load.LoadPar.SrcName*.
2	Load value	Text Indication	If configured in the *GlobalPrioConfig.Global. HideLoadValue* parameter, this field indicates the value assign to *Load.LoadPar.PLoad*.
3	Priority definition	Checkbox	Configuration of the priority for the restarting function. Only users with *Tune* permission can access the button.
4	Manual Inhibit	Checkbox	Inhibit load from restarting sequence. Only users with *Tune* permission can access the button.
5	Auto Inhibit	Indication symbol	Automatic inhibit of load from restarting sequence (External Inhibit, I/O Error,).
6	Running Feedback	Indication symbol	A green filled circle indicates the load is running. A grey filled circle indicates the load is not running and a yellow filled circle indicates the signal is forced from the I/O channel.
7	General Alarm	Indication symbol	A red filled circle indicates a general alarm is high (Configuration error or I/O error). A grey filled circle indicates that no alarm is active.

3.1.16.3 Alarm tab

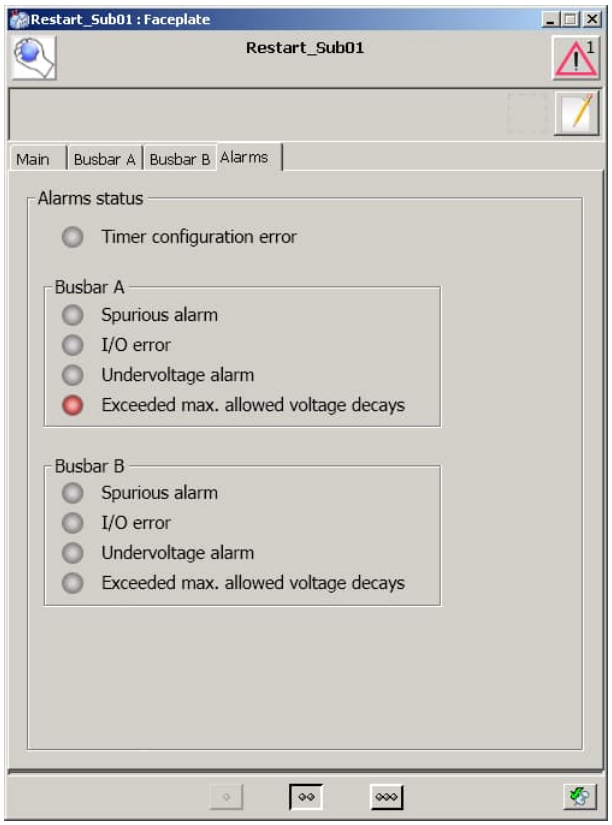


Figure 15 Faceplate element Alarms

See chapter 3.1.11.2 Alarms for more details.

3.1.16.4 Load Setting

See chapter 3.1.9.7 Individual configuration timers.

3.1.16.5 Priority Setting

See chapter 3.1.15 Configuration.

3.1.17 Reporting

The restart & reacceleration function provides a reporting feature to supplement the event list for troubleshooting and monitoring of the function. This chapter describes the step to setup a report for control module pmsReAccSub.

3.1.17.1 PMS data collector

How to install and configure the PMS data collector service is described in the PMS Report Data Collector User Manual. Please ignore steps in the manual about setting up the PMS Report Data Configuration aspect as the chapters below describe the steps for control module pmsReAccSub.

3.1.17.2 Setup reporting aspect

From the installation package, the library provides a preconfigured *Reporting_Template* aspect of type PMS report data configuration that has all necessary OPC variables available for the report.

This *Reporting_Template* aspect is only available in the Object Type Structure in 800xA PPA environment. The following steps must be completed for the reporting function to be available for the projects application instance of pmsReAccSub.

Step 1) Go to Object Type Structure and right click on the aspect *Reporting_Template* and click copy, as shown in Figure 16.

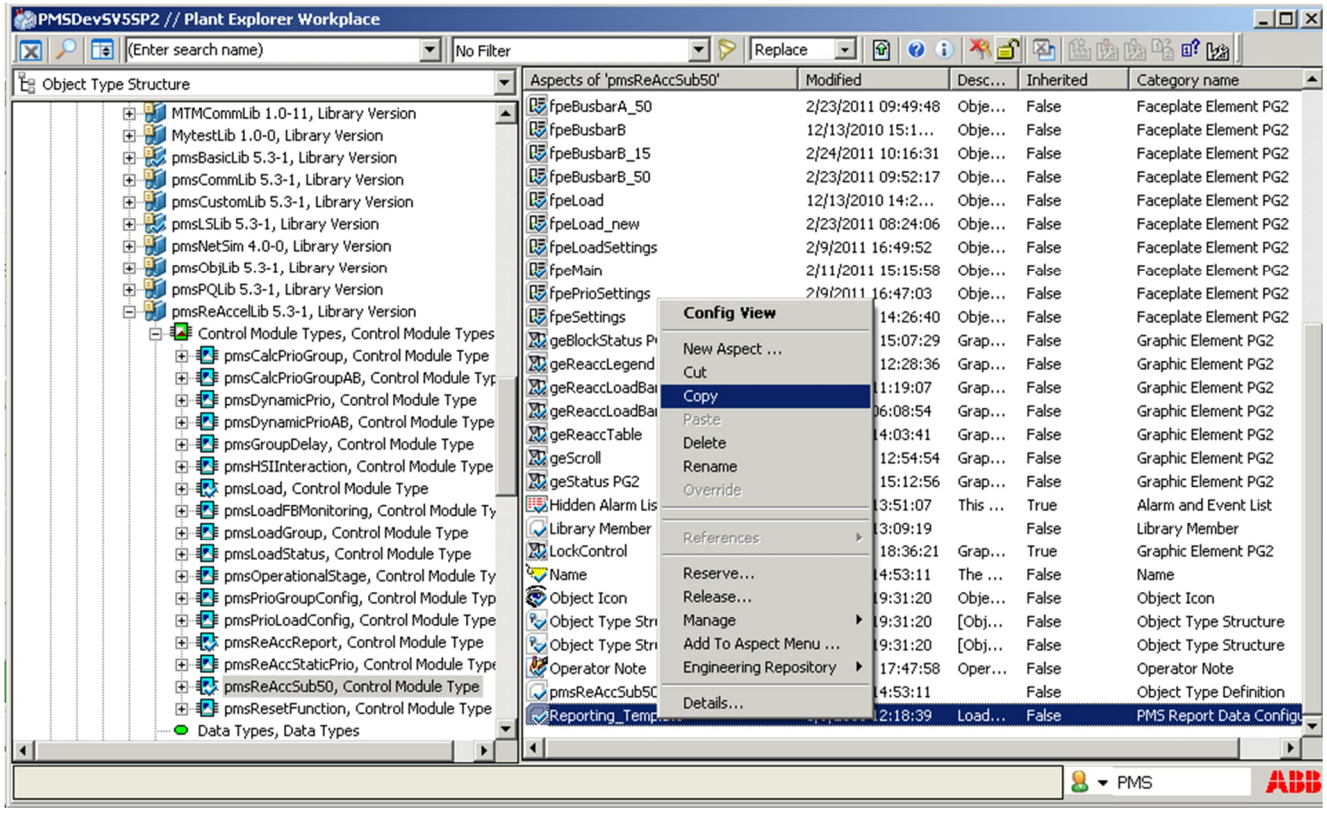


Figure 16 Export reporting template

Step 2) Go to Control Structure where the pmsReAccSub is instantiated, right click and select Paste. It should look similar to Figure 17.

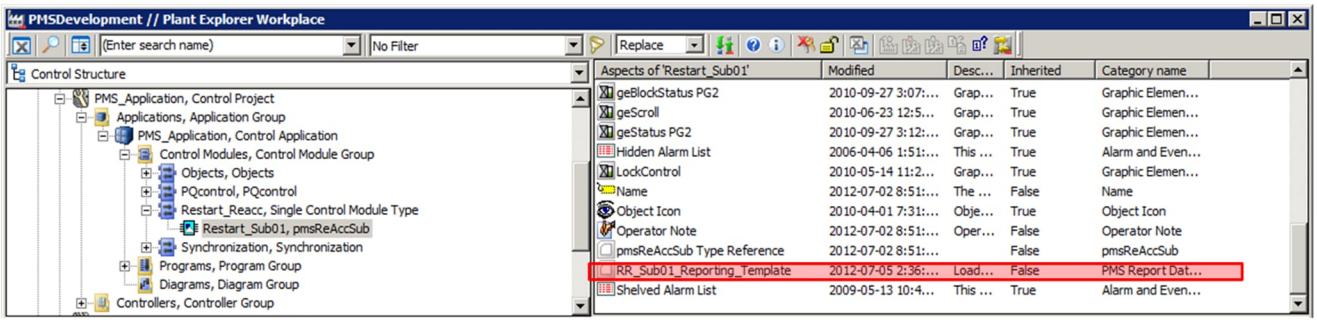


Figure 17 Copied reporting template

Step 3) Right click on the newly copied *Reporting_Template* (Sub01_Reporting_Template in this example) and click on Config View to edit Report paths, as shown Figure 18.

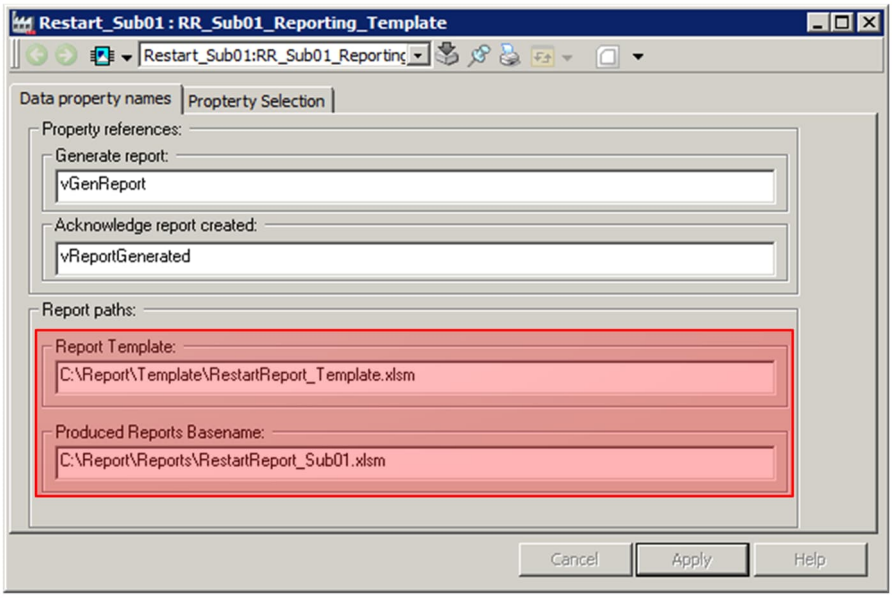


Figure 18 Reporting template Config view

Do not change Generate report or Acknowledge report created as this will lead to failure to report.

Step 4) Create folders for the Microsoft Excel template and reports as configured in the previous step and copy the excel template provided to the template folder.

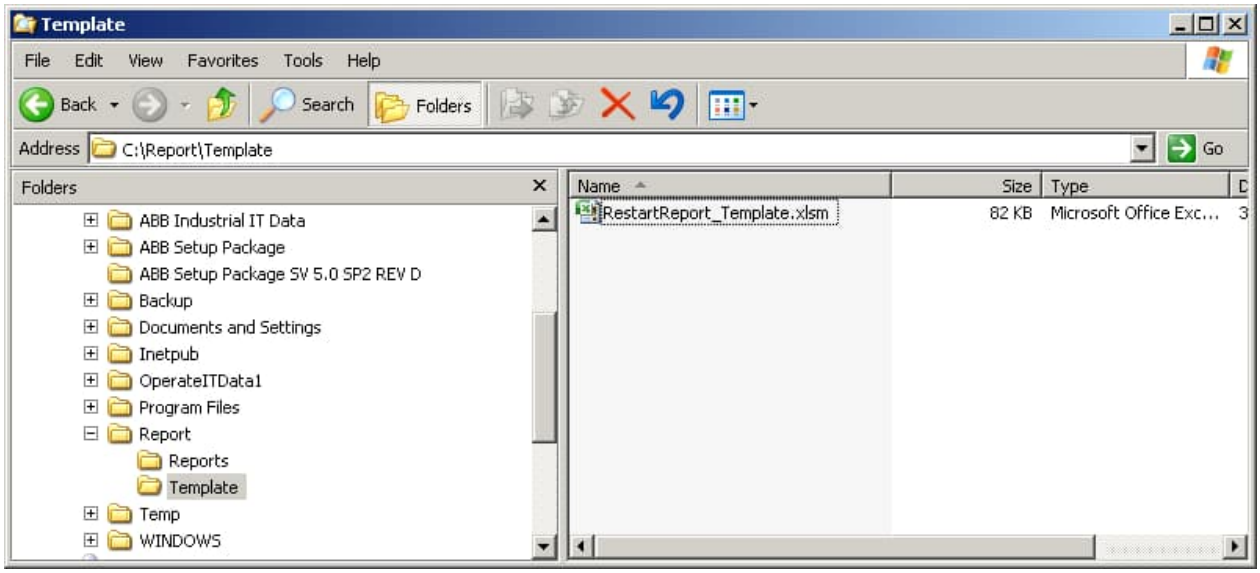


Figure 19 Excel reporting template

Step 5) Trigger the report and find the generated report in the report folder as shown Figure 20.

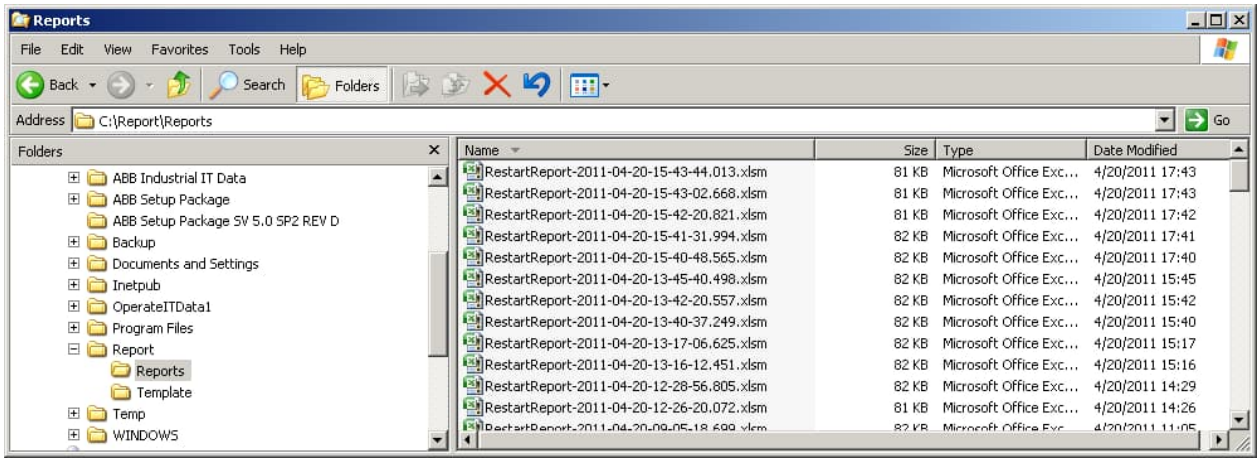
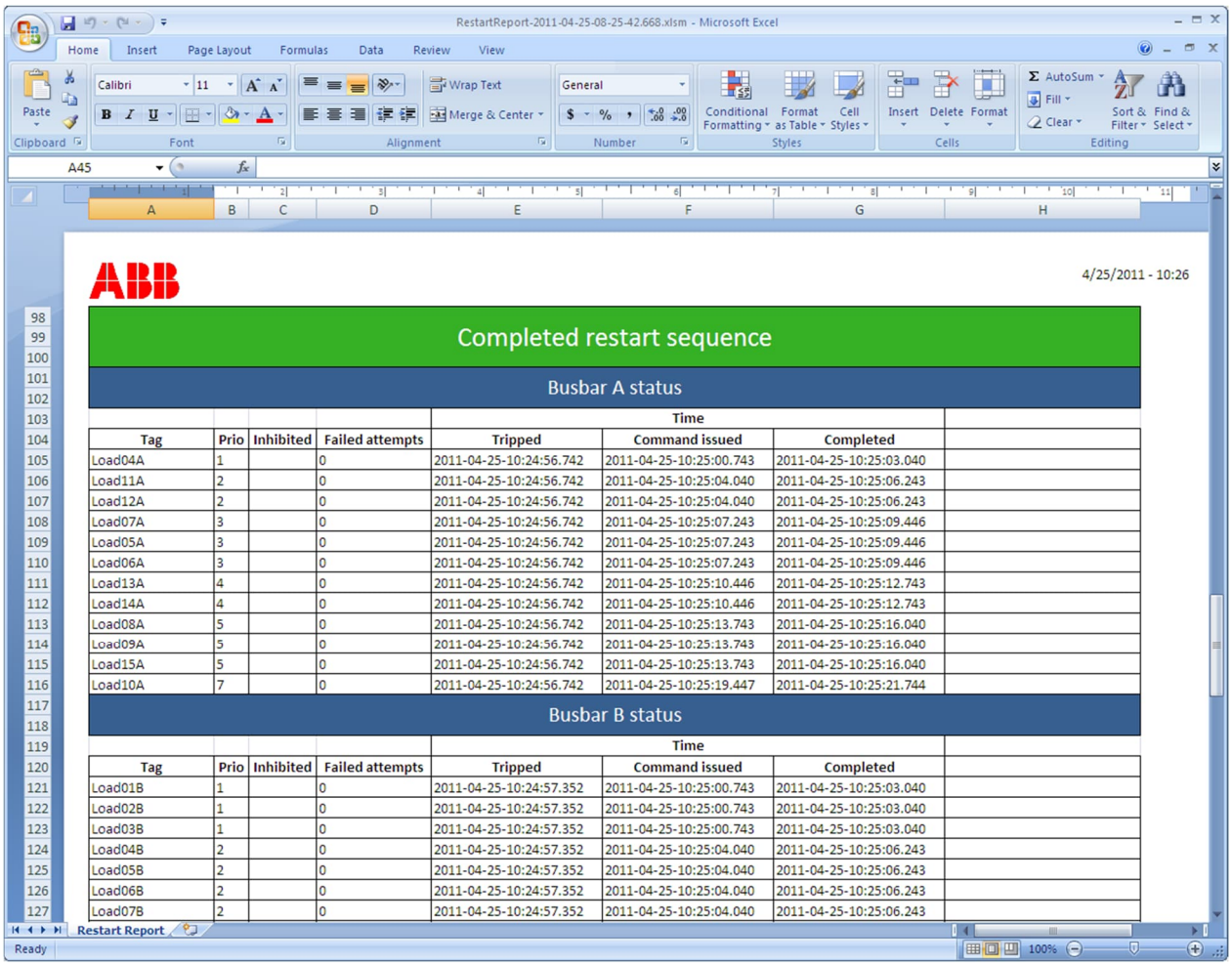


Figure 20 Completed reports

Step 6) Double-click the excel file to view. The report is divided into several sections; Voltage sag occurrence, Failed to reconnect, Inhibited from restart sequence, Completed restart sequence and Not running prior to voltage sag. See Figure 21 for an example of one of the sections.



