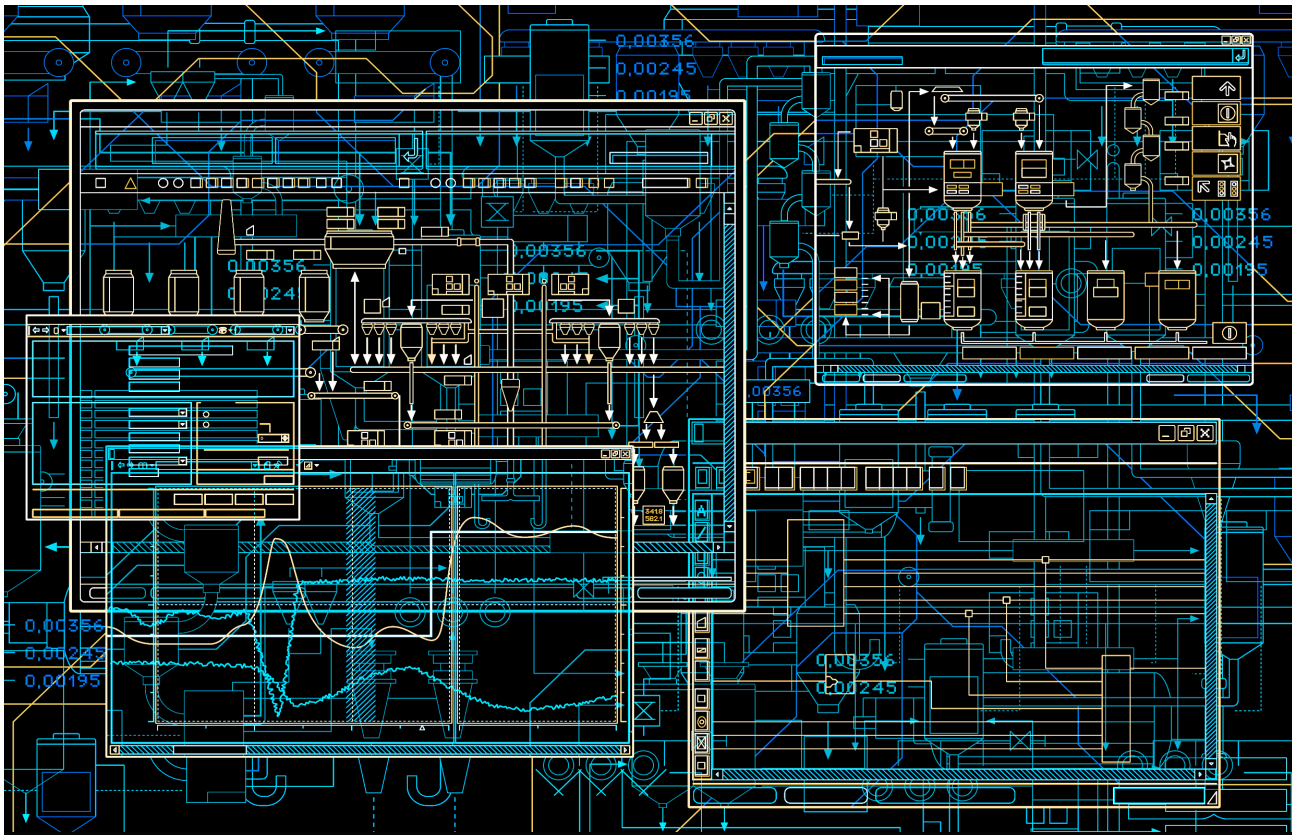


800xA for Advant Master

Configuration

System Version 6.0

Power and productivity
for a better world™

800xA for Advant Master

Configuration

System Version 6.0

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.

Products described or referenced in this document are designed to be connected, and to communicate information and data via a secure network. It is the sole responsibility of the system/product owner to provide and continuously ensure a secure connection between the product and the system network and/or any other networks that may be connected.

The system/product owners must establish and maintain appropriate measures, including, but not limited to, the installation of firewalls, application of authentication measures, encryption of data, installation of antivirus programs, and so on, to protect the system, its products and networks, against security breaches, unauthorized access, interference, intrusion, leakage, and/or theft of data or information.

ABB verifies the function of released products and updates. However system/product owners are ultimately responsible to ensure that any system update (including but not limited to code changes, configuration file changes, third-party software updates or patches, hardware change out, and so on) is compatible with the security measures implemented. The system/product owners must verify that the system and associated products function as expected in the environment they are deployed.

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.

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: October 2015
Document number: 3BSE030340-600 A

Table of Contents

About This User Manual

General 11
User Manual Conventions 11
Feature Pack 11
Warning, Caution, Information, and Tip Icons 12
Terminology 13
Released User Manuals and Release Notes 14

Section 1 - Introduction

Product Overview 15
Product Scope 15
What You Can Do with 800xA for Advant Master 15
Prerequisites and Requirements 16

Section 2 - Configuration

Before You Start 17
Libraries and Object Types 17
Controller Documentation 19
Getting Started 21
Steps for New Systems 21
RTA Board Configuration 22
Setting Up the RTA Board 22
Configuration of Time Synchronization Settings 23
Process Object Lock 29
Process Object Sensitive Hot Keys 31
Predefined Hot Keys 32
The Verb Map Aspect 34
Min and Max Dialog 35

Min and Max Dialog According to Security 35

Section 3 - Engineering

Building the Control Structure 37
Building the Control Structure On-line 39
Building the Control Structure Off-line 43
Building a Control Structure for Test and Demo Purposes 47
Engineering with Control Builder A (CBA) in an 800xA System 49
Maintenance 50
Process Objects in Lost and Found 50
Re-use of Object Names with existing Functionality 50
Reloading Controller Applications 51

Section 4 - Alarm and Event

Defining Event Treatments Using RTA Board Configuration 53
Defining Event Treatment for Process and System Objects 53
Default Alarm and Event List Configurations 53
Defining Event Treatment 55
Defining Event Filter on Node Level 64
Advant Master Alarm Refresh 65
Configuring the Alarm Refresh 66
Controller behavior to consider 68
Alarm Hiding 69
Audit List for Operator Actions 69
External Alarm 70
Events to 3’rd party clients 70

Section 5 - History Logs

Time Tagged Data (TTD) 72
The TTD Source Aspect 74
The Log Template Aspect 74
TTD Node Configuration Aspect 77
Working with TTD logs 83

Direct OPC Logs 93
OPC Direct Logging - 1, 3 and 9 Seconds 94
OPC Direct Logging - Other Storage Intervals 94

Section 6 - Quick List

Quick List Data Source Aspect 95

Section 7 - Data Transfer via OPC DA

OPC Subscriptions 97
OPC Read Operations 98
OPC Write Operations 98
Performance 99

Section 8 - System Administration

System Diagnostics 103
Fault Finding and User Repair 103
System Status 103
System Alarms 120
Backup/Restore Procedures 122
Backup 122
Restore 124
Advanced Connectivity Server Configuration 125
Connect Several MB 300 Networks to an 800xA System 126
Add Multiple Connectivity Servers to the Same MB 300 Network 126
Required Settings to Make an Upload from a Client 129

Section 9 - Advant Master Central Backup

Configuration and Operation 133
Configuring the Advant Master Central Backup 134
Starting a backup manually 135
Configuring a scheduled backup 139
Restoring a backup 140
Maintenance 140

Appendix A - Event Treat Elements

Introduction 143
The Event Treat Database Element 145

Appendix B - MB300 Uploader

Using the From FCB Tab to Upload 153

Appendix C - Event Texts

Analog Input Signal AI 158
Analog Output Signal AO 159
Digital Input Signal DI 160
Digital Output Signal DO 161
Sequence SEQ 161
Process Controller PIDCON 164
Process Controller PIDCONA 166
Manual Station MANSTN 168
Ratio Station RATIOSTN 169
User Defined Controller GENCON 170
Binary Object GENBIN 173
User Defined Object GENUSD 176
Motor Control MOTCON 179
Valve Control VALVECON 182
Group Start GROUP 185
Group Alarm GRPALARM 187
Motor Control MOTCONI 188
Standard Alarm Text from MOTCON PC Element 190
Standard Warning Texts from MCU 192
Standard Alarm Text from MCU 195
Standard Event Text from MCU 198
Event Treatment in Operator Station 199
Engineered Drive, DRICONE 200
Standard Drive, DRICONS 221

Appendix D - Control Aspect

RTA Board Control Aspect....231
RTA Board Control Tab....232
RTA Board - Network and Node Configuration Tab....235
RTA Board Configuration Tab....236
Control Connection Aspect (CCA)....237
CCA Property View Tab....237
CCA Property Info Tab....238
CCA - Additional Info Tab....239
CCA MasterBus 300 Tab....240

Appendix E - Cyclic OPC Properties

OPC Object Type Properties....243

Appendix F - Object Type Names

Appendix G - Hardware Installation

RTA Unit PU410....261
PU410 Specification....261
Hardware Use Cases....270

Appendix H - Special Configuration

Alarm and Event....273
Data Access....276

Revision History

Index

About This User Manual

General

Any security measures described in this User Manual, for example, for user access, password security, network security, firewalls, virus protection, etc., represent possible steps that a user of an 800xA System may want to consider based on a risk assessment for a particular application and installation. This risk assessment, as well as the proper implementation, configuration, installation, operation, administration, and maintenance of all relevant security related equipment, software, and procedures, are the responsibility of the user of the 800xA System.

This manual describes the 800xA for Advant Master product which is used for connecting the workplaces to a MasterBus 300 control network with connected AC 400 Controller Series, including MasterPiece 200/1.

User Manual Conventions

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

Feature Pack

The Feature Pack content (including text, tables, and figures) included in this User Manual is distinguished from the existing content using the following two separators:

Feature Pack Functionality

<Feature Pack Content>

Feature Pack functionality included in an existing table is indicated using a table footnote (*) :

*Feature Pack Functionality

Feature Pack functionality in an existing figure is indicated using callouts.

Unless noted, all other information in this User Manual applies to 800xA Systems with or without a Feature Pack installed.

Warning, Caution, Information, and Tip Icons

This User Manual includes Warning, Caution, and Information 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, fully comply with all Warning and Caution notices.