4/25/2011 - 10:26

Completed restart sequence

Busbar A status

				Time		
Tag	Prio	Inhibited	Failed attempts	Tripped	Command issued	Completed
Load04A	1		0	2011-04-25-10:24:56.742	2011-04-25-10:25:00.743	2011-04-25-10:25:03.040
Load11A	2		0	2011-04-25-10:24:56.742	2011-04-25-10:25:04.040	2011-04-25-10:25:06.243
Load12A	2		0	2011-04-25-10:24:56.742	2011-04-25-10:25:04.040	2011-04-25-10:25:06.243
Load07A	3		0	2011-04-25-10:24:56.742	2011-04-25-10:25:07.243	2011-04-25-10:25:09.446
Load05A	3		0	2011-04-25-10:24:56.742	2011-04-25-10:25:07.243	2011-04-25-10:25:09.446
Load06A	3		0	2011-04-25-10:24:56.742	2011-04-25-10:25:07.243	2011-04-25-10:25:09.446
Load13A	4		0	2011-04-25-10:24:56.742	2011-04-25-10:25:10.446	2011-04-25-10:25:12.743
Load14A	4		0	2011-04-25-10:24:56.742	2011-04-25-10:25:10.446	2011-04-25-10:25:12.743
Load08A	5		0	2011-04-25-10:24:56.742	2011-04-25-10:25:13.743	2011-04-25-10:25:16.040
Load09A	5		0	2011-04-25-10:24:56.742	2011-04-25-10:25:13.743	2011-04-25-10:25:16.040
Load15A	5		0	2011-04-25-10:24:56.742	2011-04-25-10:25:13.743	2011-04-25-10:25:16.040
Load10A	7		0	2011-04-25-10:24:56.742	2011-04-25-10:25:19.447	2011-04-25-10:25:21.744

Busbar B status

				Time		
Tag	Prio	Inhibited	Failed attempts	Tripped	Command issued	Completed
Load01B	1		0	2011-04-25-10:24:57.352	2011-04-25-10:25:00.743	2011-04-25-10:25:03.040
Load02B	1		0	2011-04-25-10:24:57.352	2011-04-25-10:25:00.743	2011-04-25-10:25:03.040
Load03B	1		0	2011-04-25-10:24:57.352	2011-04-25-10:25:00.743	2011-04-25-10:25:03.040
Load04B	2		0	2011-04-25-10:24:57.352	2011-04-25-10:25:04.040	2011-04-25-10:25:06.243
Load05B	2		0	2011-04-25-10:24:57.352	2011-04-25-10:25:04.040	2011-04-25-10:25:06.243
Load06B	2		0	2011-04-25-10:24:57.352	2011-04-25-10:25:04.040	2011-04-25-10:25:06.243
Load07B	2		0	2011-04-25-10:24:57.352	2011-04-25-10:25:04.040	2011-04-25-10:25:06.243

Figure 21 Opened report in Excel

3.1.17.3 Report delay configuration

To achieve a correct data collection for all the necessary internally timestamped data, the project must configure two delay setting in Project constants. Both can be found under **pmsConstants -> cRestartReportDelayT and cRestartReportDelayT 2**, see Figure 22.

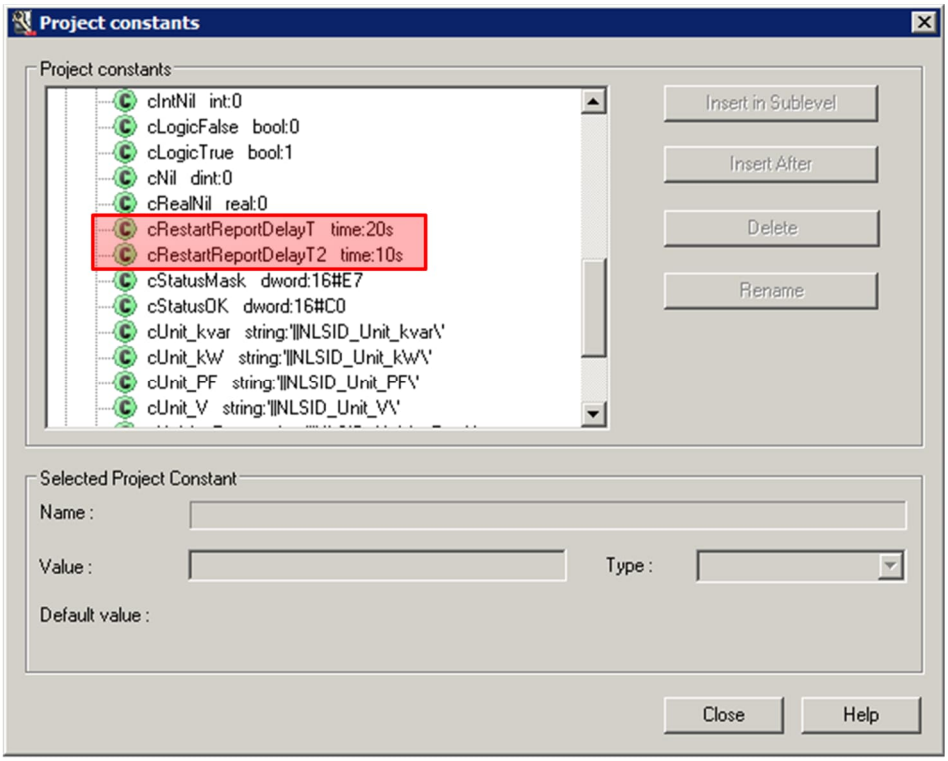


Figure 22 Faceplate element Alarms

If these settings are set too “fast” the data collector will collect data before the timestamps are written to the variables, hence the long default delay

The delay timer cRestartReportDelayT sets the delay from when the sequence is completed to report trigger is given to the reporting service. The delay timer cRestartReportDelayT2 sets the delay between two reports in cases where there are several reports waiting in temporary storage before being sent to the data collector.

The reporting function can temporarily store up to four reports before the oldest is overridden.

Enable the Reporting via *GlobalPrioConfig.Global.EnableReporting*.

3.1.18 Alarms

The control module pmsReAccSub shall monitor the alarm conditions and will generate the alarm messages, as described in table below.

Table 22 Alarm configuration

No	Description	Alarm condition	Alarm message (Resource Id)
1	Undervoltage busbar A	When undervoltage is detected either via parameter *BusAUnderVoltage* OR analogue voltage measurement < Voltage decay level (EnableSecUndervoltageCheck must be enabled).	*BusbarA* + ‘ ‘ + NLSID_RestartUnderVoltage
2	Undervoltage busbar B	When undervoltage is detected either via parameter *BusBUnderVoltage* OR analogue voltage measurement < Voltage decay level (EnableSecUndervoltageCheck must be enabled).	*BusbarB* + ‘ ‘ + NLSID_RestartUnderVoltage
3	Spurious busbar A	When the Undervoltage and recovery voltage is detected at the same time. This results in an internal block for busbar A. If both busbars are interconnected both have to experience the problem to block the function.	*BusbarA* + ‘ ‘ + NLSID_RestartSpuriousCond
4	Spurious busbar B	When the Undervoltage and recovery voltage is detected at the same time. This results in an internal block for busbar B. If both busbars are interconnected both have to experience the problem to block the function.	*BusbarB* + ‘ ‘ + NLSID_RestartSpuriousCond
5	Exceeded max. allowed voltage decays busbar A	When the No. of voltage decays on busbar A is equal to Max no. of voltage decays.	*BusbarA* + ‘ ‘ + NLSID_RestartMaxVoltDecay
6	Exceeded max. allowed voltage decays busbar B	When the No. of voltage decays on busbar B is equal to Max no. of voltage decays.	*BusbarB* + ‘ ‘ + NLSID_RestartMaxVoltDecay

No	Description	Alarm condition	Alarm message (Resource Id)
7	Configuration error	When GlobalPrioConfig.Global.AllowedReaccel_time < GlobalPrioConfig.Global.OpenCircuit_time OR GlobalPrioConfig.Global.AllowedReaccel_time < GlobalPrioConfig.Global.CloseCMD_time + GlobalPrioConfig.Global.RunningFb_timeout OR GlobalPrioConfig.Global.MaxPriority < 1 OR GlobalPrioConfig.Global.MaxPriority > 20 OR GlobalPrioConfig.Global.VoltageDecayLevel > GlobalPrioConfig.Global.VoltageRecoveryLevel OR GlobalPrioConfig.Global.VoltageRecoveryHyst > GlobalPrioConfig.Global.AllowedVoltRecoveryTime OR GlobalPrioConfig.Global.MaxNrVoltageDecays < 1	NLSID_RestartConfigurationEror
8	IO Error busbar A	When BusAUnderVoltage.Status AND pmsConstants.cStatusMask OR BusARecoveryVoltage.Status AND pmsConstants.cStatusMask OR BusAVoltage.Status AND pmsConstants.cStatusMask is not equal pmsConstants.cStatusOK	*BusbarA* + ‘ ‘ + NLSID_IOError
9	IO Error busbar B	When BusBUnderVoltage.Status AND pmsConstants.cStatusMask OR BusBRecoveryVoltage.Status AND pmsConstants.cStatusMask OR BusBVoltage.Status AND pmsConstants.cStatusMask is not equal pmsConstants.cStatusOK	*BusbarB* + ‘ ‘ + *NLSID_IOError*
10	Load IO error	When Close_CMD.Status AND pmsConstants.cStatusMask OR RunningStatus.Status AND pmsConstants.cStatusMask OR POCLoc.Status AND pmsConstants.cStatusMask is not equal pmsConstants.cStatusOK	*Load.SrcName* + ‘ ‘ NLSID_IOError
11	Load configuration error	When Max allowed restart time < Open circuit time constant OR Max allowed restart time < Close command pulse time + Running feedback time.	*Load.SrcName* + ‘ ‘ NLSID_ConfigError

The text messages (i.e. in Alarm List) for the above alarms are configurable via the Alarm and Event Translator aspect in Plant Explorer under **Library Structure > Preferences & Customizations > PmsLibrary_NLS_Alarm&Event**. These messages can be changed from English language to another language by modifying the relevant Resource Id (i.e. NLSID_….) to a desired message (a text string up to 60 characters).

3.1.19 Events

The control module pmsReAccSub shall monitor the event conditions and will generate the event messages, as described in Table 23.

Table 23 Event configuration

No	Description	Event condition	Event message (Resource Id)
1	Report generated	When acknowledge report created is received.	*BusbarA* + ‘ ‘ + NLSID_ReportGenerated
2	Active stage busbar A	When undervoltage is detected on busbar A the active stage is set to arming. When recovery voltage is detected and filter time has elapsed after an undervoltage, the active stage is set to operational. When the sequence is complete (i.e all loads has either started or failed to start) or cancelled due to voltage failed to recover within allowed voltage recovery time, the active stage is set to idle.	*BusbarA* + ’ ’ + NLSID_RestartArmingState OR *BusbarA* + ’ ’ + NLSID_RestartOperationState OR *BusbarA* + ’ ’ + NLSID_RestartIdleState
3	Active stage busbar B	When undervoltage is detected on busbar B the active stage is set to arming. When recovery voltage is detected on busbar B and filter time has elapsed after an undervoltage, the active stage is set to operational. When the sequence is complete (i.e all loads has either started or failed to start) or cancelled due to voltage failed to recover within allowed voltage recovery time, the active stage is set to idle.	*BusbarB* + ’ ’ + NLSID_RestartArmingState OR *BusbarB* + ’ ’ + NLSID_RestartOperationState OR *BusbarB* + ’ ’ + NLSID_RestartIdleState
4	Voltage recovery timeout bus A	When voltage has not recovered on busbar A before Max allowed voltage recovery time setting.	*BusbarA* + ‘ ‘ + NLSID_RestartOperationAbort – NLSID_RestartVoltageTO
5	Voltage recovery timeout bus B	When voltage has not recovered on busbar B before Max allowed voltage recovery time setting.	*BusbarB* + ‘ ‘ + NLSID_RestartOperationAbort – NLSID_RestartVoltageTO

No	Description	Event condition	Event message (Resource Id)
6	Block bus A	When operator check block busbar A in faceplate element Main is set, the block event is set to operator block. When *ExtPar.ExBlock_BusbarA* is set, the block event is set to *ExtPar.ExBlock_BusbarAEvent*. When I/O Error (on both busbar if interconnected) OR spurious condition (on both busbar if interconnected) OR no. of voltage decays > Max no. of voltage decays then the block event is set to Internal	*Busbar A* + ‘ ‘ + NLSID_RestartIntBlock OR *Busbar A* + ‘ ‘ + NLSID_RestartOpBlock OR *Busbar A* + ‘ ‘ + *ExtPar.ExBlock_BusbarAEvent* *Busbar A* + ‘ ‘ + NLSID_RestartUnBlocked
7	Block bus B	When operator check block busbar B in faceplate element Main is set, the block event is set to operator block. When *ExtPar.ExBlock_BusbarB* is set, the block event is set to *ExtPar.ExBlock_BusbarBEvent*. When I/O Error (on both busbar if interconnected) OR spurious condition (on both busbar if interconnected) OR no. of voltage decays > Max no. of voltage decays then the block event is set to Internal.	*Busbar B* + ‘ ‘ + NLSID_RestartIntBlock OR *Busbar B* + ‘ ‘ + NLSID_RestartOpBlock OR *Busbar B* + ‘ ‘ + *ExtPar.ExBlock_BusbarAEvent* *Busbar B* + ‘ ‘ + NLSID_RestartUnBlocked
8	Function active	When parameter *Enable* is activated or deactivated.	NLSID_RREnableOn on activation and NLSID_RREnableOff on deactivation
9	Priority released bus A	When priority is released for busbar A in operational stage.	*Busbar A* + ‘ ‘ + NLSID_RestartPrioRelease + ‘ ‘ + X
10	Priority released bus B	When priority is released for busbar A in operational stage.	*Busbar B* + ‘ ‘ + NLSID_RestartPrioRelease + ‘ ‘ + X
11	Load – Operator inhibit	When manually inhibited is checked via faceplate element busbar A\B.	*LoadSrc* + “ “ + NLSID_RestartInhOpOn AND *LoadSrc* + “ “ + NLSID_RestartInhOpOff
12	Load – Auto inhibit	When RunningStatus or Close_CMD or POCLoc feedback signals are in error condition OR when Load.ExternalInhibit.	*LoadSrc* + “ “ + NLSID_RestartInhSysOn AND *LoadSrc* + “ “ + NLSID_RestartInhSysOff
13	Load – Failed to restart	When either 1) Number of restart attempts has been reached 2) Time since disconnection > Max allowed restart time 3) Sequence Cancelled	1) *LoadSrc* + ‘ ‘ + NLSID_RestartFailed + ‘ ‘ + NLSIDFailCause2 2) *LoadSrc* + ‘ ‘ + NLSID_RestartFailed + ‘ ‘ + NLSIDFailCause1 3) *LoadSrc* + ‘ ‘ + NLSID_RestartFailed + ‘ ‘ + NLSIDFailCause3

No	Description	Event condition	Event message (Resource Id)
14	Load – Running status	When parameter *Load.RunningStatus* is True or False.	NLSID_RestartRunningFbOn AND NLSID_RestartRunningFbOff
15	Load – command issued	When parameter *Load.Close_CMD* is set high.	*LoadSrc* + ‘ ‘ + NLSID_CmdClose
16	Load – Priority change	When operator or external logic change priority setting.	*LoadSrc* + ‘ ‘ + NLSID_RestartPrioChange + ‘ ‘ + X

The text messages (i.e. in Event List) for the above events are configurable via the Alarm and Event Translator aspect in Plant Explorer under **Library Structure > Preferences & Customizations > PmsLibrary_NLS_Alarm&Event**. These messages can be changed from English language to another language by modifying the relevant Resource Id (i.e. NLSID_….) to a desired message (a text string up to 60 characters).

3.1.20 Required I/O

An overview of the required I/O is described in Table 24 with respect to significance when setting up the interface connections for the pmsReAccSub control module. The description “mandatory” implies that the internal logic of the control modules requires data from the variable in order to function properly. The description “optional” implies that the internal logic of the control modules can also handle data from the variable, but it could be omitted without affecting the functionality of the module.

Table 24 Required I/O connections to the parameters

Field	Parameter	Description	Type	Importance
1	BusAUnderVoltage BusBUnderVoltage	Undervoltage stage activation of the busbars, which have connected loads, that are part of the reacceleration function – used for detection of undervoltage stage.	Digital Input	Mandatory
2	BusARecoveryVoltage BusBRecoveryVoltage	Recoveryvoltage stage activation of the busbars, which have connected loads that are part of the reacceleration function – used for detection voltage recovery.	Digital Input	Mandatory
3	BusAVoltage BusBVoltage	Voltage of the busbars, which have connected loads, that are part of the reacceleration function.	Analogue Input	Optional
4	LoadPar.Running status	Running status for the motors, that are part of the reacceleration sequence.	Digital Input	Mandatory
5	LoadPar.Close_CMD	Breaker close command for the motors, that are part of the reacceleration sequence.	Digital Output	Mandatory
6	LoadPar.POCLoc	Circuit breaker local point of control.	Digital Input	Optional
7	CBA,CBB,CBT	Circuit breaker confirmed close position.	Digital Input	Mandatory

Item 2: the functionality of an overvoltage type signal is preferred opposed to an undervoltage type as it is considered a safer approach for reacceleration to enter operational state based on a high (true) value of the input.