Terminology

A complete and comprehensive list of terms is included in *System 800xA System Guide Functional Description (3BSE038018*)*. The listing includes terms and definitions that apply to the 800xA System where the usage is different from commonly accepted industry standard definitions and definitions given in standard dictionaries such as Webster's Dictionary of Computer Terms. Terms that uniquely apply to this User Manual are listed in the following table.

Term/Acronym	Description
AC 400	Advant Controller 400 - The ABB family of controllers: AC 410and AC 450.
AMPL	ABB Master Programming Language. The application language for Advant Master Controllers.
MB 300	MasterBus 300 - the control network communication protocol that is used by the AC 400 controllers.
Control Builder A (CBA)	The configuration tool for Advant Master Controller. Control Builder A consists of Application Builder, Bus Configuration Builder, Function Chart Builder and the option On-Line Builder.
Control connection aspect (CCA)	Contains the name, data type, access rights (read/write) and subscription update rate of each attribute and the name of the corresponding OPC item (object of the controller). CCA also contains a user interface to inspect the object type attribute information. It can also be used to subscribe for the current value of each attribute.
Function Chart Builder	Part of the configuration tool Control Builder A.
On-line Builder	The online configuration tool for Advant Master Controllers.

Term/Acronym	Description
RTA board	Real Time Accelerator board The Communication board used for connection to the MasterBus 300 control network, the PU410 RTA unit. The RTA configuration, dialogs, and status indications do not differ or depend on the type of RTA hardware used, once the installation is completed.
TTD	Time Tagged Data - the name of the log functionality in Safeguard and AC 400 Controllers.

Released User Manuals and Release Notes

A complete list of all User Manuals and Release Notes applicable to System 800xA is provided in *System 800xA Released User Manuals and Release Notes (3BUA000263*)*.

System 800xA Released User Manuals and Release Notes (3BUA000263)* is updated each time a document is updated or a new document is released. It is in pdf format and is provided in the following ways:

- Included on the documentation media provided with the system and published to ABB SolutionsBank when released as part of a major or minor release, Service Pack, Feature Pack, or System Revision.
- Published to ABB SolutionsBank when a User Manual or Release Note is updated in between any of the release cycles listed in the first bullet.

A product bulletin is published each time *System 800xA Released User Manuals and Release Notes (3BUA000263*)* is updated and published to ABB SolutionsBank.

Section 1 Introduction

Product Overview

The 800xA for Advant Master is a product for process monitoring and control. It has a generic design and can be used for different process control systems.

This product enables you to integrate an Operator Workplace to a system of AC 400 Series (with Master software) controllers in a MasterBus 300 network.

Product Scope

The 800xA for Advant Master is a product that is integrated in the Operator Workplace. This provides the following features:

- A connection to the control system through a dedicated communication board - the RTA board (Real-Time Accelerator board).
- Tools to build your Operator Workplace Control Structure on-line from the AC 400 controllers or off-line from the AC 400 Engineering tools databases.
- The possibility to customize the event and alarm reporting and presentation with just a few configuration actions.
- To use the AC 400 TTD historical logs as data source to Operator Workplace Historian, but also tools to configure new TTD logs in the controllers.
- Powerful maintenance and supervision of the Advant Master Control system itself using the System Status, System Alarms and RTA Board Control functions.

What You Can Do with 800xA for Advant Master

This list is an example of the configuration that you can do with 800xA for Advant Master.

- Configure the connection of the workplace to the MasterBus 300 control network.
- Build new graphic displays.
- Setup the workplace for handling the process data that is defined in the database of the AC 400 controllers. You do not need to define the database again, it is read from the controllers over the network.
- Setup the event handling for the process events that are reported from the AC 400 controllers.
- Define and activate the logging of process data, including presentation of the stored data logs.
- Collect and present System status for the entire control system, including the AC 400 controllers and the MB 300 control network.

Prerequisites and Requirements

The general hardware and software requirements for the Operator Workplace product are described in the *System 800xA Installation and Upgrade Getting Started (2PAA111708*).*

Section 2 Configuration

Before You Start

Ensure to have access to one of the following:

- The Connectivity Server is connected to the AC 400 Series controllers (the MasterBus 300 control network).
- The database instances from the Function Chart Builder (AC 400 Series Controller Engineering tools).

For more information, refer to *System 800xA, Operations, Operator Workplace Configuration (3BSE030322*).*

The user should login as an Application Engineer or System Engineer to be able to perform the configuration work described in this section.

The Application Engineer or System Engineer should belong to the MS Windows “Power Users” group or the “Administrators” group to perform the configuration.

Libraries and Object Types

800xA for Advant Master defines a set of object types. An object type is a predefined aspect object defined in the **Object Type Structure**.

Object types in 800xA for Advant Master (for example, AC 450 Controller, MB300 AI, MB300 PID Controller, and MB300 Valve) represent the data base element types in the AC 400 series controller.

During upload from controllers, the **Uploader** aspect creates instances of the controller objects with the respective object types in **Control Structure**.

The Advant Master object types are included in libraries in the 800xA system version 5.1 and later versions. The Base Library *AdvantMasterObjectTypes* defines all Advant Master object types including the graphics independent aspects. The Extension Library *AdvantMasterObjectTypesPG2Ext* defines all process graphics aspects.

Figure 1. 800xA for Advant Master Libraries in the Library Structure

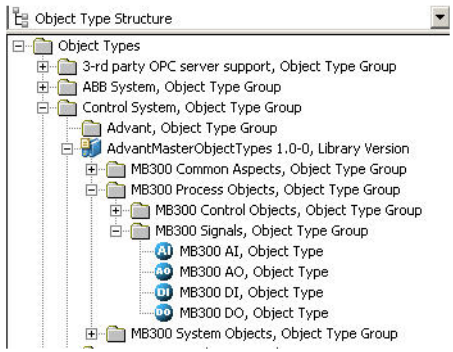


Figure 2. Base Library AdvantMasterObjectTypes in the Object Type Structure

A library is in one of the following states, *Open*, *Closed*, or *Released*.

After installation of the product, libraries are in the *Closed* state and are protected from unintentional modifications. Custom modifications are possible if the library is forced to the *Open* state. Observe that the modifications may be lost in future system upgrades/updates.

For more information on library handling and object types, refer to *System 800xA, Configuration (3BDS011222*)*.

Controller Documentation

The AC 400 Series controller documentation covering all manuals needed for configuration, operation or maintenance of AC 450 or AC 410 controllers are included in 800xA for Advant Master. The manuals are available in the following ways:

- Via object type specific bookmarks available as aspects in the Control Structure on the uploaded Advant Master objects. The manuals can also be opened via the context menu on Advant Master objects in the System Status viewer. See Figure 3 below.

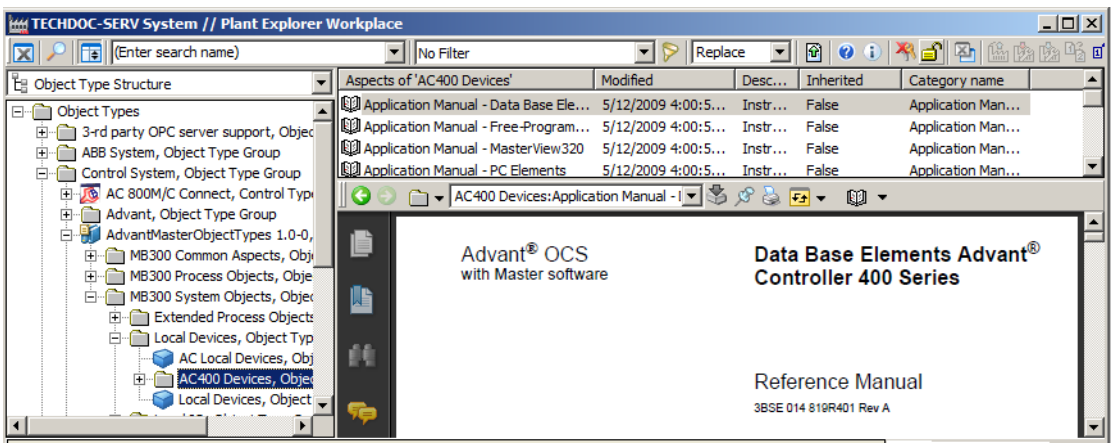


Figure 3. Documentation

- The complete collection of AC 400 Series manuals can be found on the AC 400 Documentation object in the Object type structure. The manuals that do not have a related object type in 800xA for Advant Master can only be opened via the AC 400 Documentation object. See Figure 4 below.

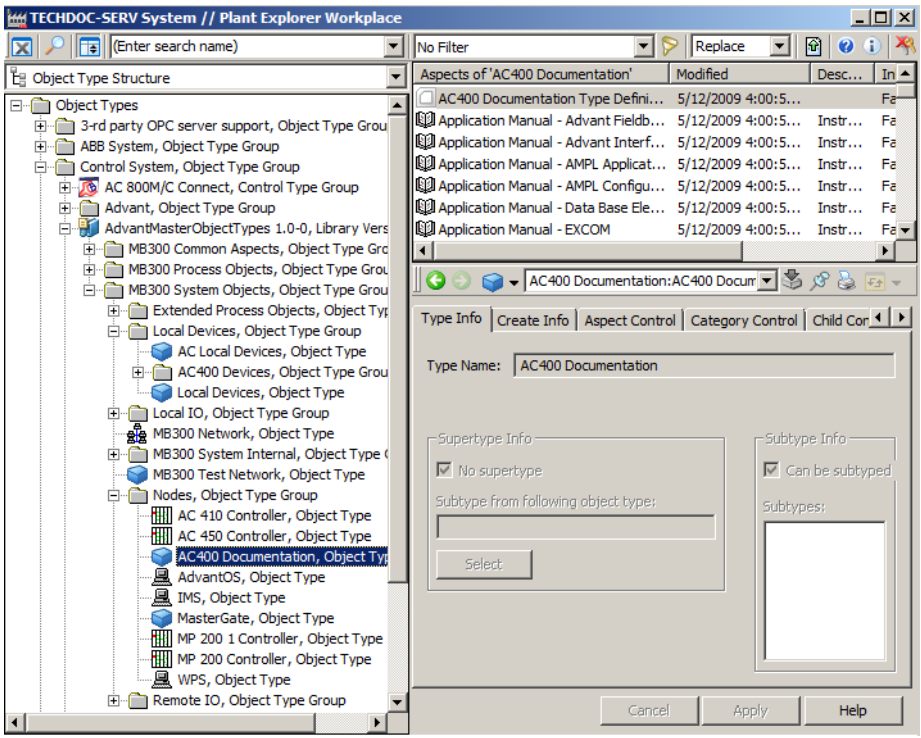


Figure 4. AC 400 Documentation Object

Getting Started

Below are some different steps to take into consideration when configuring an 800xA for Advant Master system.

Steps for New Systems

1. Build the Controller Application, if required, in the Engineering station.
   a. Create and configure Process and System objects
   b. Define Log groups (TTD)
   c. Write your AMPL applications
2. Setup the parameters for the RTA Board (network and node address, time synchronization, character conversion).
3. Download the applications to the controllers, using the Online Builder.
4. Build the Control Structure (Create controller objects and retrieve the information about the Process and system objects from the respective controller).
5. Modify (if required) the Alarm and Event default setting.
6. Configure the TTD logs in the controller and the corresponding log hierarchies above them.

The Engineering of the controllers and the various steps to perform is not described in this book. Please read the corresponding User´s Guides.

RTA Board Configuration

Setting Up the RTA Board

The configuration settings for the RTA Board include:

- Setting Network and Node number on the MasterBus 300 Control Network
- Setting PU410 redundant cable supervision
- Setting Time synchronization for RTA / PC clock synchronization
- Character Conversion

The network and node numbers must be setup before the 800xA system can communicate with the control network.

The RTA Board configuration view (Network and Node configuration) will also come up during the configuration of the system. The view is included when you run the Configuration wizard.

Setting Network and Node Number

Use Configuration Wizard in order to setup network and node number. For more information on adding the RTA Board, which also creates the network object, refer to *System 800xA, Post Installation (2PAA111693*).*

For more information on adding multiple Connectivity Servers to the same MB 300 network, refer to Add Multiple Connectivity Servers to the Same MB 300 Network on page 126.

Setting PU410 Redundant Cable Supervision

If the RTA unit PU410 is connected to Connectivity Server with redundant cables, it is recommended to select the **PU410 redundant cable supervision** checkbox in the Network/Node Configuration tab. Status changes are then reported by the MasterBus 300 RTA Management service provider object in Service Structure. For more information, refer to System Status - Service Structure on page 104.

Selecting RTA Unit Character Conversion

Select a character conversion in the RTA Board Configuration tab that matches the controller application. Possible values are English, German and Swedish with English as default.

Configuration of Time Synchronization Settings

This section describes two possible configuration settings for time synchronization with MB 300 network.

- **AC 400 Series controller as Clock Master**, in a configuration with MB 300.
- **800xA for Advant Master Connectivity Server as Clock Master**, in a configuration with AC 400 Series controllers on MB 300 and 800xA for AC 800M controllers on a TCP/IP network.

A thorough description of synchronization strategies and overall configuration is provided in *System 800xA, Automation System Network, Design and Configuration (3BSE034463*).*

AC 400 Series Controller as Clock Master

When you have MB 300 as the only control network in your configuration, it is recommended to use an AC 400 Series controller as clock master. The described configuration will receive time synchronization from the MB 300 network and distribute it to other Operator Workplace nodes (e.g Aspect Servers, Clients) in the system. Manual time settings (e.g from client nodes) or local time shifts in the system are propagated to the RTA Board and the MB 300 network.

The following settings are used for this configuration:

- Verify that the Clock Master setting in the Configuration Wizard is correct. The **800xA as Clock Master (REVERSE_SYNCH_MODE)** check box shall be unchecked.

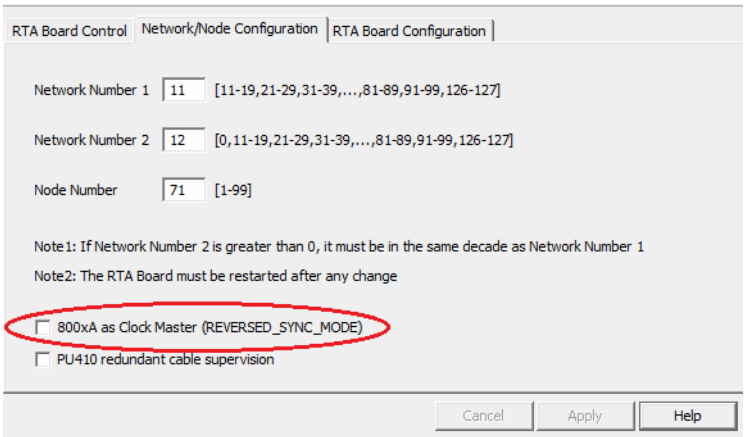


Figure 5. AC 400 as Clock Master

- **RTA Board Settings**

```
CLK_MAST = 0
LOC_TIME = 2
CLK_SEND = 1
```

No changes of the default configuration values are required.

- **Time Service Master Settings**

To distribute the time received from the MB 300 network to other Operator Workplace nodes (e.g Aspect Servers, Clients) in the system, you have to configure the time service in the dedicated Connectivity Server as time service master. This is done in the following way.

1. Go to the **Service Structure** and select **Time Service**.
2. Select **Basic**, **Service Group**.

3. Select the **Service Group Definition** aspect.

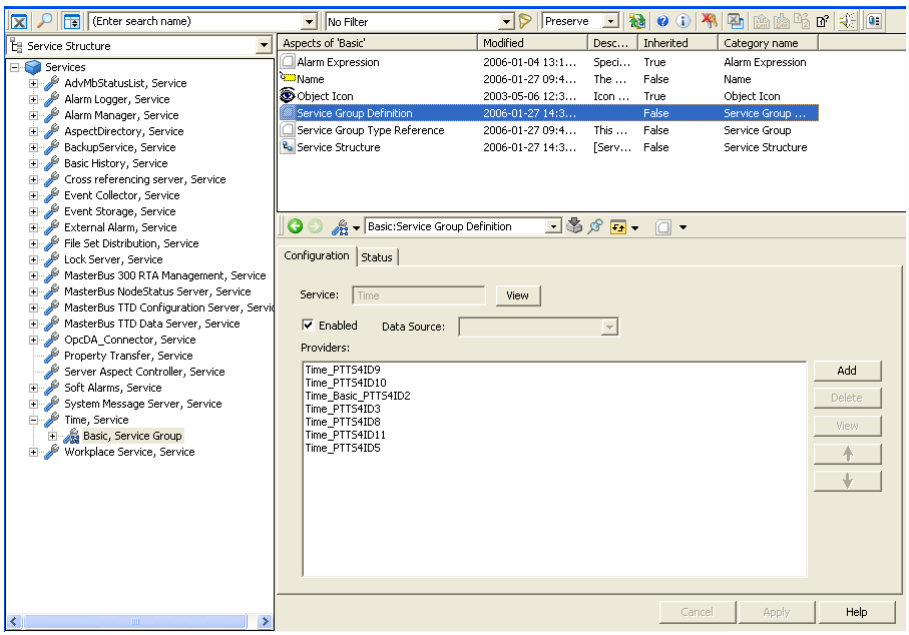


Figure 6. Service Structure

4. Select the **Configuration** tab in the aspect window.
5. Place the Provider, selected to be master, first in the list. The providers below will automatically be configured as standby.
6. Place the redundant provider second in the list.

All Time Synchronization services should be enabled.

800xA for Advant Master Connectivity Server as Clock Master

This description is valid for a system configuration with AC 400 Series controllers on MB 300 and 800xA for AC 800M controllers on a TCP/IP network. The described configuration will synchronize all nodes on the MB 300 network from the RTA Board in the configured CS.

1. Expand the Control Structure in the Plant Explorer, and select the RTA Board object.
2. Select the RTA Board Control aspect in the aspect list.
3. Select the **Network/Node Configuration** tab and change the synchronization direction between the Connectivity Server and the RTA Board. Check the **800xA as Clock Master (REVERSED_SYNC_MODE)** check box, **Apply**.

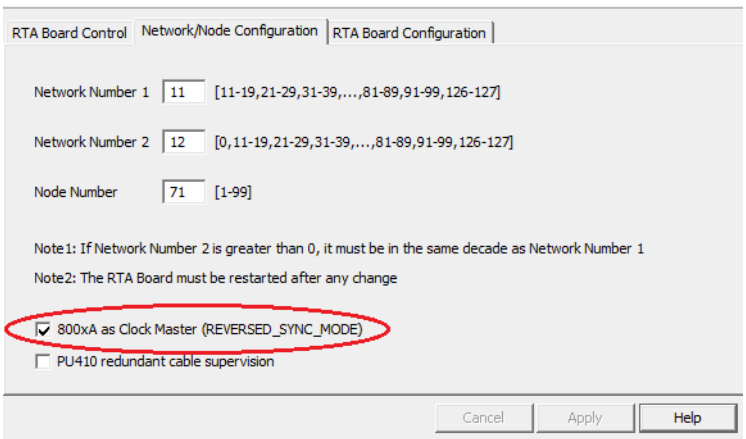


Figure 7. 800xA as Clock Master Check Box

4. Select the **RTA Board Configuration** tab and click the **RTA Board Config** button.

   In the RTA Board Configuration window, enter **mdb CLOCK_SYNCH.**

   Change the “CLK_MAST” parameter to “1” and the ‘’LOC_TIME’’ to ‘’0’’ as described in window, see Figure 9.

This is done locally in the Connectivity Servers with a RTA Board.