Item 3: from above table is used only when a secondary check of reacceleration triggering condition is desired.

Item 4: should give the information that motor has normal operating parameters (speed, torque, etc.). If this cannot be supplied, then motor breaker close status should be connected as minimum.

Item 5: must be field/hardwired. A software signal to PMS breaker control module would introduce undesired delays in the reconnection function.

Section 4 Capacity & Performance

Table 25 Firmware version downloaded in AC800M controllers

Unit	Firmware version
PM866	FW866 5.1.48.40 2010-07-02 (BasicHwLib 5.1-0)
PM891	FW891 5.1.48.40 2010-07-02 (BasicHwLib 5.1-0)

4.1 Heap utilization

4.1.1 pmsReAccSub

Table 26 Heap utilization of pmsReAccSub in AC800M controller

	First instance	2nd and following instances
Heap utilization(MB) in PM866	1.457	0.847
Heap utilization(MB) in PM891	1.452	0.841

4.2 Execution Time

4.2.1 pmsReAccSub

Table 27 Execution time of pmsReAccSub in AC800M controller

Execution time (ms)	Configuration	Notes
9 ms(idle) – 38 ms(peak)	100 loads, all prio 1, no disabled events	For one instance in PM866
9 ms(idle) – 15 ms(peak)	100 loads, all prio 1, 3 disabled events *	For one instance in PM866
6 ms(idle) – 24 ms(peak)	60 loads, all prio 1, no disabled events	For one instance in PM866
6 ms(idle) – 10 ms(peak)	60 loads, all prio 1, 3 disabled events *	For one instance in PM866
4 ms(idle) – 23 ms(peak)	40 loads, all prio 1, no disabled events	For one instance in PM866
4 ms(idle) – 7 ms(peak)	40 loads, all prio 1, 3 disabled events *	For one instance in PM866
3 ms(idle) – 13 ms(peak)	100 loads, all prio 1, no disabled events	For one instance in PM891
3 ms(idle) – 4 ms(peak)	100 loads, all prio 1, 3 disabled events *	For one instance in PM891
2 ms(idle) – 8 ms(peak)	60 loads, all prio 1, no disabled events	For one instance in PM891
2 ms(idle) – 3 ms(peak)	60 loads, all prio 1, 3 disabled events *	For one instance in PM891
1 ms(idle) – 6 ms(peak)	40 loads, all prio 1, no disabled events	For one instance in PM891
1 ms(idle) – 2 ms(peak)	40 loads, all prio 1, 3 disabled events *	For one instance in PM891

* Disabled events; *ExtPar.DisLoadEvCMD*, *ExtPar.DisLoadEvRunningFb* and *ExtPar.DisLoadEvRunningFbLost*

Section 5 Application Notes

A configuration example of a restart/reacceleration application will be described in the following sections. The substation shown in figure below will be used as a basis.

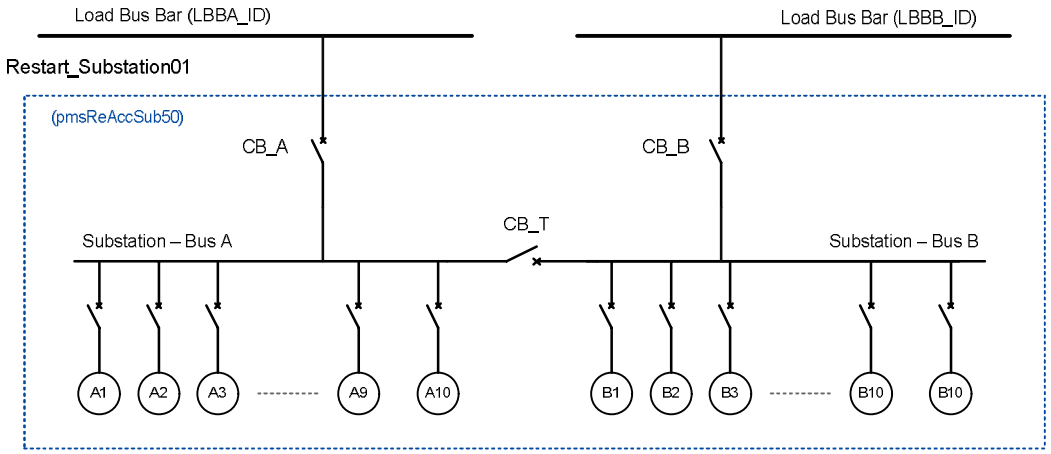


Figure 23 Substation configuration for pmsReAccSub

One instance of the type pmsReAccSub will be configured, denoted Restart_Substation01, configured with 20 loads, 10 on bus A and 10 on bus B.

5.1 Project Structure

A PMS project configured with the Control Builder M Professional is shown in Figure 24. The examples described in the following sections will be based on this project structure:

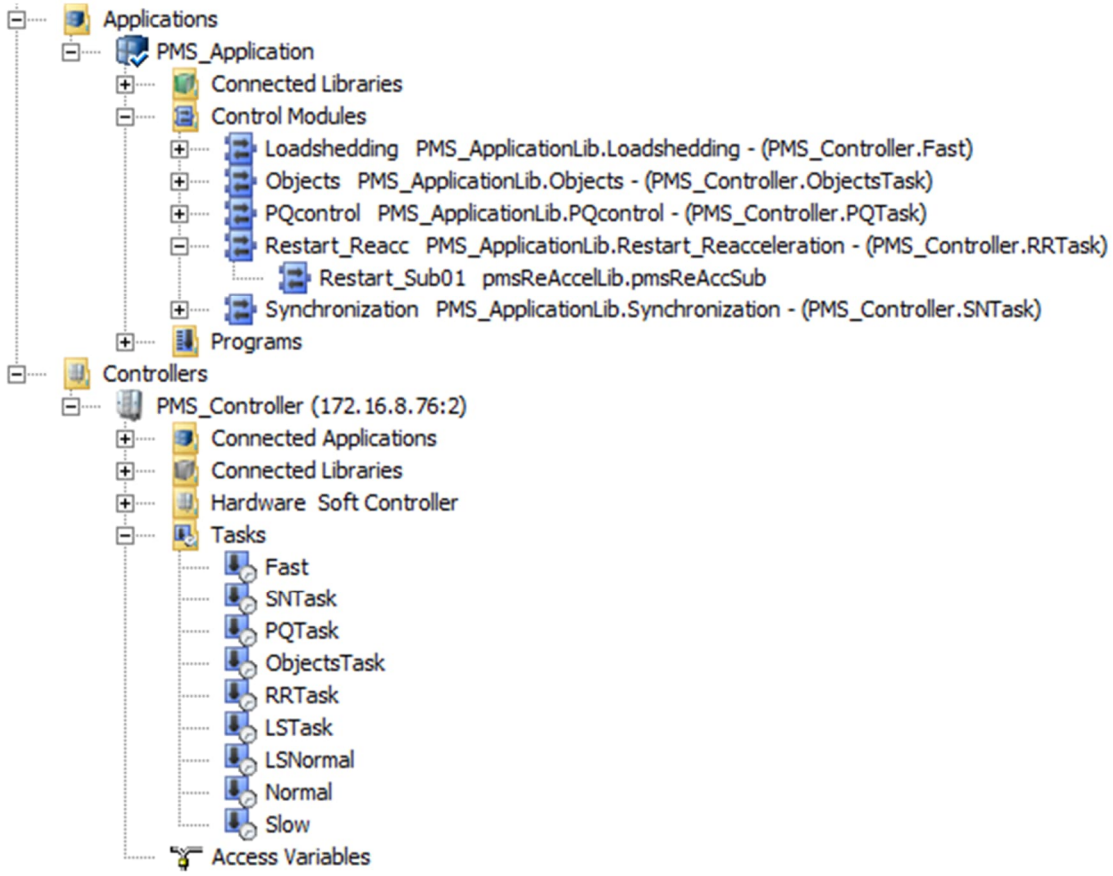


Figure 24 Example of project structure and instantiation

5.2 Instantiation and application data types

The control module type pmsReAccSub is instantiated in CMT (Control module type) under library application named *Restart_Reacceleration*, as shown in Figure 25.

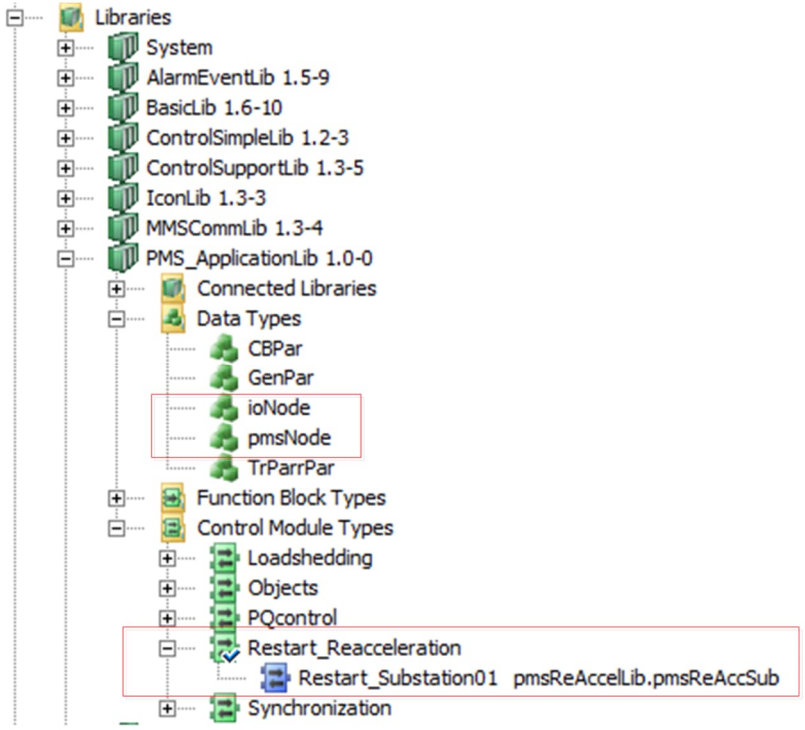


Figure 25 Instantiated type in application library

Create instances and name them e.g. *Restart_Substation01*. One instance must be created for each substation.

To connect the parameters of the instances it is recommended to create application specific data types. The application example in Figure 26 shows the data types named *pmsNode* and *ioNode*. *pmsNode* holds data types and configuration values, while *ioNode* holds IO variables.

These must be configured as global variables in the application:

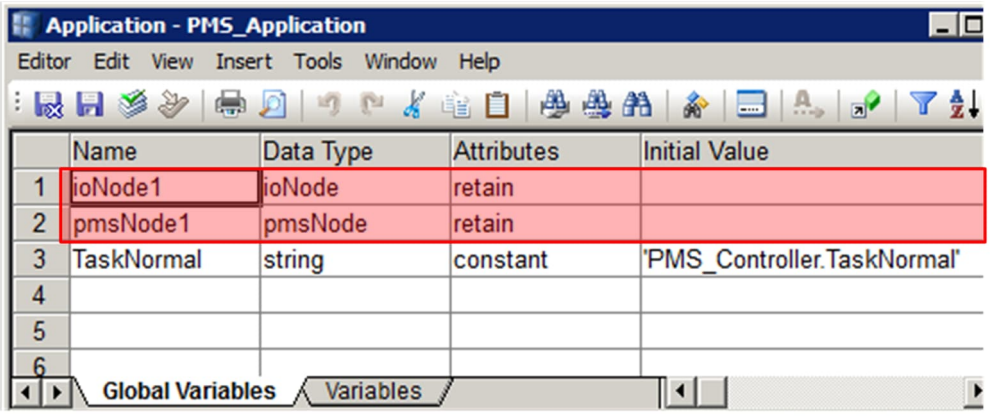


Figure 26 Defining global variables of the application

Add these to the External Variables list of the single control module:

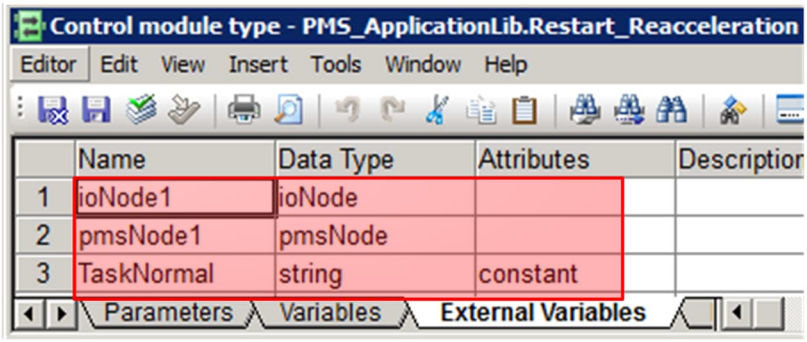


Figure 27 Parameter connections of single control module RestartReacc

A separate task RRFast is created for the execution of the restart/reacceleration instances; connect the control module Restart_Reacc to this task.

5.2.1 Configuration of parameters

Below is shown the connections of the first instance, items in *pmsNode1* must be created to provide the parameters shown, described in the following sections.

Connections - Restart_Substation01 pmsReAccelLib.pmsReAcc5ub50

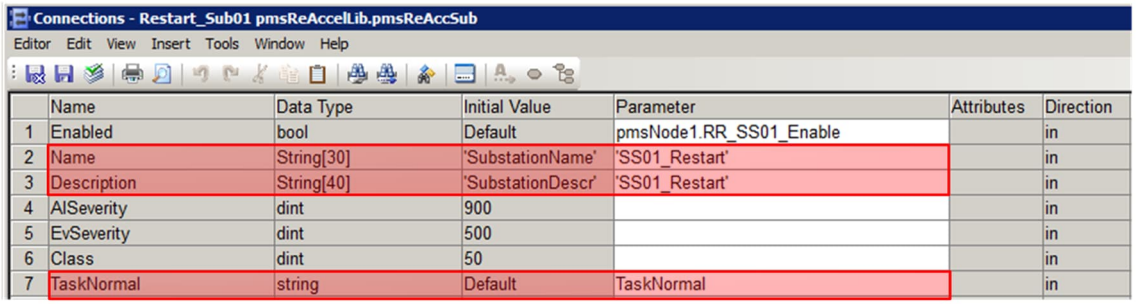
Editor Edit View Insert Tools Window Help

	Name	Data Type	Initial Value	Parameter	Attributes	Direction	Description
1	Enabled	bool	Default	pmsNode1.Restart_SS01Enable		in_out	IN: Enable Function
2	Name	String[30]	'SubstationName'	'Restart_SS01'		in_out	IN EDIT: Name
3	Description	String[40]	'SubstationDescr'	'Restart_SS01'		in_out	IN EDIT: Description
4	AlSeverity	dint	900			in_out	IN EDIT: Alarm severity
5	EvSeverity	dint	500			in_out	IN EDIT: Event severity
6	Class	dint	50			in_out	IN EDIT: Alarm and Event Class
7	TaskNormal	string	Default	TaskNormal		in_out	IN EDIT: Normal task for initial running loads and HSI (connectic
8	BusA	String[10]	'Busbar A'			in_out	IN EDIT: Text for busbar A
9	BusB	String[10]	'Busbar B'			in_out	IN EDIT: Text for busbar B
10	FunctionMode	dint	1	1		in_out	IN EDIT: [1=Restart][2=Reacceleration]
11	LBBA_ID	dint	1	pmsNode1.Restart_SS01_LBBA_ID		in_out	IN EDIT: ID of the Load Bus Bar which the substation A-side is
12	LBBB_ID	dint	2	pmsNode1.Restart_SS01_LBBA_ID		in_out	IN EDIT: ID of the Load Bus Bar which the substation B-side is
13	LoadsANOF	dint	5	pmsNode1.Restart_SS01_NumberOfLoadBusA		in_out	IN EDIT: Number of Loads in Bus A
14	LoadsBNOF	dint	5	pmsNode1.Restart_SS01_NumberOfLoadBusB		in_out	IN EDIT: Number of Loads in Bus B
15	BusAUnderVoltage	pmsReAccBoolIOex	Default	pmsNode1.Restart_SS01_BusAUnderVoltage		in_out	IN: Bus A Undervoltage
16	BusBUnderVoltage	pmsReAccBoolIOex	Default	pmsNode1.Restart_SS01_BusBUnderVoltage		in_out	IN: Bus B Undervoltage
17	BusARecoveryVoltage	pmsReAccBoolIOex	Default	pmsNode1.Restart_SS01_BusARecVoltage		in_out	IN: Bus A recovery voltage level
18	BusBRecoveryVoltage	pmsReAccBoolIOex	Default	pmsNode1.Restart_SS01_BusBRecVoltage		in_out	IN: Bus B recovery voltage level
19	BusAVoltage	RealIO	Default	pmsNode1.Restart_SS01_BusAVoltage		in_out	IN: Bus A Voltage
20	BusBVoltage	RealIO	Default	pmsNode1.Restart_SS01_BusBVoltage		in_out	IN: Bus B Voltage
21	EnableSecUnderVoltCheck	bool	Default	pmsNode1.Restart_SS01_SecUnderVoltCheck		in_out	IN: Enables Bus A secondary Voltage check.
22	EnableSecRecVoltCheck	bool	Default	pmsNode1.Restart_SS01_SecRecVoltCheck		in_out	IN: Enables Bus B secondary Voltage check.
23	GlobalPrioConfig	pmsReAccConfigPar	Default	pmsNode1.Restart_SS01_GlobalPrioConfig		in_out	IN: Priority Group configuration parameters
24	ExtPar	pmsReAccExtPar	Default	pmsNode1.Restart_SS01_ExtPar		in_out	IN: External parameters
25	ExtPrio	pmsReAccExtPrio	Default			in_out	IN: External Priority (Future)
26	CBA_C	bool	Default	pmsNode1.Restart_SS01_CBA_Closed		in_out	IN: Circuit breaker SS_A checked closed pos
27	CBB_C	bool	Default	pmsNode1.Restart_SS01_CBB_Closed		in_out	IN: Circuit breaker SS_B checked closed pos
28	CBT_C	bool	Default	pmsNode1.Restart_SS01_CBT_Closed		in_out	IN: Circuit breaker SS_T checked closed pos
29	LoadA	pmsReAccLoadParAll	Default	pmsNode1.Restart_SS01_LoadParA		in_out	IN_OUT: Load parameters, up to 50 Loads in substation A-side
30	LoadB	pmsReAccLoadParAll	Default	pmsNode1.Restart_SS01_LoadParB		in_out	IN_OUT: Load parameters, up to 50 Loads in substation B-side
31	Status	pmsReAccStatusPar	Default			in_out	OUT: Function Status