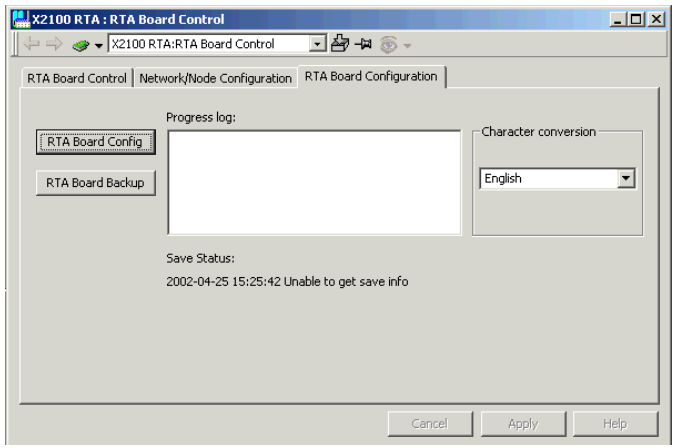


Figure 8. RTA Board Control Aspect

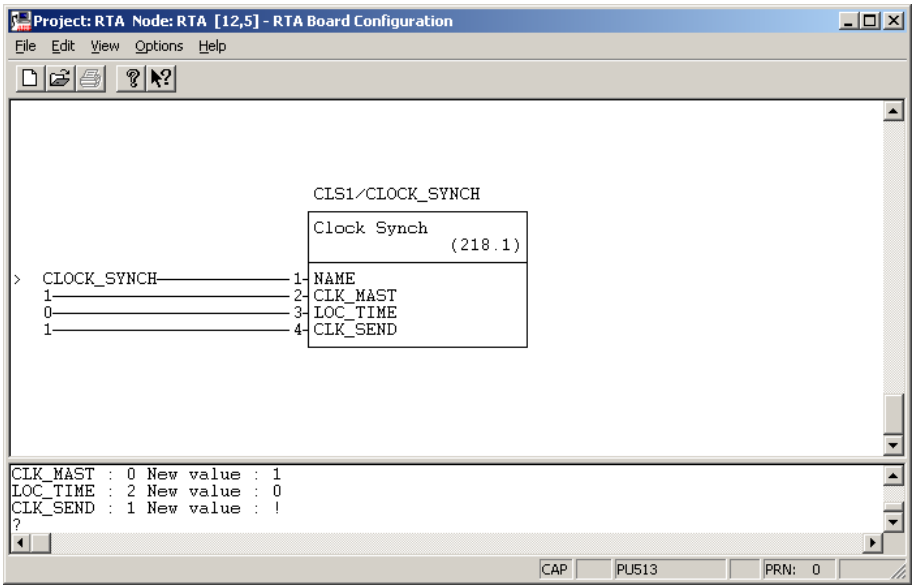


Figure 9. CLOCK_SYNCH DB Element

5. Close the window.
6. Start **RTA Board Backup**.
7. You have to **Stop** and then **Start** the RTA Board, to activate the changes. This is done from the **RTA Board Control** tab in the RTA Board Control aspect.

No Synchronization

If you for some reason (mixed systems) do not want any synchronization between the RTA Board (MB 300 network) and the Connectivity Server, you can set the parameter SYNC_INTERVAL to a specific value in the registry (**FFFFFFFF**).

It is always a risk to change values in the Windows registry edit. Be sure you have a backup for safety reasons, so the computer can be restored if something goes wrong.

1. Open **Run** and enter `regedit`. The registry is opened.
2. Open the KEY:
   HKEY_LOCAL_MACHINE\SOFTWARE\ABB\AFW\SystemModules\
   AfwTimeServerAdaptor\1.0-0\Private
3. Change the parameter SYNC_INTERVAL from A to FFFFFFFF.
4. Restart the time server.

Process Object Lock

800xA for Advant Master uses the manual lock (no configuration required) or the autolock (configuration via Graphics Profile Values aspect). This lock mechanism uses the lock flag (select) in the Advant Master controller.

It is possible to lock a process object via the lock button in the Faceplate to exclude other operators from operation on the specific object. The lock is released when the Faceplate is closed, the lock button is clicked or after a specific time-out period of inactivity. It is however not required to lock an object before starting to operate it.

The autolock function can be configured so that the object is automatically locked at Faceplate call-up. This function can be enabled for a specific user via the User Profile aspect Graphics Profile Values in the User Structure (see Figure 10).
See *System 800xA, Operations, Operator Workplace Configuration (3BSE030322*)* for details on how to configure User Profile aspects.

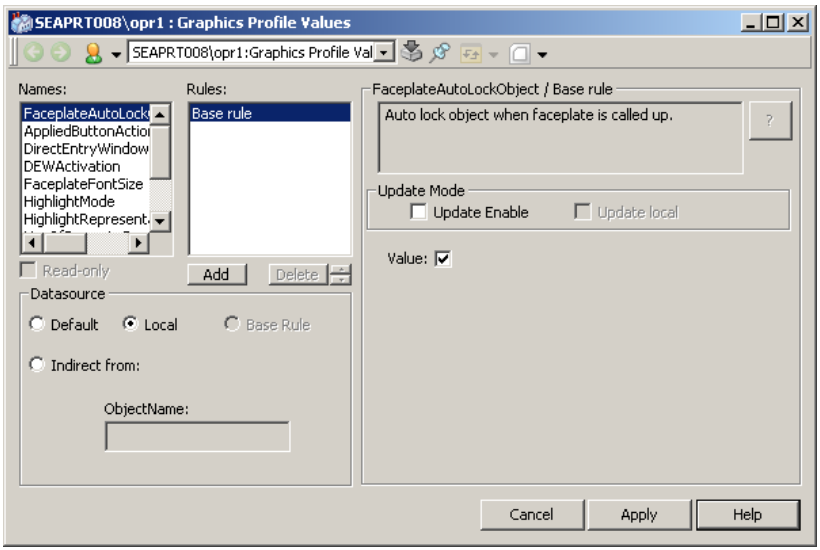


Figure 10. Autolock Enabled in an Graphics Profile Values Aspect

800xA for Advant Master does not support the 800xA Base System Lock Server function and its extended functions, such as **Break Lock**.

It is thus not supported to uncheck the checkbox "Do not use standard lock server for process object locking" in Special Configuration tab of OPC DA Service Group Definition.

In the controller, a process object may remain locked up to 5 minutes in the following scenarios:

1) The power of the connectivity server system is turned off.

2) The connectivity server is disconnected from the MB300 network.

The object can be operated after the timeout but the lock (SELECTED) flag in the controller will not be reset to indicate this in the Faceplate.

This problem does not occur if the following versions of Advant Controller 400 series are used:

1) Advant Controller 410 *1.5/8 (or later).

2) Advant Controller 450 *2.3/8 (or later).

The faceplates that are open will display a yellow colored *locked-by-other* symbol. It is not possible to lock the object by clicking the Lock button.

To resolve a *locked-by-other* situation in a faceplate, execute the following steps:

1) Wait 5 minutes after the lock is released for the process object.

2) If autolock is configured, close and reopen the required faceplates.

If manual lock is used, ignore the yellow indication and perform any operation that is safe for the process (for example, block and directly deblock the alarm).

Process Object Sensitive Hot Keys

The Hot Keys Aspect is located in the workplace structure.

You use the Hot Keys Aspect System to configure Hot Keys. It enables configuration of key combinations that perform specific functions, such as opening an aspect view or activating a verb. The hot key operation is done on highlighted objects.

To secure that all hotkeys can be used, refer to the procedure described in *Hot Keys* section in *System 800xA, Operations, Operator Workplace Configuration (3BSE030322*).*

For more information about highlighted objects, see *System 800xA, Operation (3BSE036904*).*

Predefined Hot Keys

The following set of Hot Keys are predefined in 800xA for Advant Master:

Table 1. Predefined Hot Keys 1

Object Type	Acknowledge CTRL+ SHIFT+ Q	On/Start/ Open/True CTRL+ SHIFT+ 1	Off/Stop Close/False CTRL+ SHIFT+ 0	Man CTRL+ SHIFT+ Y	Auto CTRL+ SHIFT+ U
AI	X				
AO	X			X	X
DI	X	X	X		
DO	X	X	X	X	X
PIDCONA	X			X	X
PIDCON	X			X	X
MANSTN	X			X	
RATIONSTN	X			X	X
MOTCON	X	X	X	X	X
VALVECON	X	X	X	X	X
GROUP	X	X	X	X	X
SEQUENCE	X	X	X	X	X
GENBIN	X	X	X	X	X
GENCON	X			X	X
GENUSD	X			X	X
DATB		X	X		
TEXT_DATA	X	X	X	X	X
DRICONE	X	X	X	X	X

Table 1. Predefined Hot Keys 1 (Continued)

Object Type	Acknowledge CTRL+ SHIFT+ Q	On/Start/ Open/True CTRL+ SHIFT+ 1	Off/Stop Close/False CTRL+ SHIFT+ 0	Man CTRL+ SHIFT+ Y	Auto CTRL+ SHIFT+ U
DRICONS	X	X	X	X	X
MOTCON_I	X	X	X	X	X

Table 2. Predefined Hot Keys 2

Object Type	E1 CTRL+ SHIFT+ O	E2 CTRL+ SHIFT+ P	Small Increase CTRL+ SHIFT+ K	Small Decrease CTRL+ SHIFT+ H	Large Increase CTRL+ SHIFT+ L	Large Decrease CTRL+ SHIFT+ J
AI						
AO			X	X		
DI						
DO						
PIDCONA	X	X	X	X	X	X
PIDCON	X	X	X	X	X	X
MANSTN	X		X	X	X	X
RATIONSTN	X		X	X	X	X
MOTCON						
VALVECON						
GROUP						
SEQUENCE						

Table 2. Predefined Hot Keys 2 (Continued)

Object Type	E1 CTRL+ SHIFT+ O	E2 CTRL+ SHIFT+ P	Small Increase CTRL+ SHIFT+ K	Small Decrease CTRL+ SHIFT+ H	Large Increase CTRL+ SHIFT+ L	Large Decrease CTRL+ SHIFT+ J
GENBIN						
GENCON	X	X	X	X	X	X
GENUSD			X	X	X	X
DATB						
TEXT_DATA						
DRICONE						
DRICONS						
MOTCON_I						

The Verb Map Aspect

The Verb Map aspect extends the support provided by the Hot Keys aspect. It provides a cross connection, which makes it easier to apply the Hot Key support for highlighted and selected objects. It makes it also easier for different connected OCS systems to apply the same hot keys for their objects.

Objects with the Verb Map aspect can be part of a hot key scheme using the selected and highlighted concepts.

For more information about the Verb Map Aspect see *System 800xA, Operations, Operator Workplace Configuration (3BSE030322*).*

Min and Max Dialog

Min and Max Dialog According to Security

In order to reflect min and max dialog of the AS500OS all writable properties in the control connection for each of the Advant Master objects types have been defined to require either tune or operate permission. This together with the corresponding system security definition will allow the operators to operate the min dialog but not the max dialog properties.

The introduction of min/max dialog does not hide any information from the user; instead it will give the system configurator the means to restrict the operations available for a single user or a group of users.

When the user does not possess the permissions required for an operation the buttons in the faceplate will be dimmed. Number fields will not be open for input etc.

The most common is that an operator is permitted to operate, e.g. enter values. If you have more permissions (as an Application Engineer) you may be allowed to tune, e.g. set alarm limits or blocking parameters.

For more information about Security settings, refer to *System 800xA, Administration and Security (3BSE037410*).*

Section 3 Engineering

This section contains information about how to build your Control Structure (on-line, off-line and for test purposes) and how to include Control Builder A in an 800xA System.

Building the Control Structure

The Control Structure is based on the controllers in your Advant Master System. Preferably it should be built on-line, with connection to the controllers, see Building the Control Structure On-line on page 39.

The Control Structure can also be built using the output from the Controller Engineering tool FCB, see Building the Control Structure On-line on page 39. This has some advantages for a new site:

- You can create the Control Structure step by step as the Controller Engineering proceeds and thereby extend your Operator Workplace applications stepwise.
- You can work off-line from the controllers (without connection to the control network), also in a site geographically separated from where the Controller Engineering is made.

The RTA Board object should be available before starting to build the Control Structure.

The MB300 Uploader aspect is used both when building the Control Structure on-line and off-line. The MB 300 Uploader aspect is placed in the Control Structure on network and node level. See Figure 11.

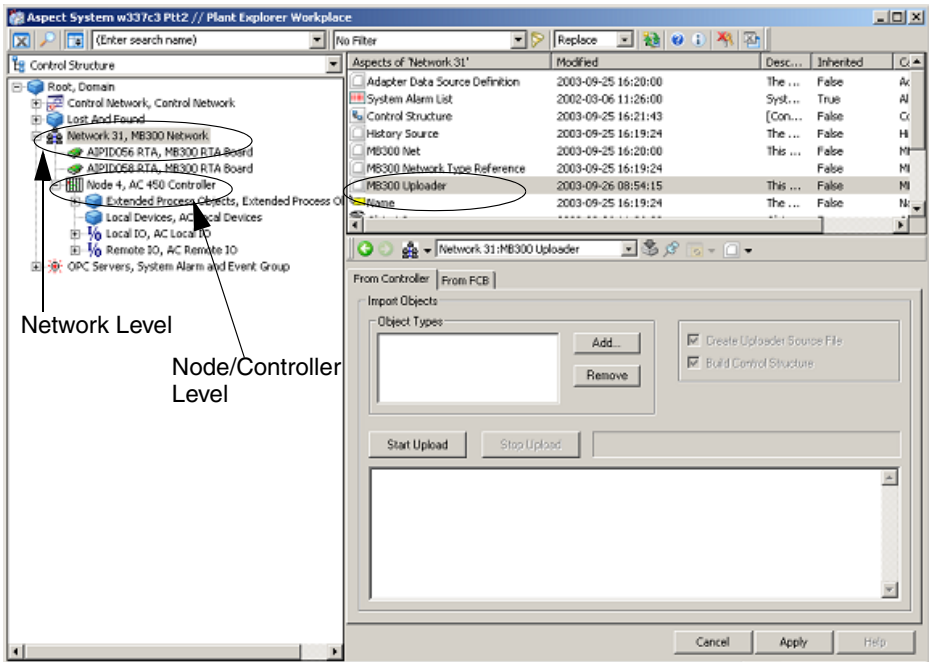


Figure 11. Upload - Control Structure

If you want to make an upload from an 800xA client see Required Settings to Make an Upload from a Client on page 129.

Building the Control Structure On-line

Add each AC 400 Series or MP200/1 Controller to the structure using the correct node object type as template and enter the correct network and node number.

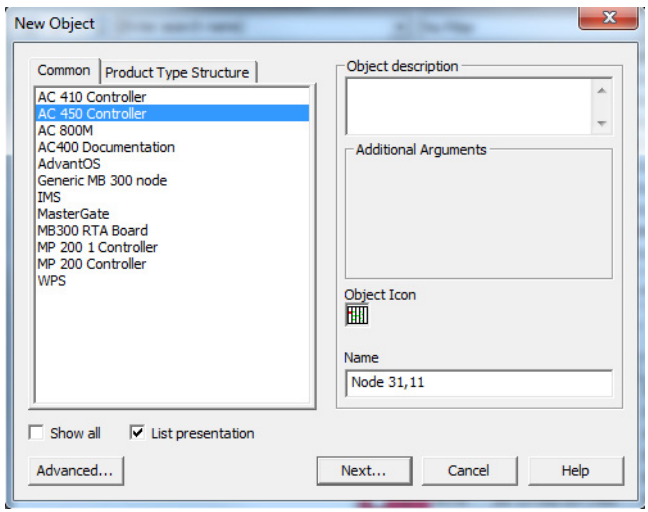


Figure 12. Creating a Controller Node Object

It is also possible to add other node types such as AC 800M and Generic MB 300 node. These node types are not genuine Advant Master Controllers, and have no support for Upload and system status. Add these node types to Control Structure to make them visible in the Advant Master Connectivity Server Communication Display (Figure 60) as nodes on MasterBus 300.

The From Controller tab in the MB300 Uploader aspect is used to upload from one controller or from all controllers on a MB300 network depending on the location of the MB300 Uploader aspect.

Before starting the upload procedure it is important to select a character conversion in the RTA Board Control aspect that matches the controller application to prevent incorrect naming. Possible values are English, German and Swedish with English as default.

If character conversion is changed after the upload and a new upload is done, object identities might change. For example, links from Trends and process graphics can be lost as well as event treatment element settings. See *System 800xA, NLS Localization (2PAA101940*)* for more information.

Perform Upload in the MB300 Uploader aspect on a specific controller node to upload the whole structure of I/O boards and Process and System objects from one controller. If Upload instead is performed in the MB300 Uploader aspect on network level (see Figure 11), all nodes below are uploaded, one node at a time.

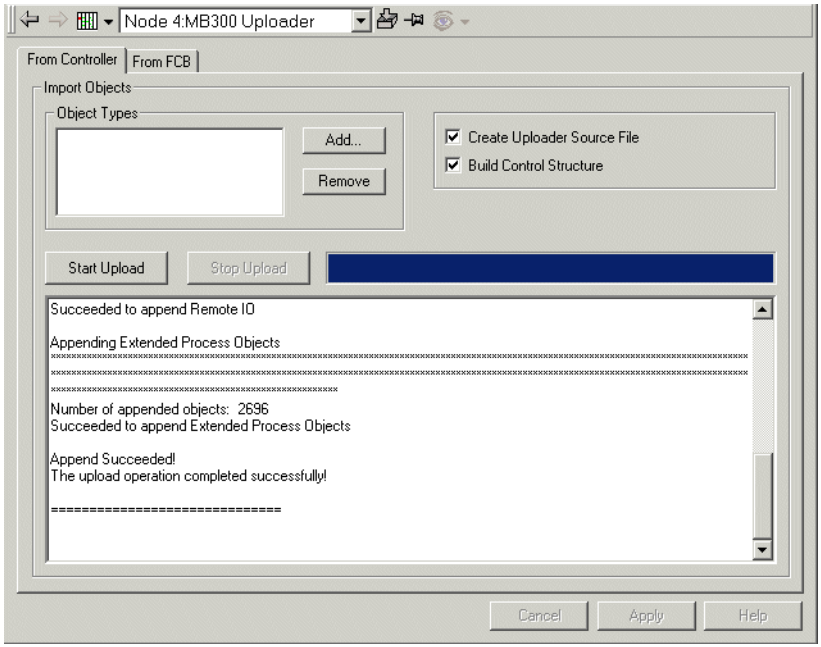


Figure 13. MB 300 Uploader, From Controller

Adding or Removing Object Types to Upload

Click **Add** or **Remove** to modify the list with object types that you want to include in the Upload. If you leave the list blank, all object types are included (this is the default).

The following object types are valid for Upload:

- All object types located in MB 300 Process Objects group.
- Bus and Station object types under Advant Fieldbus 100, Lonbus and Profibus.
- S400 MasterFieldbus object type.

Create Uploader from Source File and Build Control Structure

These check boxes should always be marked. They are only used for fault tracing and simulation, and should not be modified. They should only be modified on direct request from the supplier.

Starting the Upload

Click on **Start Upload** when you are ready for the upload. The window in the lower part of the aspect view presents messages describing the progress, see Figure 13.

Stop the Upload

Click on **Stop Upload** if you want to interrupt the upload.

Uploading may take time depending on application and controller type.

If you have two parallel connectivity servers, the upload is performed through the primary connectivity server, irrespective of which connectivity server or client you are making the upload from. However, if the primary connectivity server is down, the upload is performed through the secondary connectivity server.

The primary connectivity server is the first provider in the OpcDA service group.

Building the Control Structure Off-line

Building the Control Structure Off-line involves the following additional work compared to the on-line procedure:

- Use the FCB MB 300 Uploader option on the Engineering PC. Create source files for each controller and copy them to the 800xA for Advant Master Server.
- Run the MB 300 Uploader aspect for each controller using the source files as input. You can populate one controller at a time or one or a couple of object types at a time.
- When the controller is available on the network, you can retrieve some of the addresses, that are only available in the controller, to the Control Structure. Run the MB 300 Uploader directly towards the controller. This is required when the controller is ready to go on-line.

The following step by step tutorial describes how to create the complete Control Structure for the AC 450 Controller (node address 31,18) starting with an empty Control Structure. We assume that you have already installed the MB 300 Uploader option on the PC where you have the engineering software running.

Refer to the Function Chart Builder User's Guide for a detailed description on how to build the controller objects.

1. Build the Controller objects with the Function Chart Builder engineering tool.
2. Use FCB MB 300 Uploader option.

   Select **ABB Start Menu> ABB Control Builder A> Utilities > MB 300 Upload >CBA Uploader FCB** on the Engineering PC.

   For information on accessing the ABB Start Menu, refer to *System 800xA, Tools (2PAA101888*)*.