Figure 28. The connections of the instance Restart_Substation01

5.2.1.1 Name, Description and tasks

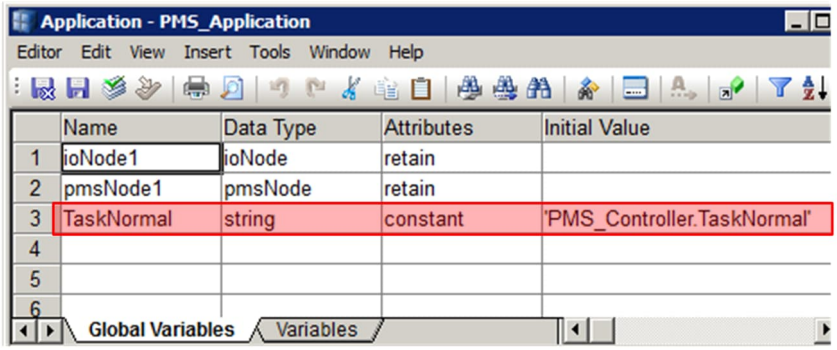
For the parameters *Name, Description*, use e.g. these strings:



	Name	Data Type	Initial Value	Parameter	Attributes	Direction
1	Enabled	bool	Default	pmsNode1.RR_SS01_Enable		in
2	Name	String[30]	'SubstationName'	'SS01_Restart'		in
3	Description	String[40]	'SubstationDescr'	'SS01_Restart'		in
4	AlSeverity	dint	900			in
5	EvSeverity	dint	500			in
6	Class	dint	50			in
7	TaskNormal	string	Default	TaskNormal		in

Figure 29 Name, Description and Task settings

The parameter "*TaskNormal*" should be connected to the global variable TaskNormal. In the application *RestartReaccAppl* shown above, there must then be defined a global variable that connects this string to the Normal task of the controller:



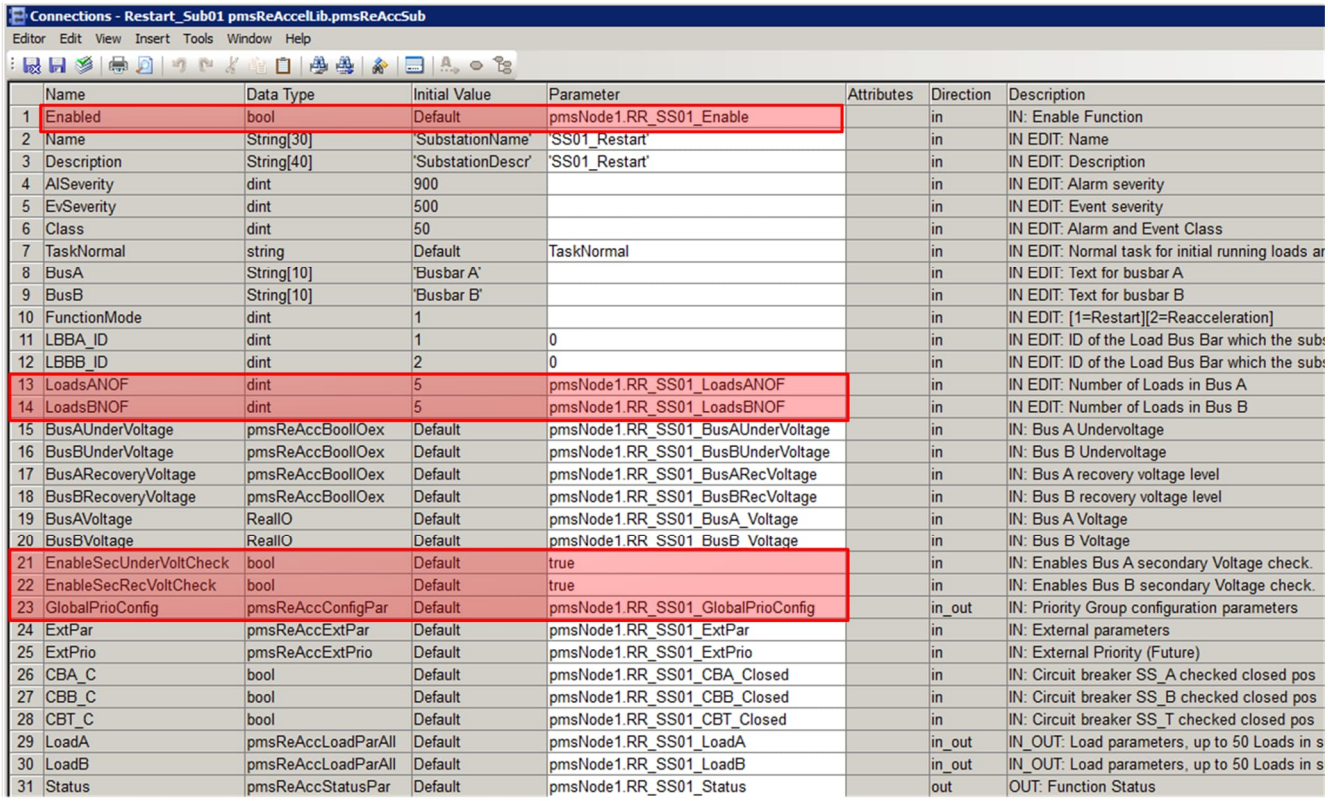
	Name	Data Type	Attributes	Initial Value
1	ioNode1	ioNode	retain	
2	pmsNode1	pmsNode	retain	
3	TaskNormal	string	constant	'PMS_Controller.TaskNormal'
4				
5				
6				

Figure 30 Task Normal

This task is used by the sub module HSI. The task "Normal" has typically an interval time of 1000 ms.

5.2.1.2 Configuration parameters

Parameter for enabling the instance, setting the load bus bar IDs, the number of loads on each bus bar must be created and connected; in this example also the optional secondary voltage checks are used:



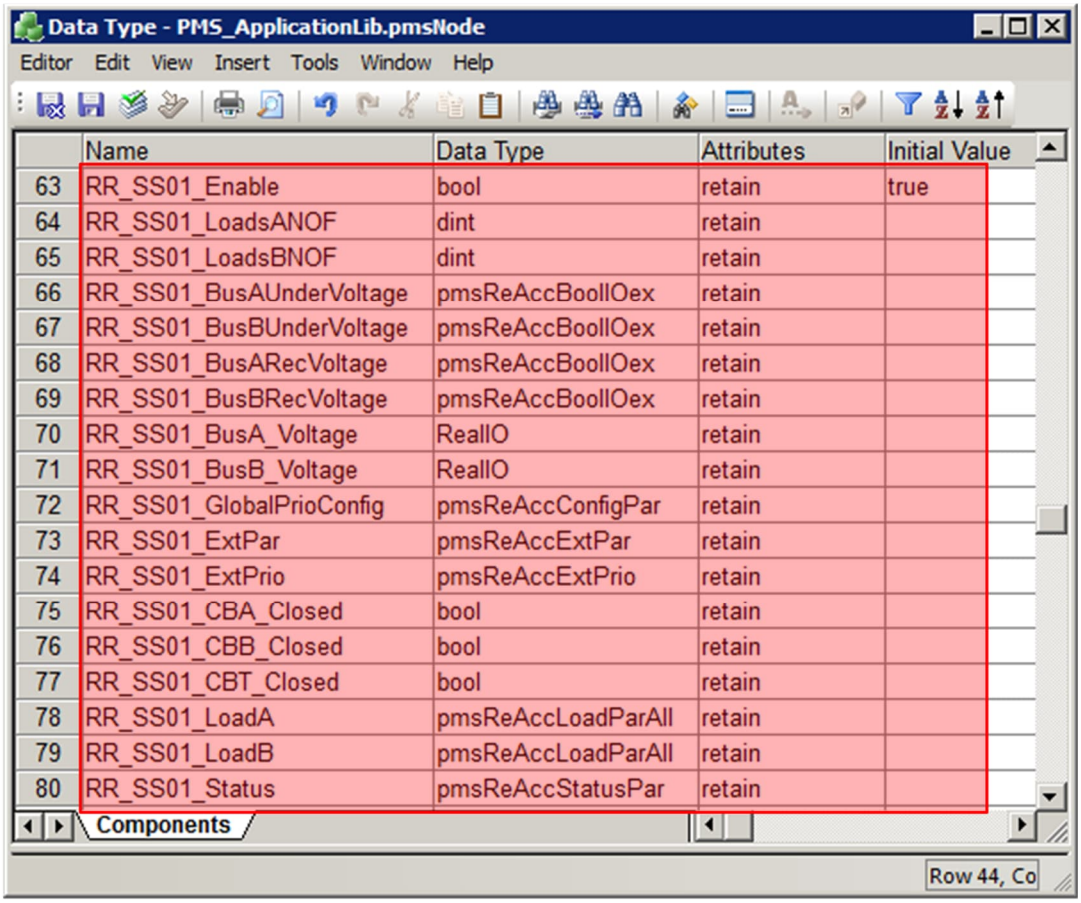
Connections - Restart_Sub01 pmsReAccelLib.pmsReAccSub

Editor Edit View Insert Tools Window Help

	Name	Data Type	Initial Value	Parameter	Attributes	Direction	Description
1	Enabled	bool	Default	pmsNode1.RR_SS01_Enable		in	IN: Enable Function
2	Name	String[30]	'SubstationName'	'SS01_Restart'		in	IN EDIT: Name
3	Description	String[40]	'SubstationDescr'	'SS01_Restart'		in	IN EDIT: Description
4	AlSeverity	dint	900			in	IN EDIT: Alarm severity
5	EvSeverity	dint	500			in	IN EDIT: Event severity
6	Class	dint	50			in	IN EDIT: Alarm and Event Class
7	TaskNormal	string	Default	TaskNormal		in	IN EDIT: Normal task for initial running loads ar
8	BusA	String[10]	'Busbar A'			in	IN EDIT: Text for busbar A
9	BusB	String[10]	'Busbar B'			in	IN EDIT: Text for busbar B
10	FunctionMode	dint	1			in	IN EDIT: [1=Restart][2=Reacceleration]
11	LBBA_ID	dint	1	0		in	IN EDIT: ID of the Load Bus Bar which the subs
12	LBBB_ID	dint	2	0		in	IN EDIT: ID of the Load Bus Bar which the subs
13	LoadsANOF	dint	5	pmsNode1.RR_SS01_LoadsANOF		in	IN EDIT: Number of Loads in Bus A
14	LoadsBNOF	dint	5	pmsNode1.RR_SS01_LoadsBNOF		in	IN EDIT: Number of Loads in Bus B
15	BusAUnderVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusAUnderVoltage		in	IN: Bus A Undervoltage
16	BusBUnderVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusBUnderVoltage		in	IN: Bus B Undervoltage
17	BusARecoveryVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusARecVoltage		in	IN: Bus A recovery voltage level
18	BusBRecoveryVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusBRecVoltage		in	IN: Bus B recovery voltage level
19	BusAVoltage	RealIO	Default	pmsNode1.RR_SS01_BusA_Voltage		in	IN: Bus A Voltage
20	BusBVoltage	RealIO	Default	pmsNode1.RR_SS01_BusB_Voltage		in	IN: Bus B Voltage
21	EnableSecUnderVoltCheck	bool	Default	true		in	IN: Enables Bus A secondary Voltage check.
22	EnableSecRecVoltCheck	bool	Default	true		in	IN: Enables Bus B secondary Voltage check.
23	GlobalPrioConfig	pmsReAccConfigPar	Default	pmsNode1.RR_SS01_GlobalPrioConfig		in_out	IN: Priority Group configuration parameters
24	ExtPar	pmsReAccExtPar	Default	pmsNode1.RR_SS01_ExtPar		in	IN: External parameters
25	ExtPrio	pmsReAccExtPrio	Default	pmsNode1.RR_SS01_ExtPrio		in	IN: External Priority (Future)
26	CBA_C	bool	Default	pmsNode1.RR_SS01_CBA_Closed		in	IN: Circuit breaker SS_A checked closed pos
27	CBB_C	bool	Default	pmsNode1.RR_SS01_CBB_Closed		in	IN: Circuit breaker SS_B checked closed pos
28	CBT_C	bool	Default	pmsNode1.RR_SS01_CBT_Closed		in	IN: Circuit breaker SS_T checked closed pos
29	LoadA	pmsReAccLoadParAll	Default	pmsNode1.RR_SS01_LoadA		in_out	IN_OUT: Load parameters, up to 50 Loads in s
30	LoadB	pmsReAccLoadParAll	Default	pmsNode1.RR_SS01_LoadB		in_out	IN_OUT: Load parameters, up to 50 Loads in s
31	Status	pmsReAccStatusPar	Default	pmsNode1.RR_SS01_Status		out	OUT: Function Status

Figure 31 Parameter connections

These variables must first be created in pmsNode1:



	Name	Data Type	Attributes	Initial Value
63	RR_SS01_Enable	bool	retain	true
64	RR_SS01_LoadsANOF	dint	retain	
65	RR_SS01_LoadsBNOF	dint	retain	
66	RR_SS01_BusAUnderVoltage	pmsReAccBoolIOex	retain	
67	RR_SS01_BusBUnderVoltage	pmsReAccBoolIOex	retain	
68	RR_SS01_BusARecVoltage	pmsReAccBoolIOex	retain	
69	RR_SS01_BusBRecVoltage	pmsReAccBoolIOex	retain	
70	RR_SS01_BusA_Voltage	RealIO	retain	
71	RR_SS01_BusB_Voltage	RealIO	retain	
72	RR_SS01_GlobalPrioConfig	pmsReAccConfigPar	retain	
73	RR_SS01_ExtPar	pmsReAccExtPar	retain	
74	RR_SS01_ExtPrio	pmsReAccExtPrio	retain	
75	RR_SS01_CBA_Closed	bool	retain	
76	RR_SS01_CBB_Closed	bool	retain	
77	RR_SS01_CBT_Closed	bool	retain	
78	RR_SS01_LoadA	pmsReAccLoadParAll	retain	
79	RR_SS01_LoadB	pmsReAccLoadParAll	retain	
80	RR_SS01_Status	pmsReAccStatusPar	retain	

Figure 32 Configuration parameters in pmsNode1

Code must be added to the single control module RestartReacc to set these parameters:

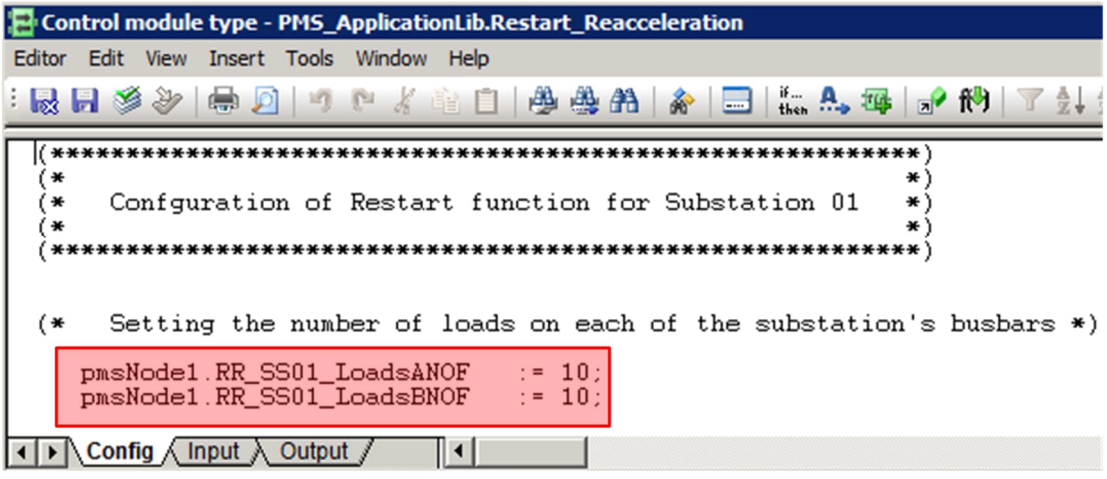


Figure 33 Code for setting configuration parameters

When these parameters are changed, it is sufficient with a warm download to the controller.