The following dialog box appears:

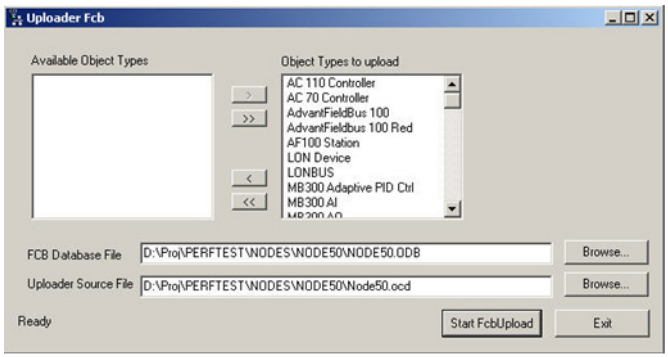


Figure 14. FCB Upload

3. Browse to the FCB database file in the project structure.
4. Enter the name of the file that should be created (the Uploader Source File). For example. set the name to *Node50.ocd*.
5. Ensure that only the object types that you want to upload are shown in the **Object Types to upload** list. You move object types between the lists by selecting them and using the arrow buttons.
6. Start the upload process by clicking the **Start FcbUpload** button.
7. Make the source file you just created accessible from the Operator Workplace server. There are several ways to make the file accessible.
   - Copy the file to a shared disk that is available from the server.
   - Copy the file to a removable media.

Login to the 800xA server machine as a System Engineer and perform the following:

1. Select the Control Structure in Plant Explorer.
2. Open the context menu on Network object and select **New Object**.

3. Select the **AC 450 Controller** Object Type, enter the Node name, and click **Create**.

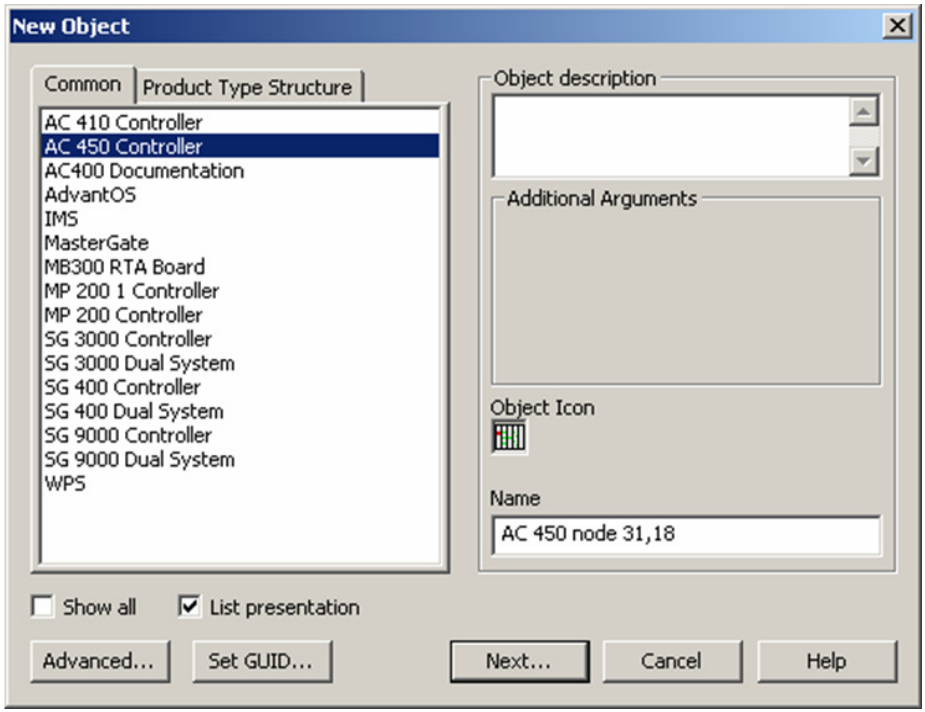


Figure 15. Selecting the Object Type AC 450

4. From the Controller 31, 18 object, select the MB 300 Uploader aspect, and select the **From FCB** tab. See Appendix Appendix B, MB300 Uploader for details and limitations regarding the FCB Upload.
5. Clear the **Create Uploader Source File** check box. This allows the user to edit the **Uploader Source File** text field.
6. Enter the name of the transported source file, for example, *Node50.ocd*.
7. Select **Start Upload** and then click **Apply** to save the changes. Refer to Figure 16.

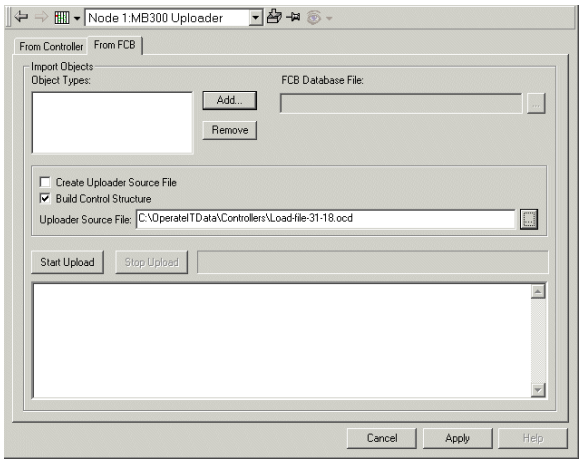


Figure 16. Creating the Control Structure for Node 31, 18

This step can be split up into several parts, as it is possible to work with one object type at a time. If you know that you have only added or modified objects of type AI in the engineering database, then other object types do not have to be uploaded.

8. Now that the Uploader step is done, the Control Structure is updated with the objects included in the FCB source file. You can start building applications based on these objects. For more information, refer to *800xA, Operations, Operator Workplace Configuration (3BSE030322*)*.

9. Before you can go on-line with your system, you need to make an additional upload session, directly toward the controllers. The reason is that some of the physical addresses in the controllers are not available in the FCB files.

   However, you have saved considerable time by using the time-lag between Controller Engineering and the time the controllers are installed and up and running.

10. Select the **From Controller** tab and click **Start Upload**. This will upload all object data from the controller.

Building a Control Structure for Test and Demo Purposes

It may sometimes be an advantage to be able to test or demonstrate your applications in a simulated Control Structure.

Note that the graphic presentation of object data, in the shape of faceplates or display elements, does not work directly with simulated values from the Property Signal Generator. You must define supervision and presentation properties to input “simulated live data” into the presentation of faceplates or display elements.

You can build a Control Structure with simulated values through the following steps:

1. Select the Control Structure in Plant Explorer.
2. Build a structure that includes the process and system objects that you want to simulate.
3. Add the Property Signal Generator aspect to each object (process and system objects) that you want simulated data from.

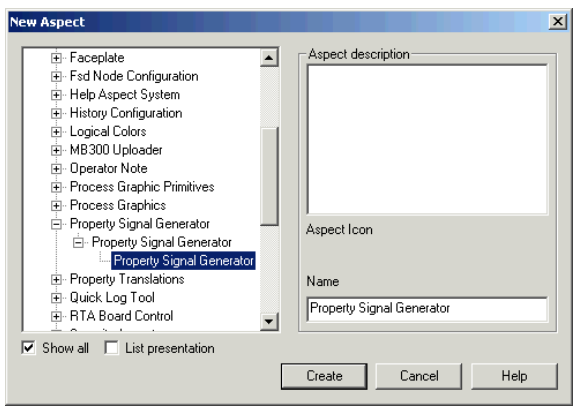


Figure 17. Add Aspect Property Signal Generator

4. Set up the configuration of the Signal Generator aspect to generate data for the specific attributes. This set up includes specifying variable data style - Sin wave, Ramp, Random, Static Value etc. You specify the set up for each object attribute.

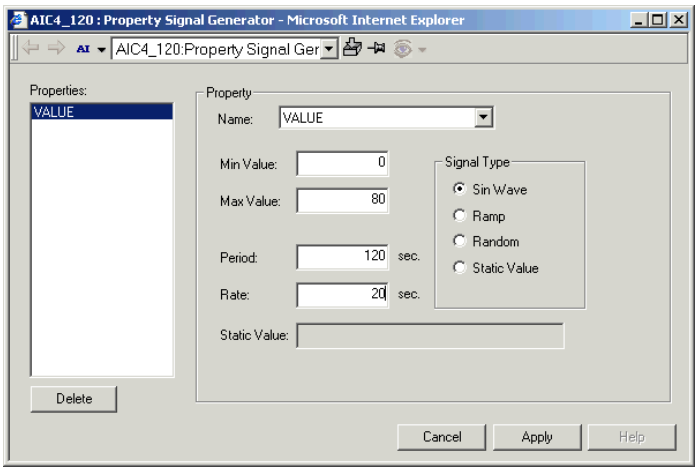


Figure 18. Setup for Property Signal Generator

5. Build your applications to work with this Control Structure with simulated attributes.
6. Test your applications (demo applications, test applications).
7. Save the controller node objects in the Control Structure for future use by exporting each node separately, one by one.

   Start the Import/Export tool through **ABB Start Menu > ABB Industrial IT 800xA > System > Import Export**. Use drag and drop when exporting the objects from the Plant Explorer to the Import/Export tool.

   For information on accessing the ABB Start Menu, refer to *System 800xA, Tools (2PAA101888*).*

Engineering with Control Builder A (CBA) in an 800xA System

Control Builder A, including Application Builder (APB), Function Chart Builder (FCB) and On-line Builder (ONB), is used for engineering of Advant Controller 400 Series.

For information on installation of CBA, refer to *Control Builder A Version 1.4 Installation (3BSE066174*)*.

ONB uses an Engineering Board (PU410 unit) to communicate with AC 400 controllers. The RTA board (PU410 unit) in an 800xA for Advant Master Connectivity Server can act as an Engineering Board. In this case, a server installation of ONB shall be made in one or more Connectivity Servers and a client installation of ONB shall be made in one or more 800xA clients.

For information on client and server installations of ONB, refer to *Control Builder A Version 1.4 Installation (3BSE066174*)*. Following are important hints on client and server installations of ONB:

- The following question will appear at the beginning of ONB installations:

  Will this computer be an 800xA for Advant Master Connectivity Server?

  Answer **Yes** when installing ONB in a Connectivity Server, otherwise **No**.
- Select **No Engineering Board** in client installations.
- Follow the instructions in Server and Client Configuration section carefully.
- In clients, select **Use Engineering Board of the following servers** and add server(s) as described in section Client configuration - Engineering Board Setup.
- In servers, click on **Configure Windows Firewall** in Autorun display of the CBA media.