The global configuration parameters are here defined in *pmsNode1.Restart_SS01_GlobalPrioConfig*. These consists of one parameter set of global parameters for the substation, and one parameter set for each priority group of the substation (20 priority groups). The parameters of the latter (type pmsReAccGroupPar) are all set in the faceplate:

Data Type - pmsBasicLib.pmsReAccGroupPar [Read-only]

Editor Edit View Insert Tools Window Help

	Name	Data Type	Attributes	Initial Value	Description
1	ReAccGroupDelayTime	time	coldretain nosort	1s	IN: Delay before this priority start
2	OpenCircuit_time	time	coldretain nosort	0.5s	IN: Waiting time before reconnection can be attempted
3	AllowedReaccel_time	time	coldretain nosort	10s	IN: Time from power outage to reacceleration could succeed
4	CloseCMD_time	time	coldretain nosort	1s	IN: Pulse length for close command
5	RunningFb_timeout	time	coldretain nosort	2s	IN: Time from close command until running feedback is expected
6	MaxNrOfAttempts	dint	coldretain nosort	3	IN: Max number of reaccel attempts.
7	Prio	dint	coldretain nosort	1	IN: Load Reaccel/Restart priority
8	Grouped	bool	coldretain nosort	true	IN: Individual settings are decided by Global settings

Figure 34 Priority group parameters – set in the faceplate

The parameters of the global parameters, are partly set in the faceplate, the rest must be set in the code:

Data Type - pmsBasicLib.pmsReAccGlobalPar

Editor Edit View Insert Tools Window Help

	Name	Data Type	Attributes	Initial Value	ISP Value	Description
1	EnableReporting	bool	coldretain nosort	false		IN: Enable reporting function
2	EnableOpReporting	bool	coldretain nosort	false		IN: Enable manual reoporting by operator
3	PrioRestartMethod	dint	coldretain nosort	0		IN EDIT: Static Restart = 0, Dynamic Restart = 1
4	BusbarsConfigured	dint	coldretain nosort	0		IN EDIT: BusbarA & BusbarB = 0 BusbarA = 1, BusbarB = 2
5	CBConfigured	dint	coldretain nosort	0		IN EDIT: CBA,CBB&CBT = 0, CBA&CBB = 1, CBA = 2, CBB = 3
6	MaxPriority	dint	coldretain nosort	20		IN: Max number of priorities used.
7	CoordinateRestart	bool	coldretain nosort	false		IN: Coordinate restart of Busbar A & B
8	HideManualInhibit	bool	coldretain nosort	false		IN: True will hide manual inhibit column
9	HideLoadValue	bool	coldretain nosort	false		IN: True will hide Load value column
10	OperatorSettable	bool	coldretain nosort	true		IN: Operator from changing settings from faceplate
11	VoltageDecayLevel	real	coldretain nosort	0		IN: Voltage decay level in present
12	VoltageRecoveryLevel	real	coldretain nosort	0		IN: Voltage Recovery level in present
13	VoltageRecoveryHyst	time	coldretain nosort	1s		IN: Minimum time for voltage recovery to be stable before restart
14	AllowedVoltRecoveryTime	time	coldretain nosort	20s		IN: Max allowed voltage recovery time before operation is canceled
15	MaxNrVoltageDecays	dint	coldretain nosort	5		IN: Max number of Voltage decays before operation is canceled
16	MaxLimit_IncomerA	real	coldretain nosort	0		IN: Max limit for Incomer A
17	MaxLimit_IncomerB	real	coldretain nosort	0		IN: Max limit for Incomer B
18	ReAccGroupDelayTime	time	coldretain nosort	1s		IN: Delay before this priority start
19	OpenCircuit_time	time	coldretain nosort	0.5s		IN: Waiting time before reconnection can be attempted
20	AllowedReaccel_time	time	coldretain nosort	10s		IN: Time from power outage to reacceleration could succeed
21	CloseCMD_time	time	coldretain nosort	1s		IN: Pulse length for close command
22	RunningFb_timeout	time	coldretain nosort	2s		IN: Time from close command until running feedback is expected
23	MaxNrOfAttempts	dint	coldretain nosort	3		IN: Max number of reaccel attempts.
24	UpdateRunningFbT	time	coldretain nosort	2s		IN: Update rate for scanning running feedback in Idle state
25	SignalTransitionT	time	coldretain nosort	1s		IN: Transition time for undervoltage and recovervoltage signals

Components

Figure 35 Global parameters

5.2.1.1 Input variables – Voltage values

Now the variables for the voltage inputs must be connected:

Connections - Restart_Sub01 pmsReAccelLib.pmsReAccSub

Editor Edit View Insert Tools Window Help

	Name	Data Type	Initial Value	Parameter	Attributes	Direction	Description
1	Enabled	bool	Default	pmsNode1.RR_SS01_Enable		in	IN: Enable Function
2	Name	String[30]	'SubstationName'	'SS01_Restart'		in	IN EDIT: Name
3	Description	String[40]	'SubstationDescr'	'SS01_Restart'		in	IN EDIT: Description
4	AlSeverity	dint	900			in	IN EDIT: Alarm severity
5	EvSeverity	dint	500			in	IN EDIT: Event severity
6	Class	dint	50			in	IN EDIT: Alarm and Event Class
7	TaskNormal	string	Default	TaskNormal		in	IN EDIT: Normal task for initial running loads ar
8	BusA	String[10]	'Busbar A'			in	IN EDIT: Text for busbar A
9	BusB	String[10]	'Busbar B'			in	IN EDIT: Text for busbar B
10	FunctionMode	dint	1			in	IN EDIT: [1=Restart][2=Reacceleration]
11	LBBA_ID	dint	1	0		in	IN EDIT: ID of the Load Bus Bar which the subs
12	LBBB_ID	dint	2	0		in	IN EDIT: ID of the Load Bus Bar which the subs
13	LoadsANOF	dint	5	pmsNode1.RR_SS01_LoadsANOF		in	IN EDIT: Number of Loads in Bus A
14	LoadsBNOF	dint	5	pmsNode1.RR_SS01_LoadsBNOF		in	IN EDIT: Number of Loads in Bus B
15	BusAUnderVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusAUnderVoltage		in	IN: Bus A Undervoltage
16	BusBUnderVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusBUnderVoltage		in	IN: Bus B Undervoltage
17	BusARecoveryVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusARecVoltage		in	IN: Bus A recovery voltage level
18	BusBRecoveryVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusBRecVoltage		in	IN: Bus B recovery voltage level
19	BusAVoltage	RealIO	Default	pmsNode1.RR_SS01_BusA_Voltage		in	IN: Bus A Voltage
20	BusBVoltage	RealIO	Default	pmsNode1.RR_SS01_BusB_Voltage		in	IN: Bus B Voltage
21	EnableSecUnderVoltCheck	bool	Default	true		in	IN: Enables Bus A secondary Voltage check.
22	EnableSecRecVoltCheck	bool	Default	true		in	IN: Enables Bus B secondary Voltage check.
23	GlobalPrioConfig	pmsReAccConfigPar	Default	pmsNode1.RR_SS01_GlobalPrioConfig		in_out	IN: Priority Group configuration parameters
24	ExtPar	pmsReAccExtPar	Default	pmsNode1.RR_SS01_ExtPar		in	IN: External parameters
25	ExtPrio	pmsReAccExtPrio	Default	pmsNode1.RR_SS01_ExtPrio		in	IN: External Priority (Future)
26	CBA_C	bool	Default	pmsNode1.RR_SS01_CBA_Closed		in	IN: Circuit breaker SS_A checked closed pos
27	CBB_C	bool	Default	pmsNode1.RR_SS01_CBB_Closed		in	IN: Circuit breaker SS_B checked closed pos
28	CBT_C	bool	Default	pmsNode1.RR_SS01_CBT_Closed		in	IN: Circuit breaker SS_T checked closed pos
29	LoadA	pmsReAccLoadParAll	Default	pmsNode1.RR_SS01_LoadA		in_out	IN_OUT: Load parameters, up to 50 Loads in s
30	LoadB	pmsReAccLoadParAll	Default	pmsNode1.RR_SS01_LoadB		in_out	IN_OUT: Load parameters, up to 50 Loads in s
31	Status	pmsReAccStatusPar	Default	pmsNode1.RR_SS01_Status		out	OUT: Function Status

Figure 36 Connecting variables for voltage inputs

The related variables must be created in pmsNode1 and ioNode1. The values in ioNode1 are connected to the channels on the IO cards, while the values in pmsNode1 are reading these IO values.

Add these variables to ioNode1:

Data Type - PMS_ApplicationLib.ioNode [Read-only]

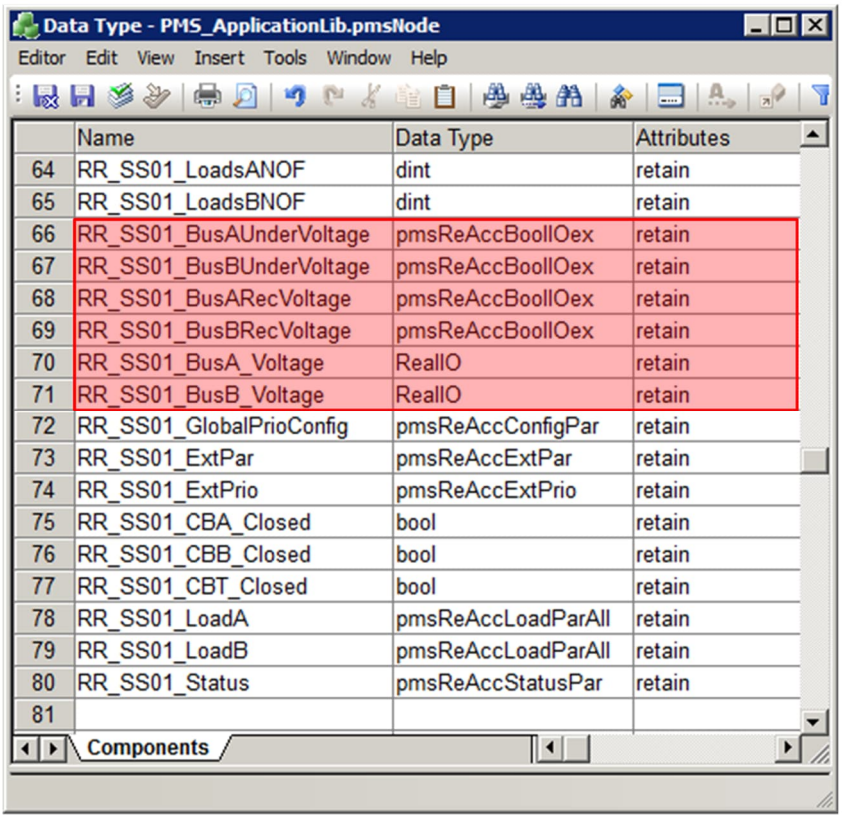
Editor Edit View Insert Tools Window Help

	Name	Data Type	Attributes
89	RR_SS01_BusAUnderVoltage	BoolIO	retain
90	RR_SS01_BusBUnderVoltage	BoolIO	retain
91	RR_SS01_BusARecoveryVoltage	BoolIO	retain
92	RR_SS01_BusBRecoveryVoltage	BoolIO	retain
93	RR_SS01_BusAVoltage	RealIO	retain
94	RR_SS01_BusBVoltage	RealIO	retain

Components

Figure 37 Variables for voltage values in ioNode1

Add these variables to pmsNode1:



	Name	Data Type	Attributes
64	RR_SS01_LoadsANOF	dint	retain
65	RR_SS01_LoadsBNOF	dint	retain
66	RR_SS01_BusAUnderVoltage	pmsReAccBoolIOex	retain
67	RR_SS01_BusBUnderVoltage	pmsReAccBoolIOex	retain
68	RR_SS01_BusARecVoltage	pmsReAccBoolIOex	retain
69	RR_SS01_BusBRecVoltage	pmsReAccBoolIOex	retain
70	RR_SS01_BusA_Voltage	RealIO	retain
71	RR_SS01_BusB_Voltage	RealIO	retain
72	RR_SS01_GlobalPrioConfig	pmsReAccConfigPar	retain
73	RR_SS01_ExtPar	pmsReAccExtPar	retain
74	RR_SS01_ExtPrio	pmsReAccExtPrio	retain
75	RR_SS01_CBA_Closed	bool	retain
76	RR_SS01_CBB_Closed	bool	retain
77	RR_SS01_CBT_Closed	bool	retain
78	RR_SS01_LoadA	pmsReAccLoadParAll	retain
79	RR_SS01_LoadB	pmsReAccLoadParAll	retain
80	RR_SS01_Status	pmsReAccStatusPar	retain
81			

Figure 38 Variables for voltage values in pmsNode1

The digital and analogue voltage values will be read into pmsNode1 from ioNode1; add this code to code block “Input”:

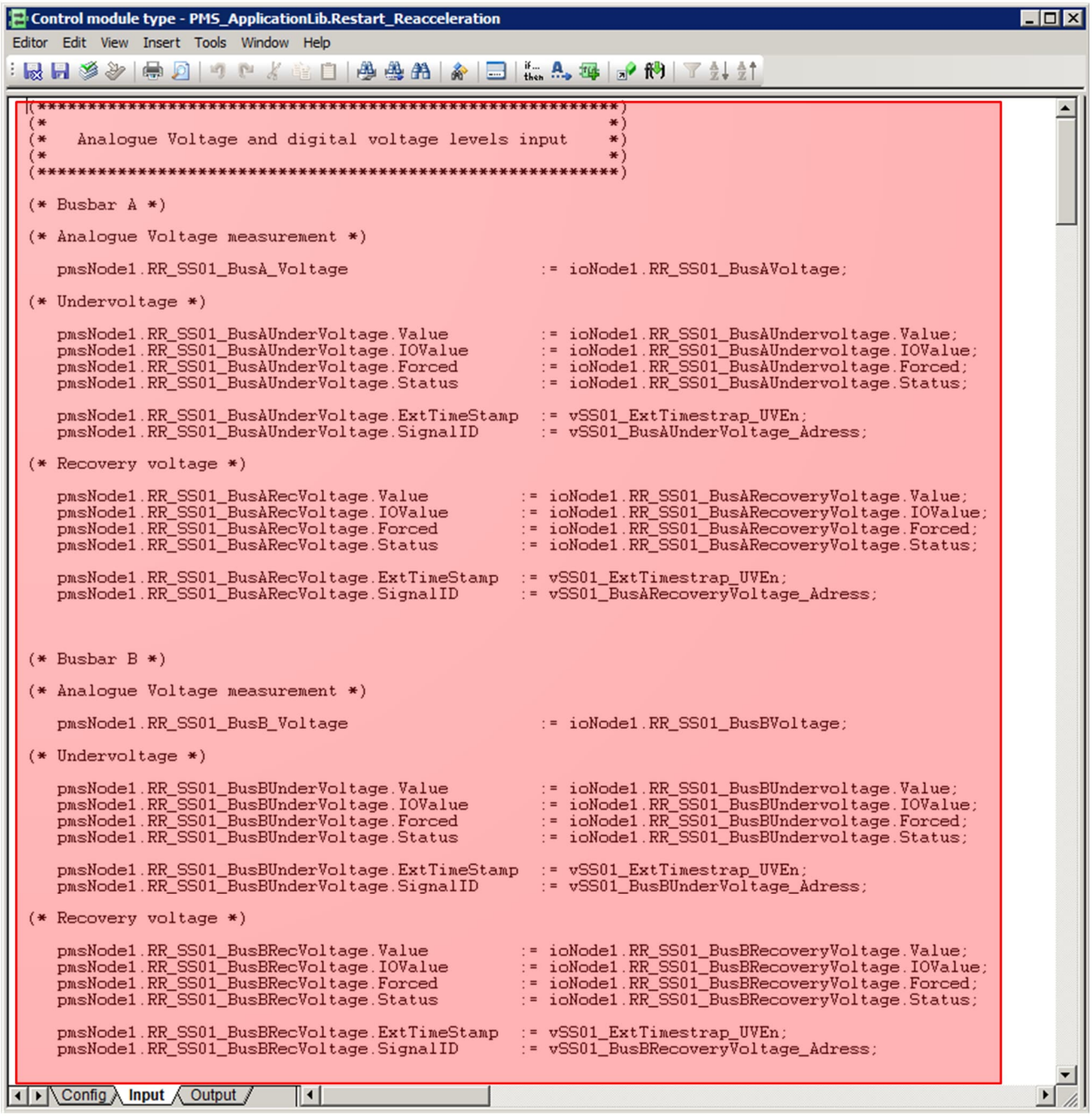


Figure 39 Code for reading voltage inputs

The Restart_SS01_BusAVoltage and Restart_SS01_BusBVoltage have the same type, while the other have different types and their sub-values must then be set separately. Also create these variables, and change them to the correct values for your project:

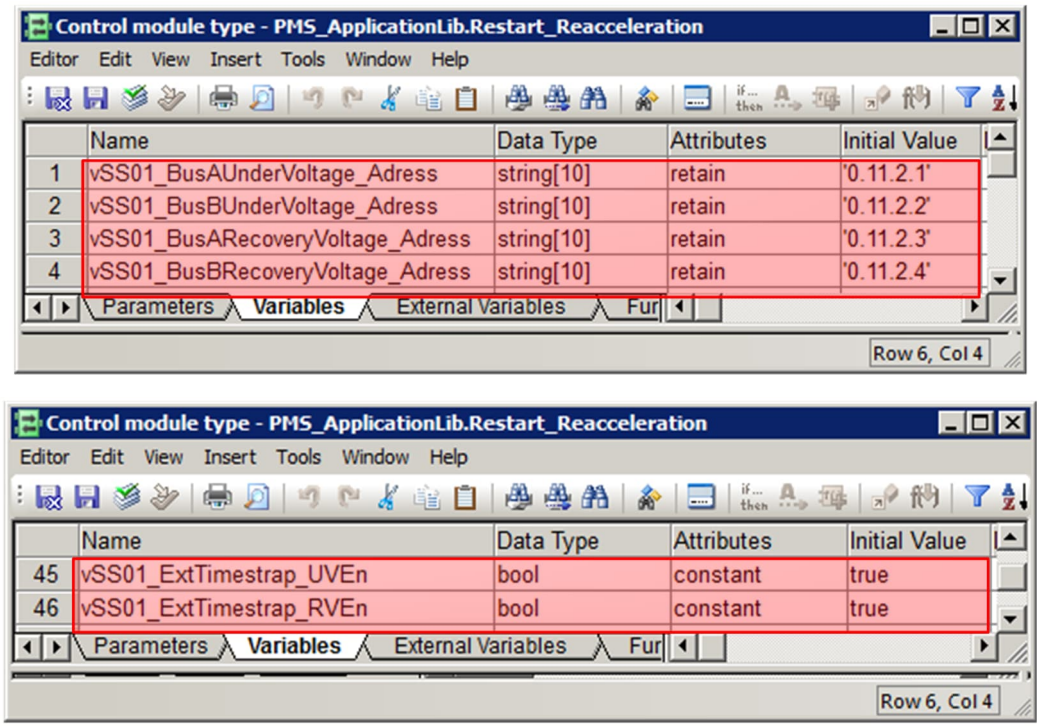


Figure 40 Variables for setting IO values

The string values represent the IO structure for the corresponding IO variable, where the last value represents the channel number on the IO card.