Maintenance

Process Objects in Lost and Found

When an 800xA system is connected to a running MB 300 network for the first time, it directly starts to receive new alarm and events from controller objects. Since the MB 300 Uploader has not been accomplished in the 800xA Control Structure, these objects are created in the Lost and Found object. Once created, alarms and events will continue to be associated with these Lost and Found object, even after upload is accomplished. The temporary objects are identified with the following name syntax:

<Object Name>(<Network decade>,<Node>)

After the Upload is completed, it is recommended to delete these objects under Lost and Found. Subsequent alarms will then be associated with the correct object in Control Structure.

Re-use of Object Names with existing Functionality

It is recommended to avoid re-using object names. Re-used object names can cause the MB 300 Uploader to be unable to resolve uploaded targets. If the user is not able to avoid re-using object names, for example, if an input channel is broken and needs to be replaced with another channel, follow the below procedure. It will prevent the uploader function from deleting the existing object, which results in broken references.

1. Ensure that no alarms exist from the object. Block alarms through the faceplate and acknowledge any unacknowledged alarms.
2. Change the physical connection from the faulty input channel to an unused channel.
3. From On-Line Builder, open the DB element for the faulty channel, and rename it to a name indicating a broken channel.
4. Open the DB element for the new channel and give it the same name as the broken channel had originally. (Other terminals may also be updated.)
5. Find the process object in the Control Structure. It is placed at the old location under the old channel. Drag and drop the process object to the new location

under the new channel, where an object may already exist as shown in the figure below.

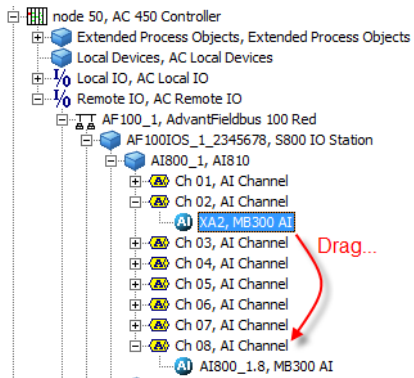


Figure 19. Process Object

6. Perform Upload.
7. Unblock alarms on the object again.
8. Graphic displays, object displays and faceplates holding data from the moved process object should be closed and reopened to present actual data. Restart the History logs.

Reloading Controller Applications

After larger controller engineering restructuring, such as AMPL Source Code transfer, some DB elements may be reallocated to another database index (logical record number). Performing this reallocation while System 800xA is up and running may lead to an unexpected system behavior, such as data is not presented or wrong data is presented in graphics and trend displays after the controller is back in operation.

If reallocation of DB elements is suspected, the following must be considered to avoid a faulty system behavior:

- Close and reopen Alarm lists, Graphic displays, and Faceplates presenting data from the controller .

- Restart History logs or restart the complete History Server.
- Restart 3rd party OPC clients.
- Restart Property Transfer service.

Section 4 Alarm and Event

To obtain a proper Alarm and Event functionality, configuration of following levels has to be done:

- The Database Element in the controller representing the process object. See Defining Event Treatment for Process and System Objects on page 53.
- Event Treatment Database element in RTA Board. See Defining Event Treatments Using RTA Board Configuration on page 53.
- Alarm and Event List configurations in the Library Structure. See Default Alarm and Event List Configurations on page 53.

Defining Event Treatments Using RTA Board Configuration

You define the Event treatment by configuring the 'Event' data base elements on the RTA Board. See Appendix A, Event Treat Elements to get a detailed description of the data base elements that you can configure. Appendix C, Event Texts contains a detailed description how to setup the event text elements.

When you have setup the event treatment, you shall perform a RTA Board Backup to save your work. Look in RTA Board Configuration Tab on page 236 for guidance how to save the RTA database.

Defining Event Treatment for Process and System Objects

Default Alarm and Event List Configurations

The Alarm and Event list used by MB 300 Object Types are configured to use the Common Alarm & Event List Configurations located in the Library Structure.

They are predefined to meet your needs.

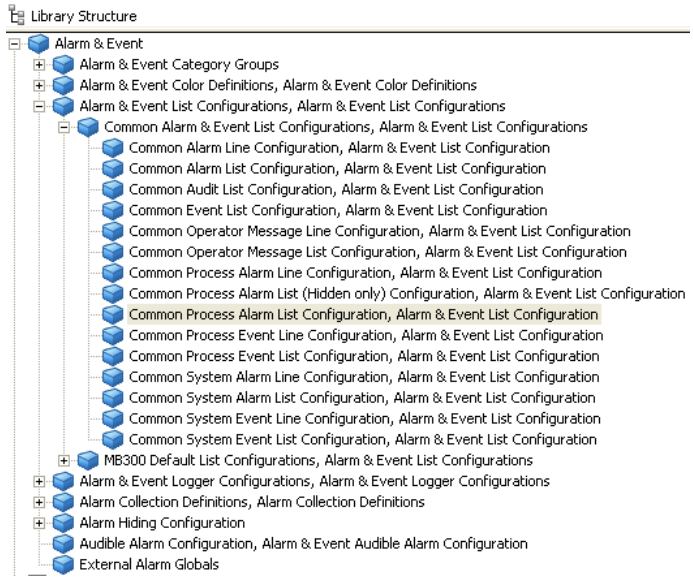


Figure 20. Common Alarm and Event List Configurations in the Library Structure

How to change Alarm and Event List Configurations are described in *System 800xA Operations Operator Workplace Configuration (3BSE030322*)* and *System 800xA Configuration (3BDS011222*)*.

800xA for Advant Master defines the following alarm categories:

- Process - Alarm and Events from process objects in the controller.
- System Alarm - System Alarms from the controller.
- PcEvent - Events from the PC Element EVENT.

Operator Messages are reported via the category ‘’Operation’’ (SimpleEvents).

Some Advant Master specific attributes, such as ProcessSection and UncerationTagTime, can be added to the list. These attributes are found in the Columns tab in the configuration view of the list. Lists with a look and feel similar to legacy product AdvaCommand can be achieved by manually importing list

configurations from the file <*installation folder* > \ *importExamples* \ *MB300AeDefaultListConfigurations.afw*.

Defining Event Treatment

The Event Treatment for Advant Master objects are defined during the engineering phase of the controller. When you build the process objects in the controller you assign values to special parameters in the DB-element that are important for the event handling. These parameters are mentioned in many “Advant OCS documents” under the collective designation “Event-Treat-pointers”. They are used, for example, to determine whether a change of status is to be handled as an alarm or an event. It is also possible to determine color priority and whether an acoustic signal is to be activated, separate handling of changes 0 to 1 and 1 to 0 etc.

Different types of events from a process object are assembled in groups, which are allocated a pointer, with which you determines the event handling of the group. For example, signal errors in an analog input are collected in a group with a pointer ERR_TR. Passages of limits Low and High (L1 and H1) are another group with a pointer LIM1_TR. A third group of passages of limits Low low and High high (L2 and H2) have a pointer LIM2_TR. The event handling, within a group, is the same in principle, but each individual event has its own text in the presentations.

Each pointer has been allocated a default value on delivery (a norm value). In many cases this selection is satisfactory and permits you to start the system without further adjustment.

Figure 21 shows the Event Treat pointers for a digital input.

The pointers are given a number, which points out a standard alarm and event handling description (on the RTA Board in the Connectivity Server).

The Event Treat pointers 1-18 are, with standard handling, ready for use and cannot be changed. Event Treat pointers 20-300 can be used by the user, without restriction, to adapt the event handling to the requirements of the process concerned.

Standard pointers adapted to the individual Object types are given in Appendix A, Event Treat Elements.

It is important that all RTA Boards in a system with redundant connectivity servers are configured identical.

Example 1: Standard Event Text

Figure 21 shows the alarm handling with a Standard event text for a DI signal object.