5.2.1.2 Input variables – Circuit breaker status

Create the IO variables for the circuit breaker status that shall be connected to the IO cards:

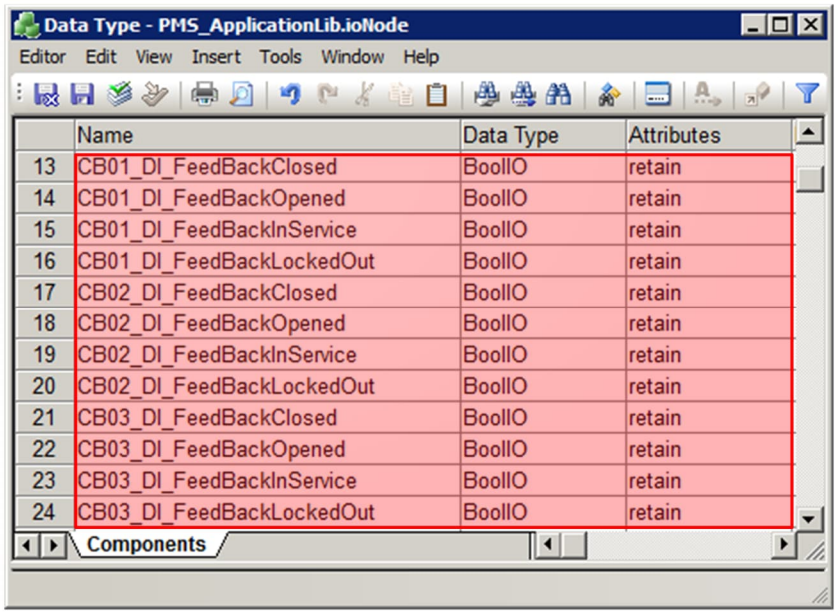
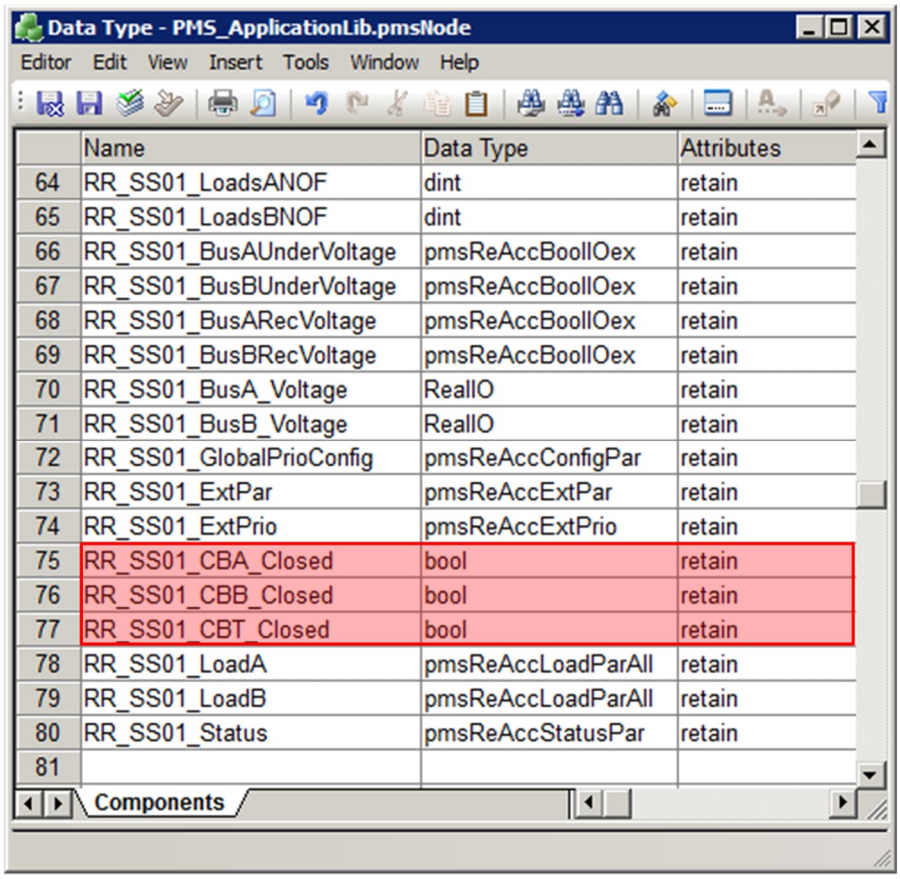


Figure 41 IO variables for reading CB status

Create the closed feedback variables in pmsNode1:



	Name	Data Type	Attributes
64	RR_SS01_LoadsANOF	dint	retain
65	RR_SS01_LoadsBNOF	dint	retain
66	RR_SS01_BusAUnderVoltage	pmsReAccBoolIOex	retain
67	RR_SS01_BusBUnderVoltage	pmsReAccBoolIOex	retain
68	RR_SS01_BusARecVoltage	pmsReAccBoolIOex	retain
69	RR_SS01_BusBRecVoltage	pmsReAccBoolIOex	retain
70	RR_SS01_BusA_Voltage	RealIO	retain
71	RR_SS01_BusB_Voltage	RealIO	retain
72	RR_SS01_GlobalPrioConfig	pmsReAccConfigPar	retain
73	RR_SS01_ExtPar	pmsReAccExtPar	retain
74	RR_SS01_ExtPrio	pmsReAccExtPrio	retain
75	RR_SS01_CBA_Closed	bool	retain
76	RR_SS01_CBB_Closed	bool	retain
77	RR_SS01_CBT_Closed	bool	retain
78	RR_SS01_LoadA	pmsReAccLoadParAll	retain
79	RR_SS01_LoadB	pmsReAccLoadParAll	retain
80	RR_SS01_Status	pmsReAccStatusPar	retain
81			

Figure 42 Variables for CB closed status

Add this code to set the confirmed closed status:

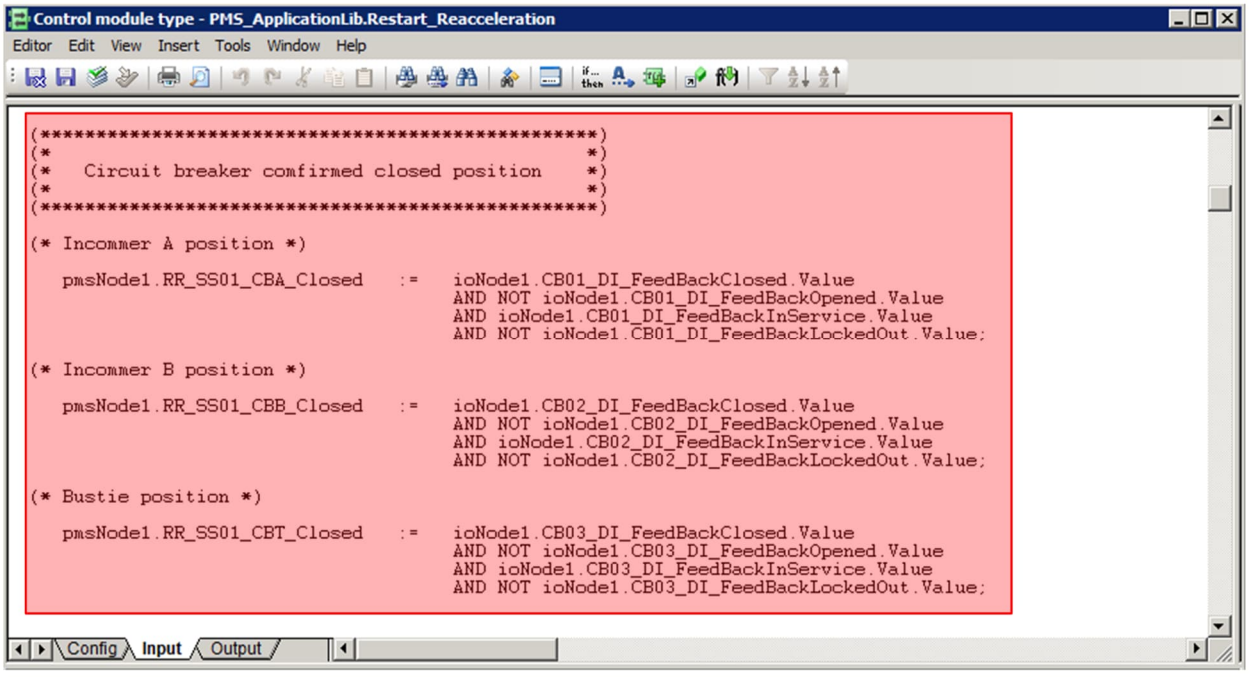
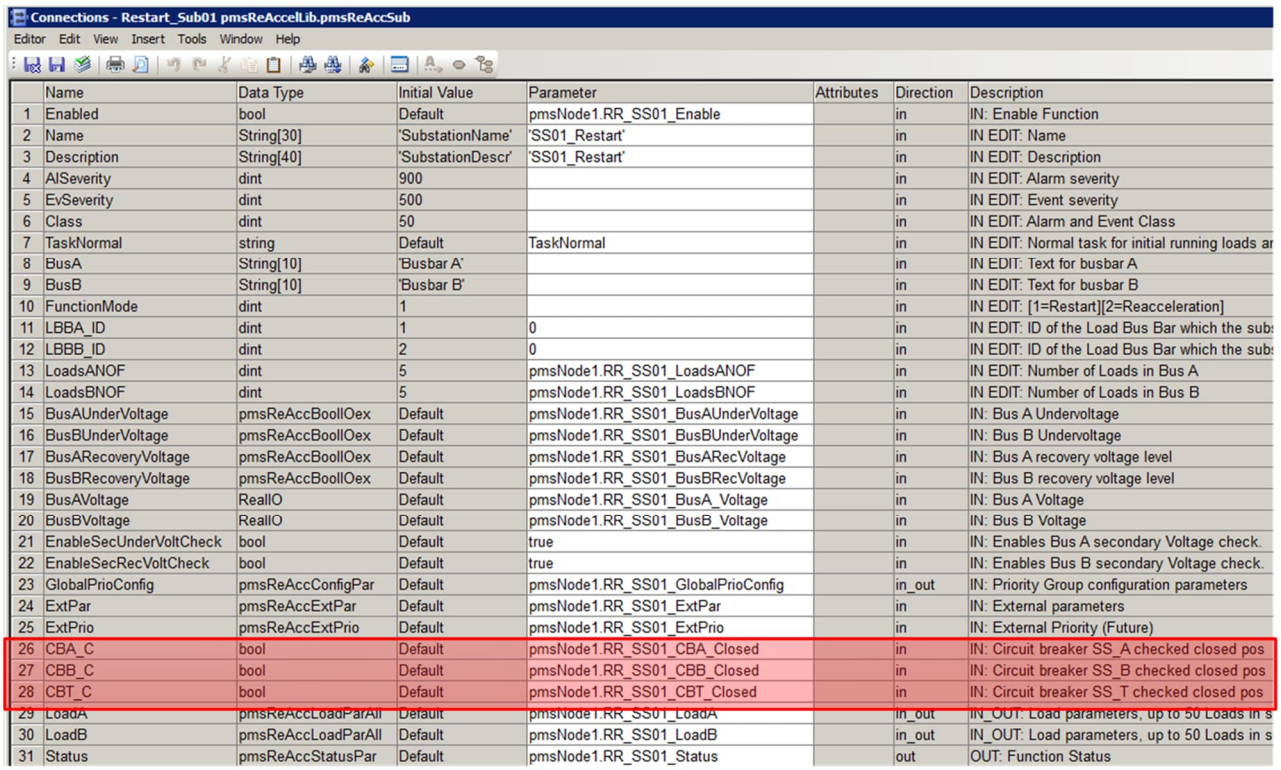


Figure 43 Code for setting CB closed status

Then the status of the circuit breakers can be connected to the Restart_Substation01 instance:



Connections - Restart_Sub01 pmsReAccelLib.pmsReAccSub

Editor Edit View Insert Tools Window Help

	Name	Data Type	Initial Value	Parameter	Attributes	Direction	Description
1	Enabled	bool	Default	pmsNode1.RR_SS01_Enable		in	IN: Enable Function
2	Name	String[30]	'SubstationName'	'SS01_Restart'		in	IN EDIT: Name
3	Description	String[40]	'SubstationDescr'	'SS01_Restart'		in	IN EDIT: Description
4	AlSeverity	dint	900			in	IN EDIT: Alarm severity
5	EvSeverity	dint	500			in	IN EDIT: Event severity
6	Class	dint	50			in	IN EDIT: Alarm and Event Class
7	TaskNormal	string	Default	TaskNormal		in	IN EDIT: Normal task for initial running loads ar
8	BusA	String[10]	'Busbar A'			in	IN EDIT: Text for busbar A
9	BusB	String[10]	'Busbar B'			in	IN EDIT: Text for busbar B
10	FunctionMode	dint	1			in	IN EDIT: [1=Restart][2=Reacceleration]
11	LBBA_ID	dint	1	0		in	IN EDIT: ID of the Load Bus Bar which the sub
12	LBBB_ID	dint	2	0		in	IN EDIT: ID of the Load Bus Bar which the sub
13	LoadsANOF	dint	5	pmsNode1.RR_SS01_LoadsANOF		in	IN EDIT: Number of Loads in Bus A
14	LoadsBNOF	dint	5	pmsNode1.RR_SS01_LoadsBNOF		in	IN EDIT: Number of Loads in Bus B
15	BusAUnderVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusAUnderVoltage		in	IN: Bus A Undervoltage
16	BusBUnderVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusBUnderVoltage		in	IN: Bus B Undervoltage
17	BusARecoveryVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusARecVoltage		in	IN: Bus A recovery voltage level
18	BusBRecoveryVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusBRecVoltage		in	IN: Bus B recovery voltage level
19	BusAVoltage	RealIO	Default	pmsNode1.RR_SS01_BusA_Voltage		in	IN: Bus A Voltage
20	BusBVoltage	RealIO	Default	pmsNode1.RR_SS01_BusB_Voltage		in	IN: Bus B Voltage
21	EnableSecUnderVoltCheck	bool	Default	true		in	IN: Enables Bus A secondary Voltage check.
22	EnableSecRecVoltCheck	bool	Default	true		in	IN: Enables Bus B secondary Voltage check.
23	GlobalPrioConfig	pmsReAccConfigPar	Default	pmsNode1.RR_SS01_GlobalPrioConfig		in_out	IN: Priority Group configuration parameters
24	ExtPar	pmsReAccExtPar	Default	pmsNode1.RR_SS01_ExtPar		in	IN: External parameters
25	ExtPrio	pmsReAccExtPrio	Default	pmsNode1.RR_SS01_ExtPrio		in	IN: External Priority (Future)
26	CBA_C	bool	Default	pmsNode1.RR_SS01_CBA_Closed		in	IN: Circuit breaker SS_A checked closed pos
27	CBB_C	bool	Default	pmsNode1.RR_SS01_CBB_Closed		in	IN: Circuit breaker SS_B checked closed pos
28	CBT_C	bool	Default	pmsNode1.RR_SS01_CBT_Closed		in	IN: Circuit breaker SS_T checked closed pos
29	LoadA	pmsReAccLoadParAll	Default	pmsNode1.RR_SS01_LoadA		in_out	IN_OUT: Load parameters, up to 50 Loads in s
30	LoadB	pmsReAccLoadParAll	Default	pmsNode1.RR_SS01_LoadB		in_out	IN_OUT: Load parameters, up to 50 Loads in s
31	Status	pmsReAccStatusPar	Default	pmsNode1.RR_SS01_Status		out	OUT: Function Status

Figure 44 Connecting values for CB closed status

5.2.1.1 Connection of LoadA and LoadB

Create these variables in pmsNode1 and ioNode1:

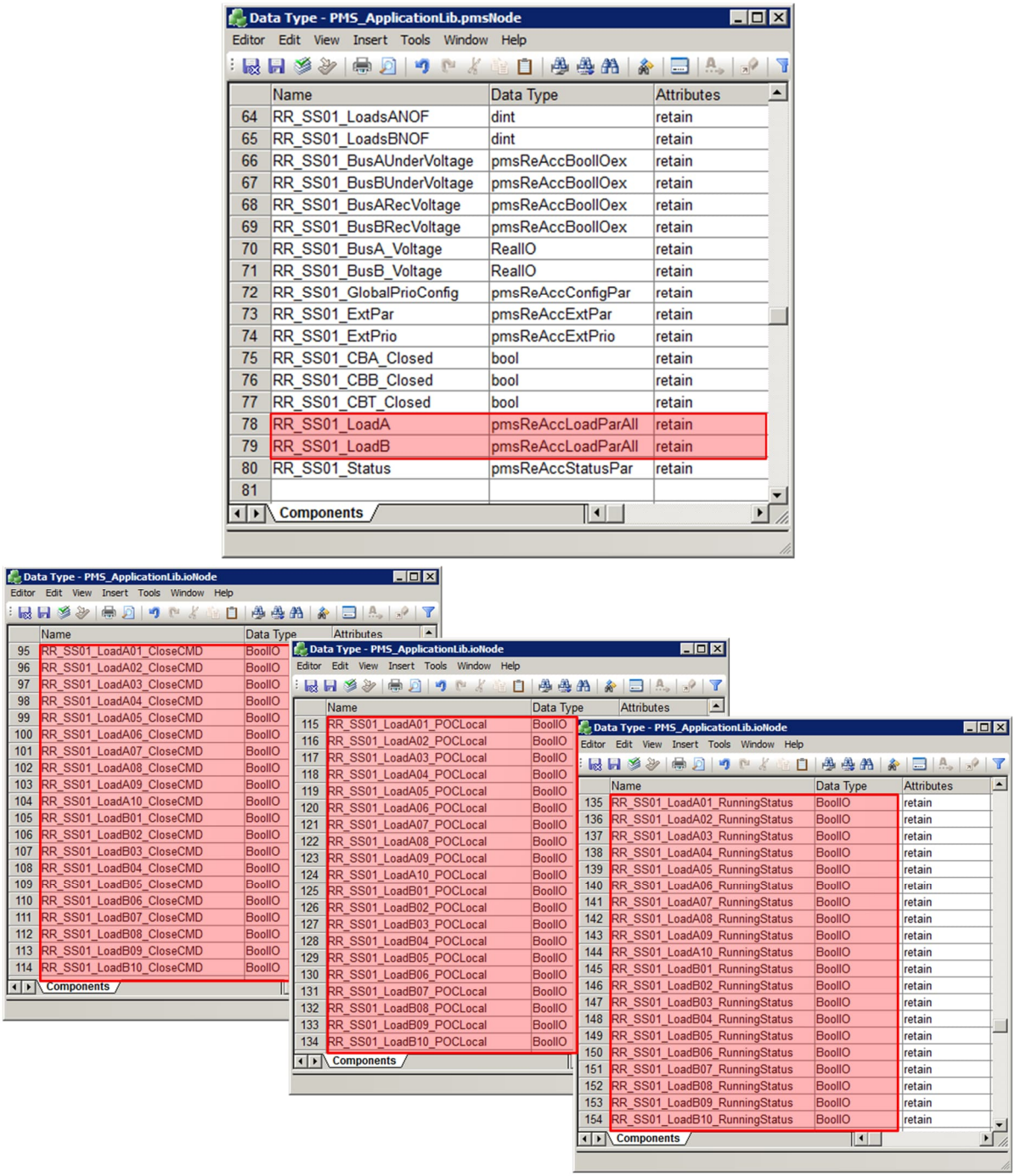
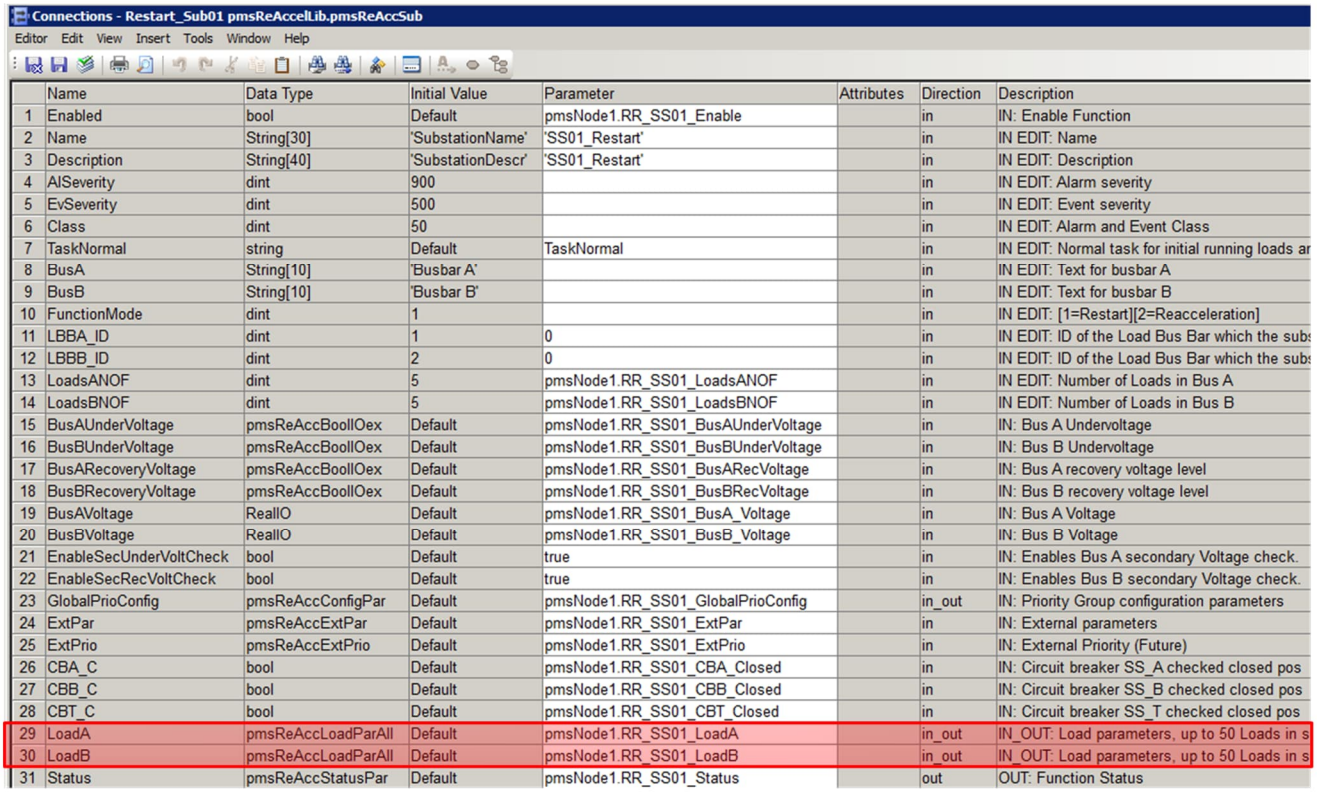


Figure 45 Variables to be added in pmsNode1 and ioNode1

In Figure 45 is shown the IO variables for 20 loads, 10 on A and 10 on B. Connect them for the instance:



Connections - Restart_Sub01 pmsReAccelLib.pmsReAccSub

Editor Edit View Insert Tools Window Help

	Name	Data Type	Initial Value	Parameter	Attributes	Direction	Description
1	Enabled	bool	Default	pmsNode1.RR_SS01_Enable		in	IN: Enable Function
2	Name	String[30]	'SubstationName'	'SS01_Restart'		in	IN EDIT: Name
3	Description	String[40]	'SubstationDescr'	'SS01_Restart'		in	IN EDIT: Description
4	AlSeverity	dint	900			in	IN EDIT: Alarm severity
5	EvSeverity	dint	500			in	IN EDIT: Event severity
6	Class	dint	50			in	IN EDIT: Alarm and Event Class
7	TaskNormal	string	Default	TaskNormal		in	IN EDIT: Normal task for initial running loads ar
8	BusA	String[10]	'Busbar A'			in	IN EDIT: Text for busbar A
9	BusB	String[10]	'Busbar B'			in	IN EDIT: Text for busbar B
10	FunctionMode	dint	1			in	IN EDIT: [1=Restart][2=Reacceleration]
11	LBBA_ID	dint	1	0		in	IN EDIT: ID of the Load Bus Bar which the subs
12	LBBB_ID	dint	2	0		in	IN EDIT: ID of the Load Bus Bar which the subs
13	LoadsANOF	dint	5	pmsNode1.RR_SS01_LoadsANOF		in	IN EDIT: Number of Loads in Bus A
14	LoadsBNOF	dint	5	pmsNode1.RR_SS01_LoadsBNOF		in	IN EDIT: Number of Loads in Bus B
15	BusAUnderVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusAUnderVoltage		in	IN: Bus A Undervoltage
16	BusBUnderVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusBUnderVoltage		in	IN: Bus B Undervoltage
17	BusARecoveryVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusARecVoltage		in	IN: Bus A recovery voltage level
18	BusBRecoveryVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusBRecVoltage		in	IN: Bus B recovery voltage level
19	BusAVoltage	ReallO	Default	pmsNode1.RR_SS01_BusA_Voltage		in	IN: Bus A Voltage
20	BusBVoltage	ReallO	Default	pmsNode1.RR_SS01_BusB_Voltage		in	IN: Bus B Voltage
21	EnableSecUnderVoltCheck	bool	Default	true		in	IN: Enables Bus A secondary Voltage check.
22	EnableSecRecVoltCheck	bool	Default	true		in	IN: Enables Bus B secondary Voltage check.
23	GlobalPrioConfig	pmsReAccConfigPar	Default	pmsNode1.RR_SS01_GlobalPrioConfig		in_out	IN: Priority Group configuration parameters
24	ExtPar	pmsReAccExtPar	Default	pmsNode1.RR_SS01_ExtPar		in	IN: External parameters
25	ExtPrio	pmsReAccExtPrio	Default	pmsNode1.RR_SS01_ExtPrio		in	IN: External Priority (Future)
26	CBA_C	bool	Default	pmsNode1.RR_SS01_CBA_Closed		in	IN: Circuit breaker SS_A checked closed pos
27	CBB_C	bool	Default	pmsNode1.RR_SS01_CBB_Closed		in	IN: Circuit breaker SS_B checked closed pos
28	CBT_C	bool	Default	pmsNode1.RR_SS01_CBT_Closed		in	IN: Circuit breaker SS_T checked closed pos
29	LoadA	pmsReAccLoadParAll	Default	pmsNode1.RR_SS01_LoadA		in_out	IN_OUT: Load parameters, up to 50 Loads in s
30	LoadB	pmsReAccLoadParAll	Default	pmsNode1.RR_SS01_LoadB		in_out	IN_OUT: Load parameters, up to 50 Loads in s
31	Status	pmsReAccStatusPar	Default	pmsNode1.RR_SS01_Status		out	OUT: Function Status

Figure 46 Connecting the LoadPar to the instance

5.2.1.1 Input/output variables – Load status/POCLoc/Close_CMD

The status from the loads, RunningStatus and POCLoc (local point of control), must be read from IO variables into pmsNode1 dataset:

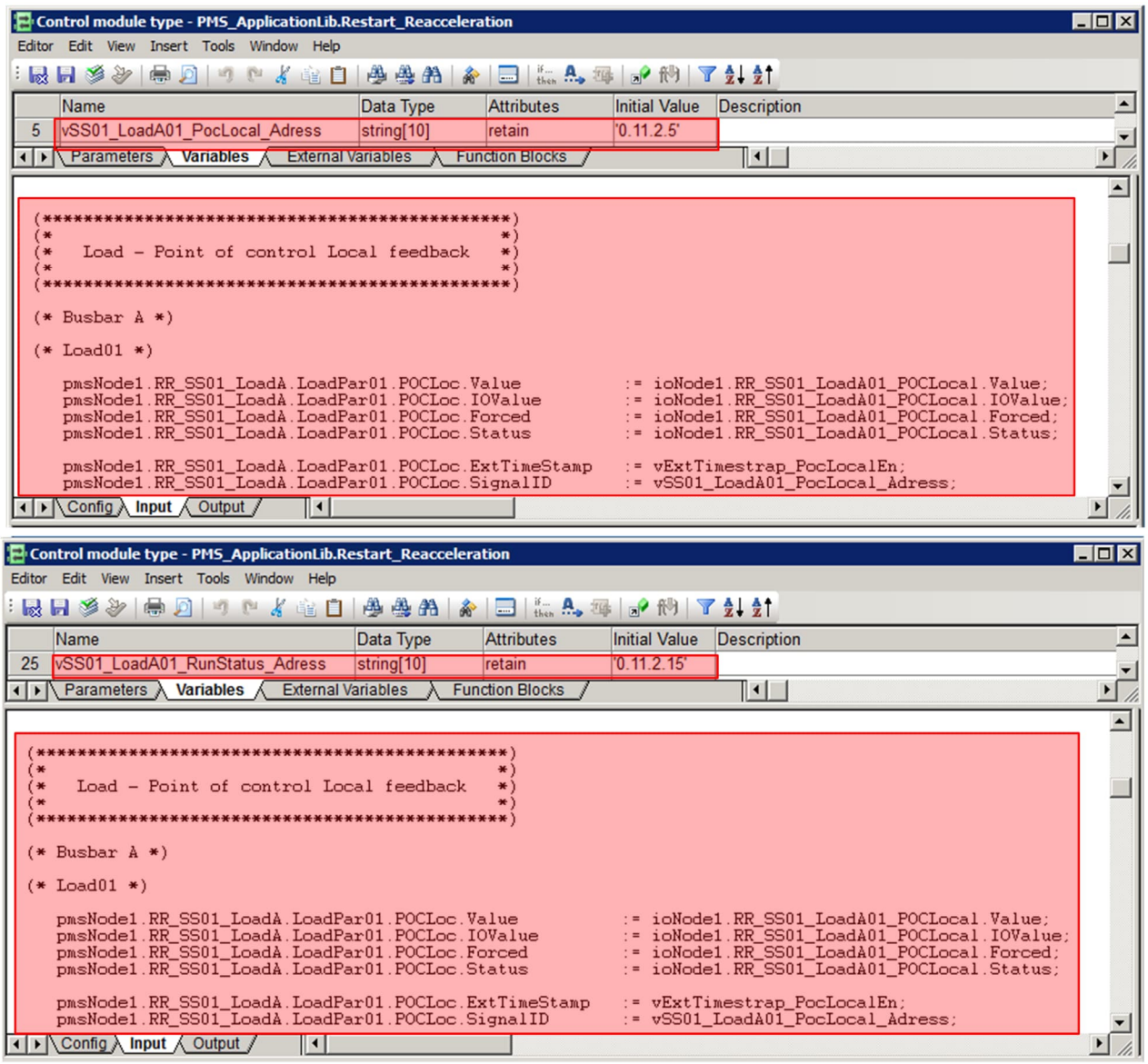


Figure 47 Reading Load status values

In Figure 47 is shown code for LoadA01, similar code must be added for LoadA02-10 and LoadB01-10. The necessary variables that must be created for these inputs are also shown in the figure. The close commands to the loads should be set in the Output code block (the variables in pmsNode1 and ioNode1 are of the same type (BoolIO)):

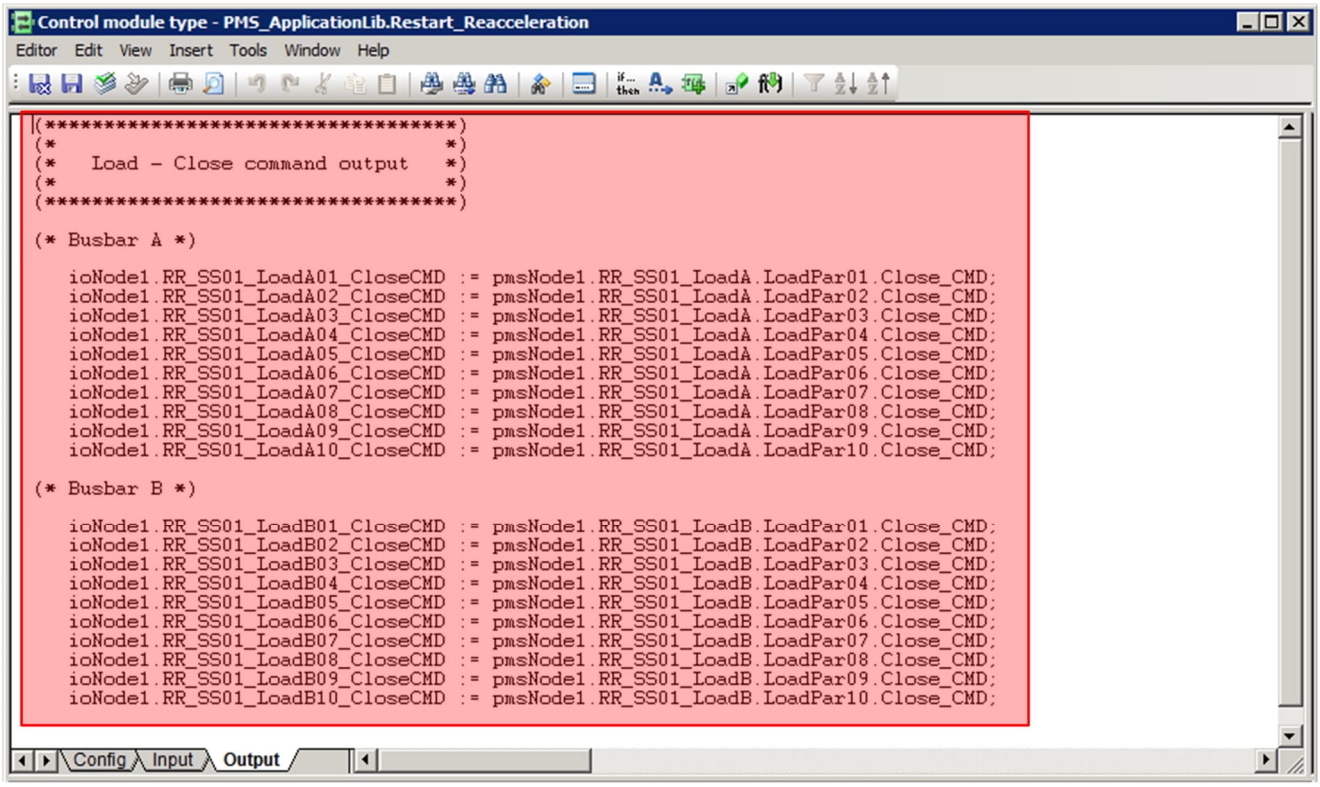
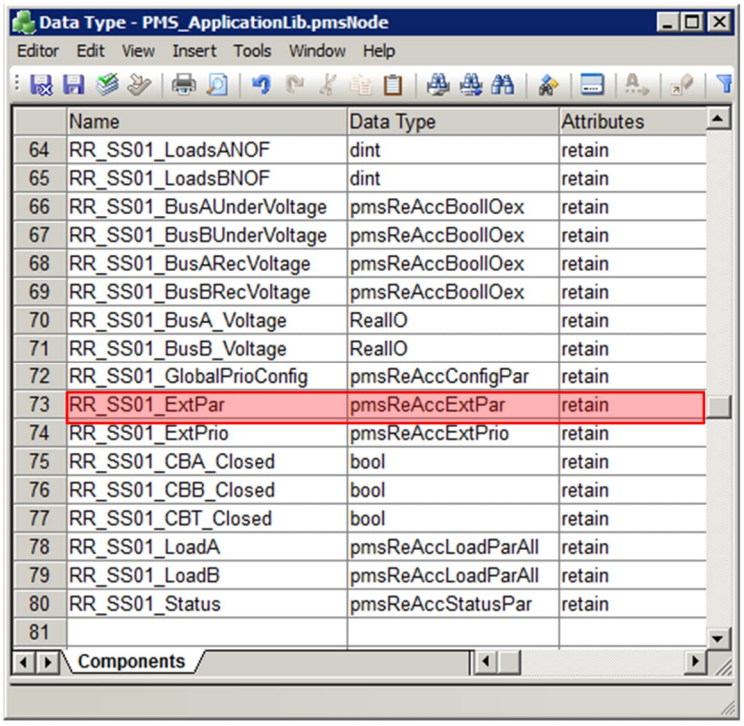


Figure 48 Setting Close command outputs

5.2.1.1 Connection of ExtPar

Create this variable in pmsNode1:



	Name	Data Type	Attributes
64	RR_SS01_LoadsANOF	dint	retain
65	RR_SS01_LoadsBNOF	dint	retain
66	RR_SS01_BusAUnderVoltage	pmsReAccBoolIOex	retain
67	RR_SS01_BusBUnderVoltage	pmsReAccBoolIOex	retain
68	RR_SS01_BusARecVoltage	pmsReAccBoolIOex	retain
69	RR_SS01_BusBRecVoltage	pmsReAccBoolIOex	retain
70	RR_SS01_BusA_Voltage	RealIO	retain
71	RR_SS01_BusB_Voltage	RealIO	retain
72	RR_SS01_GlobalPrioConfig	pmsReAccConfigPar	retain
73	RR_SS01_ExtPar	pmsReAccExtPar	retain
74	RR_SS01_ExtPrio	pmsReAccExtPrio	retain
75	RR_SS01_CBA_Closed	bool	retain
76	RR_SS01_CBB_Closed	bool	retain
77	RR_SS01_CBT_Closed	bool	retain
78	RR_SS01_LoadA	pmsReAccLoadParAll	retain
79	RR_SS01_LoadB	pmsReAccLoadParAll	retain
80	RR_SS01_Status	pmsReAccStatusPar	retain
81			

Figure 49 Variable for external parameter.

Then connect the variable to the instance:

Connections - Restart_Sub01 pmsReAccelLib.pmsReAccSub

Editor Edit View Insert Tools Window Help

	Name	Data Type	Initial Value	Parameter	Attributes	Direction	Description
1	Enabled	bool	Default	pmsNode1.RR_SS01_Enable		in	IN: Enable Function
2	Name	String[30]	'SubstationName'	'SS01_Restart'		in	IN EDIT: Name
3	Description	String[40]	'SubstationDescr'	'SS01_Restart'		in	IN EDIT: Description
4	AlSeverity	dint	900			in	IN EDIT: Alarm severity
5	EvSeverity	dint	500			in	IN EDIT: Event severity
6	Class	dint	50			in	IN EDIT: Alarm and Event Class
7	TaskNormal	string	Default	TaskNormal		in	IN EDIT: Normal task for initial running loads ar
8	BusA	String[10]	'Busbar A'			in	IN EDIT: Text for busbar A
9	BusB	String[10]	'Busbar B'			in	IN EDIT: Text for busbar B
10	FunctionMode	dint	1			in	IN EDIT: [1=Restart][2=Reacceleration]
11	LBBA_ID	dint	1	0		in	IN EDIT: ID of the Load Bus Bar which the subs
12	LBBB_ID	dint	2	0		in	IN EDIT: ID of the Load Bus Bar which the subs
13	LoadsANOF	dint	5	pmsNode1.RR_SS01_LoadsANOF		in	IN EDIT: Number of Loads in Bus A
14	LoadsBNOF	dint	5	pmsNode1.RR_SS01_LoadsBNOF		in	IN EDIT: Number of Loads in Bus B
15	BusAUnderVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusAUnderVoltage		in	IN: Bus A Undervoltage
16	BusBUnderVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusBUnderVoltage		in	IN: Bus B Undervoltage
17	BusARecoveryVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusARecVoltage		in	IN: Bus A recovery voltage level
18	BusBRecoveryVoltage	pmsReAccBoolIOex	Default	pmsNode1.RR_SS01_BusBRecVoltage		in	IN: Bus B recovery voltage level
19	BusAVoltage	RealIO	Default	pmsNode1.RR_SS01_BusA_Voltage		in	IN: Bus A Voltage
20	BusBVoltage	RealIO	Default	pmsNode1.RR_SS01_BusB_Voltage		in	IN: Bus B Voltage
21	EnableSecUnderVoltCheck	bool	Default	true		in	IN: Enables Bus A secondary Voltage check.
22	EnableSecRecVoltCheck	bool	Default	true		in	IN: Enables Bus B secondary Voltage check.
23	GlobalPrioConfig	pmsReAccConfigPar	Default	pmsNode1.RR_SS01_GlobalPrioConfig		in_out	IN: Priority Group configuration parameters
24	ExtPar	pmsReAccExtPar	Default	pmsNode1.RR_SS01_ExtPar		in	IN: External parameters
25	ExtPrio	pmsReAccExtPrio	Default	pmsNode1.RR_SS01_ExtPrio		in	IN: External Priority (Future)
26	CBA_C	bool	Default	pmsNode1.RR_SS01_CBA_Closed		in	IN: Circuit breaker SS_A checked closed pos
27	CBB_C	bool	Default	pmsNode1.RR_SS01_CBB_Closed		in	IN: Circuit breaker SS_B checked closed pos
28	CBT_C	bool	Default	pmsNode1.RR_SS01_CBT_Closed		in	IN: Circuit breaker SS_T checked closed pos
29	LoadA	pmsReAccLoadParAll	Default	pmsNode1.RR_SS01_LoadA		in_out	IN_OUT: Load parameters, up to 50 Loads in s
30	LoadB	pmsReAccLoadParAll	Default	pmsNode1.RR_SS01_LoadB		in_out	IN_OUT: Load parameters, up to 50 Loads in s
31	Status	pmsReAccStatusPar	Default	pmsNode1.RR_SS01_Status		out	OUT: Function Status

Figure 50 Connection of the external parameter variable.

The code for setting the external parameters is placed in the code block “Input”:

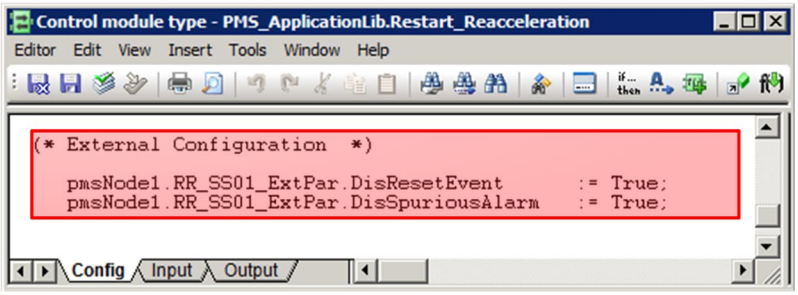


Figure 51 Setting external parameters from code or other inputs

The purpose of disabling events is to reduce the load on the controller during possible event bursts when many loads start at the same time, refer section 4.2

5.2.1.2 Setting parameters from application code

To set parameters from application code, the *GlobalPrioConfig.Global.OperatorSettable* must be set to false, as described in section 3.1.15. This dims certain parameter fields in the faceplate and then prevents these parameters to be changed by the operator.

Load parameters, global parameters and priority parameters can then be set by application code.

The load parameters that can be changed are shown in Figure 52:

Data Type - pmsBasicLib.pmsReAccLoadPar [Read-only]

Editor Edit View Insert Tools Window Help

	Name	Data Type	Attributes	Initial Value	ISP Value	Description
1	SrcName	string[30]	coldretain nosort	'Load01'		IN: Load Name
2	SrcDesc	string[60]	coldretain nosort	'Load01 desc'		IN: Load Description
3	ExternalSettings	bool	coldretain nosort	false		IN: External load settings
4	OpenCircuit_time	time	coldretain nosort	0.2s		IN: Waiting time before reconnection can be attempted
5	AllowedReaccel_time	time	coldretain nosort	10s		IN: Time from power outage to reacceleration could succeed
6	CloseCMD_time	time	coldretain nosort	1s		IN: Pulse length for close command
7	RunningFb_timeout	time	coldretain nosort	2s		IN: Time from close command until running feedback is expected
8	MaxNrOfAttempts	dint	coldretain nosort	3		IN: Max number of reaccel attempts.
9	PLoad	real	coldretain nosort	1.0		IN: Load value (MW,Current,In rush current)
10	Grouped	bool	coldretain nosort	true		IN: Individual settings are decided by Group settings
11	Prio	dint	coldretain nosort	1		IN: Load Reaccel/Restart priority
12	ExtPrioSel	bool	nosort	false		IN: External Reaccel/Restart priority selected
13	ExternalInhibit	bool	nosort	false		IN: External Inhibit
14	POCLoc	pmsReAccBoolIOex	nosort			IN: Circuit breaker point of control
15	RunningStatus	pmsReAccBoolIOex	nosort			IN: Load running status.
16	Close_CMD	BoolIO	nosort			OUT: Reaccel/Restart command
17	Disabled	bool	nosort	false		OUT: Load status, load is disabled for reaccel/restart
18	Signal_Err	bool	nosort	false		OUT: Load signal error indication
19	FailedReaccel	bool	nosort	false		OUT: Failed reaccel
20	InhibitStatus	bool	nosort	false		OUT: AutoInhibit\Operator Inhibit Active

Components

Figure 52 Load parameters that can be set by application code/external

To change the Prio from code, first set the parameter *ExtPrioSel* to true. The other parameters marked as red in Figure 52 can be set directly.

Global parameters; the global parameters are shown in table below.

Data Type - pmsBasicLib.pmsReAccGlobalPar

Editor Edit View Insert Tools Window Help

	Name	Data Type	Attributes	Initial Value	ISP Value	Description
1	EnableReporting	bool	coldretain nosort	false		IN: Enable reporting function
2	EnableOpReporting	bool	coldretain nosort	false		IN: Enable manual reoporting by operator
3	PrioRestartMethod	dint	coldretain nosort	0		IN EDIT: Static Restart = 0, Dynamic Restart = 1
4	BusbarsConfigured	dint	coldretain nosort	0		IN EDIT: BusbarA & BusbarB = 0 BusbarA = 1, BusbarB = 2
5	CBConfigured	dint	coldretain nosort	0		IN EDIT: CBA,CBB&CBT = 0, CBA&CBB = 1, CBA = 2, CBB = 3
6	MaxPriority	dint	coldretain nosort	20		IN: Max number of priorities used.
7	CoordinateRestart	bool	coldretain nosort	false		IN: Coordinate restart of Busbar A & B
8	HideManualInhibit	bool	coldretain nosort	false		IN: True will hide manual inhibit column
9	HideLoadValue	bool	coldretain nosort	false		IN: True will hide Load value column
10	OperatorSettable	bool	coldretain nosort	true		IN: Operator from changing settings from faceplate
11	VoltageDecayLevel	real	coldretain nosort	0		IN: Voltage decay level in present
12	VoltageRecoveryLevel	real	coldretain nosort	0		IN: Voltage Recovery level in present
13	VoltageRecoveryHyst	time	coldretain nosort	1s		IN: Minimum time for voltage recovery to be stable before restart
14	AllowedVoltRecoveryTime	time	coldretain nosort	20s		IN: Max allowed voltage recovery time before operation is canceled
15	MaxNrVoltageDecays	dint	coldretain nosort	5		IN: Max number of Voltage decays before operation is canceled
16	MaxLimit_IncomerA	real	coldretain nosort	0		IN: Max limit for Incomer A
17	MaxLimit_IncomerB	real	coldretain nosort	0		IN: Max limit for Incomer B
18	ReAccGroupDelayTime	time	coldretain nosort	1s		IN: Delay before this priority start
19	OpenCircuit_time	time	coldretain nosort	0.5s		IN: Waiting time before reconnection can be attempted
20	AllowedReaccel_time	time	coldretain nosort	10s		IN: Time from power outage to reacceleration could succeed
21	CloseCMD_time	time	coldretain nosort	1s		IN: Pulse length for close command
22	RunningFb_timeout	time	coldretain nosort	2s		IN: Time from close command until running feedback is expected
23	MaxNrOfAttempts	dint	coldretain nosort	3		IN: Max number of reaccel attempts.
24	UpdateRunningFbT	time	coldretain nosort	2s		IN: Update rate for scanning running feedback in Idle state
25	SignalTransitionT	time	coldretain nosort	1s		IN: Transition time for undervoltage and recovervoltage signals

Components

Figure 53 Global parameters

The priority parameters for each priority group are shown in table below.

Data Type - pmsBasicLib.pmsReAccGroupPar [Read-only]

Editor Edit View Insert Tools Window Help

	Name	Data Type	Attributes	Initial Value	Description
1	ReAccGroupDelayTime	time	coldretain nosort	1s	IN: Delay before this priority start
2	OpenCircuit_time	time	coldretain nosort	0.5s	IN: Waiting time before reconnection can be attempted
3	AllowedReaccel_time	time	coldretain nosort	10s	IN: Time from power outage to reacceleration could succeed
4	CloseCMD_time	time	coldretain nosort	1s	IN: Pulse length for close command
5	RunningFb_timeout	time	coldretain nosort	2s	IN: Time from close command until running feedback is expected
6	MaxNrOfAttempts	dint	coldretain nosort	3	IN: Max number of reaccel attempts.
7	Prio	dint	coldretain nosort	1	IN: Load Reaccel/Restart priority
8	Grouped	bool	coldretain nosort	true	IN: Individual settings are decided by Global settings

Figure 54 Priority parameters

An example of setting some parameters by application code is given Figure 55.

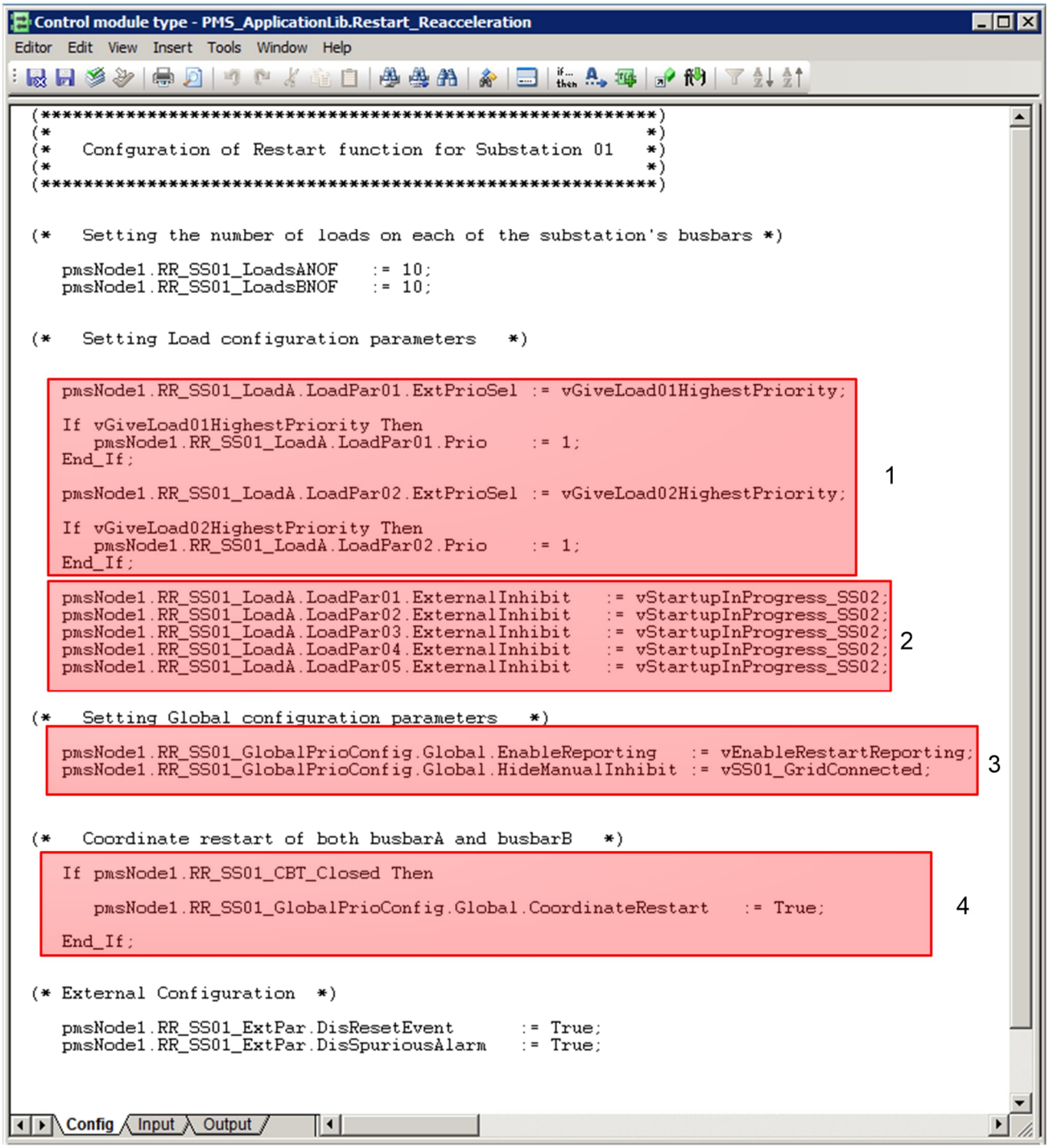


Figure 55 Example of code for settings parameters

Table 28 Examples of parameters set by application code

Field	Description
1	Setting this variable true disables operator changes of certain values in the faceplate.
2	This example sets the priority of Load01 and Load02 to the highest for a certain condition.
3	Load01-05 are inhibited for start-up for a certain condition at another substation.
4	Reporting is enabled by a global variable. The manual inhibit field is hidden for substation 1 for a certain condition.
5	The time delay of priority group 1 of substation 1 is changed for a certain condition.

Appendix A Interaction & Permissions

Table 29 User interaction permissions

Name	Permission	Faceplate element
Enable	Operate	Main
Disable	Operate	Main
Reset	Operate	Main
Block bus A	Operate	Main
Block bus B	Operate	Main
Priority	Tune	Busbar A\B
Manual Inhibit	Tune	Busbar A\B
Inherit priority group settings	Configure	Load settings
Priority	Tune	Load settings
Open circuit time constant	Configure	Load settings
Max allowed restart time	Configure	Load settings
Running feedback time	Configure	Load settings
Close command pulse time	Configure	Load settings
Max no. Of restart attempts	Configure	Load settings
Voltage recovery level	Configure	Global
Voltage decay level	Configure	Global
Voltage recovery hysteresis	Configure	Global
Allowed voltage recovery time	Configure	Global
Max no. Of voltage decays	Configure	Global
(Priority group) Restart delay	Configure	Global
(Load) Open circuit time constant	Configure	Global
(Load) Max allowed restart time	Configure	Global
(Load) Running feedback time	Configure	Global
(Load) Close command pulse time	Configure	Global
(Load) Max no. of restart attempts	Configure	Global

Appendix A Revision table

Rev. ind.	***Page (P) / Chapter ©***	***Description***	***Date***
A		Initial document release for 5.6-5	2022.08.30/KAA

Contact us

ABB AS
PA Energy Industries
Fornebu, Norway
E-mail: power.management@no.abb.com
ABB Oil and Gas

3BNP100234-0396 A

Power and productivity
for a better world™