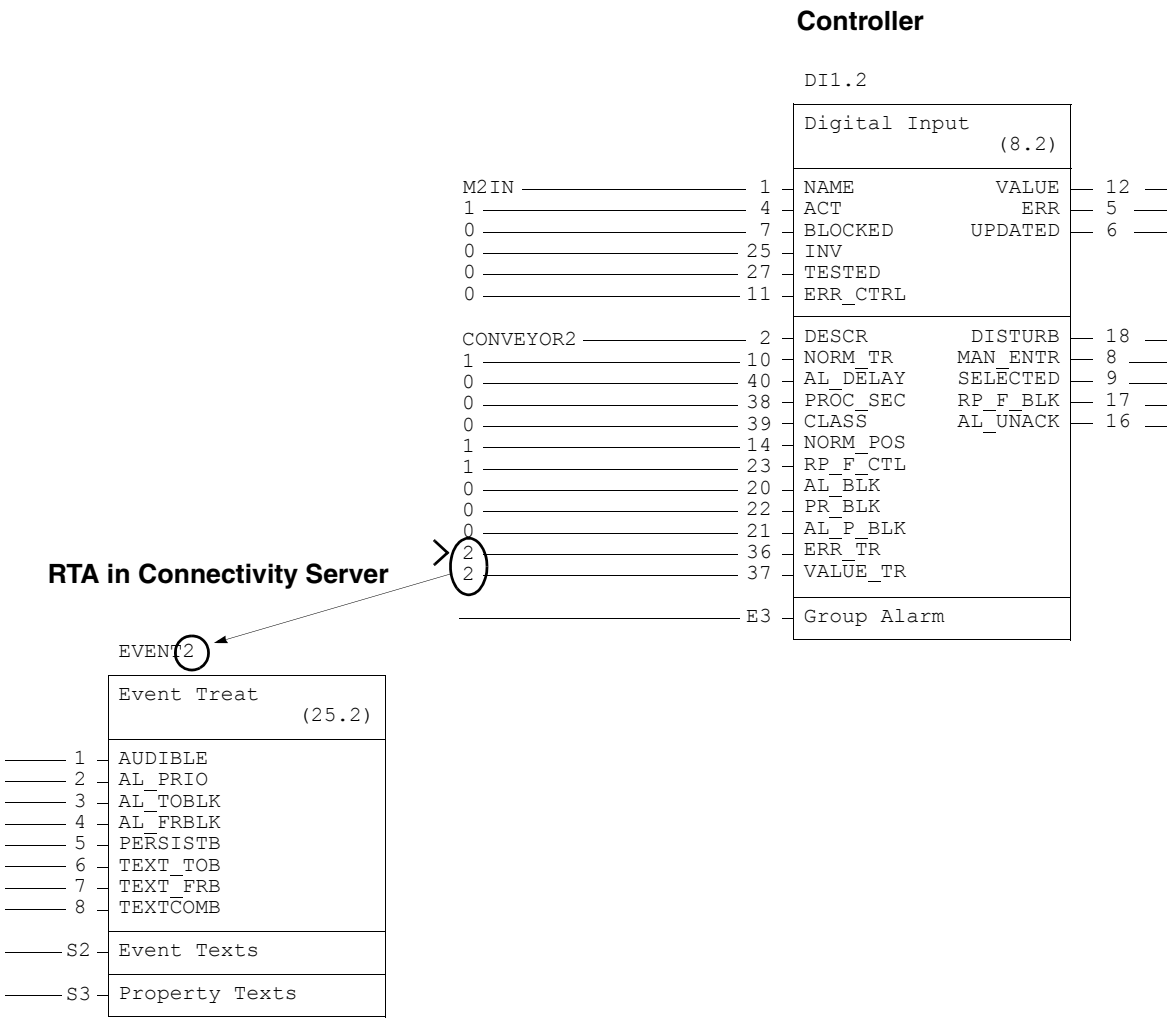


Figure 21. Linkage Between Data Base Elements

ERR_TR and VALUE_TR are pointers to elements in the Event Treat DB-element. In this example, both ERR_TR and VALUE_TR point at EVENT2.

The data base element EVENT2 has the following properties, see also Appendix A, Event Treat Elements:

AUDIBLE	1 = audible alarm
AL_PRIO	2 = Priority 2[1]
AL_TOBLK	N = acknowledgment required at alarm
AL_FRBLK	Y = blocking of acknowledgment when the alarm returns to its normal status
PERSISTB	N = remaining alarm persists in the list despite acknowledgment
TEXT_TOB	N = printout with alarm
TEXT_FRB	N = printout when the alarm returns to the normal status
TEXTCOMB	0 = standard property and standard event texts

Appendix C, Event Texts contains the standard texts for alarms and events for DI.

Example 2: How to Configure User Defined Alarm and Event Text

Following information has been built for the AI signal during the engineering phase of the controller. The signal has the name K540, the description OIL LEVEL and measurement range 0 - 1000 liters with alarm limits at 50 and 950 liters.

- EN_H2 and EN_L2 are set to 1 to enable alarm handling
- ERR_TR points at EVENT4.
- LIM_1_TR is set to 0, that is no event handling.
- LIM_2_TR points at EVENT20.

The values are shown in Figure 22.

1. This terminal also defines the alarm color in lists.

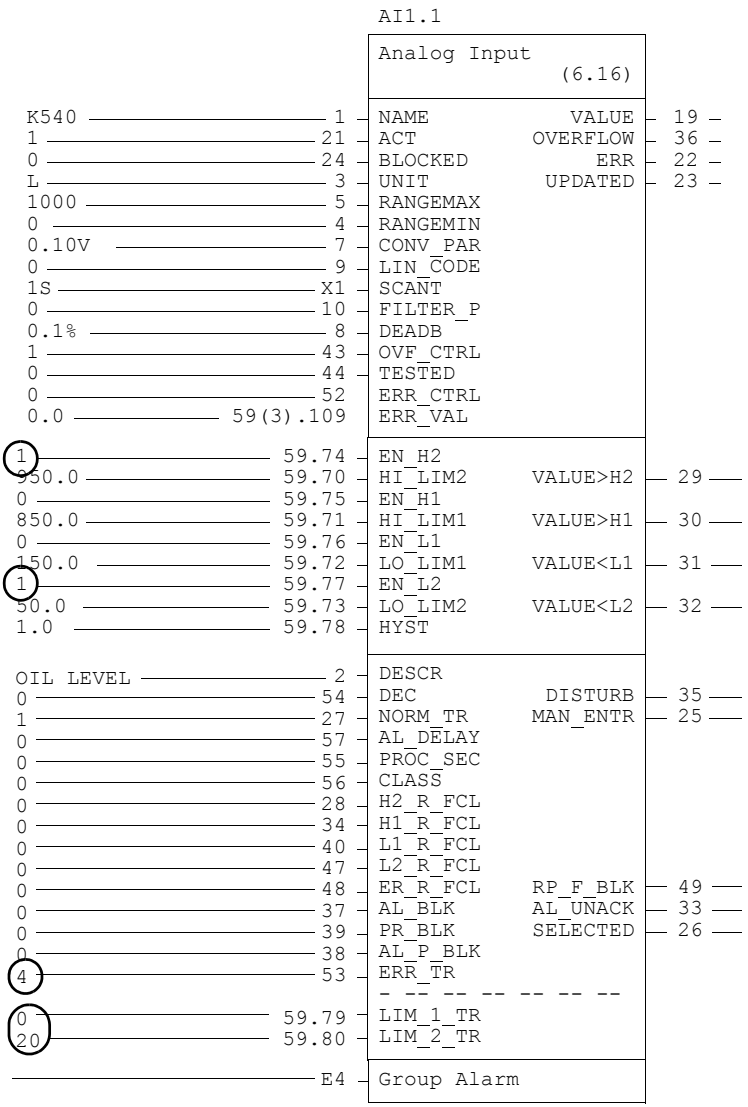


Figure 22. Data Base Element Analog Input

As presented in Appendix A, Event Treat Elements, EVENT4 has the following properties:

AUDIBLE	1 = audible alarm
AL_PRIO	2 = Priority 2
AL_TOBLK	N = acknowledgment required at alarm
AL_FRBLK	Y = no acknowledgment required when the alarm returns to its normal status
PERSISTB	N = remaining alarm persists in the list despite acknowledgment
TEXT_TOB	N = printout with alarm
TEXT_FRB	N = printout when the alarm returns to the normal status
TEXTCOMB	24 = standard property text

No pointers are set for H1 and L1. EN_H1 and EN_L1 are set to 0 since no warning limits (H1 and L1) are set in the controller.

LIM _2_TR points at a data base element, EVENT20, in the Event Treat data base on the RTA Board.

Define the alarm and event text for EVENT20.
Do the following as Application Engineer on the connectivity server.

1. Select the Control Structure in Plant Explorer.
2. Select the Network, Node, and then the RTA Board object.
3. Select the RTA Board Control aspect and when it is started, select the **RTA Board Configuration** tab.

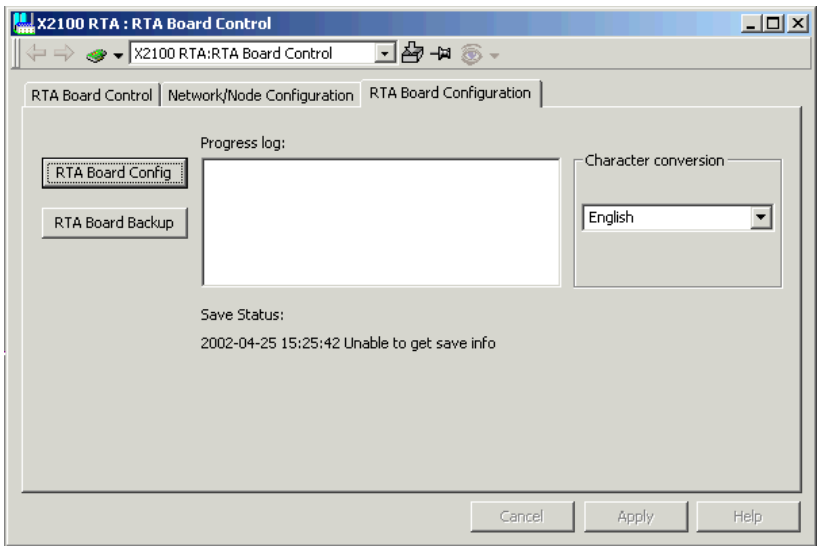


Figure 23. RTA Board Configuration

4. Click on the **RTA Board Config** button. This will startup the On-line Builder tool which is used for the configuration of the RTA Board.
5. **`MDB EVENT20`** and press **<CR>**. The data base element is displayed on the screen.
6. Enter the values as shown in Figure 24.

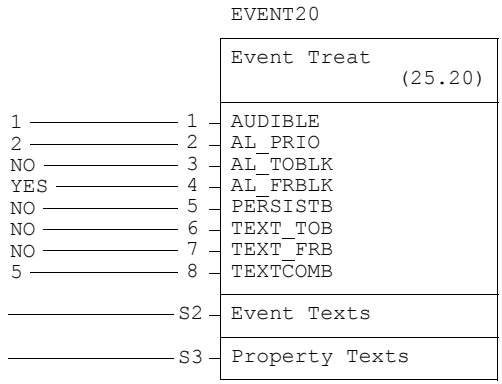


Figure 24. Changes in the Data Base Element EVENT20

The properties in the first part of EVENT20 are the same as for EVENT4 that ERR_TR points at. Another TEXTCOMB is used since property and event texts shall be user defined and not standard.

When you define the event and property texts, take note that the events are linked to predefined lines in the segment Event Texts and Property Texts. These lines are listed in the table for standard event texts for AI (see Appendix B, Event Texts).

Since only the LIM_2_TR pointer is of interest, it is selected in the following table:

Table 3. Standard Texts for the Pointer LIM_2_TR

Event	Pointer	Standard property text	Line no	Standard Event text	Line no
Upper limit H2 exceeded	LIM_2_TR	Limit H2	1	Alarm	1
Upper limit H2 re-entered	LIM_2_TR	Limit H2	1	Normal	3
Lower limit L2 exceeded	LIM_2_TR	Limit L2	2	Alarm	2
Lower limit L2 re-entered	LIM_2_TR	Limit L2	2	Normal	4

The property texts **Limit H2** on line 1 and **Limit L2** on line 2 shall be changed to **Full** and **Empty**. The standard event text **Alarm** on line 1 and 2 shall be changed to **950** l and **50** l (liters), respectively.

The standard event texts on line 3 and 4 shall be the same. These texts shall be entered in EVENT20.

7. Expand Section S2, Event Texts, and enter the text in accordance to the following:

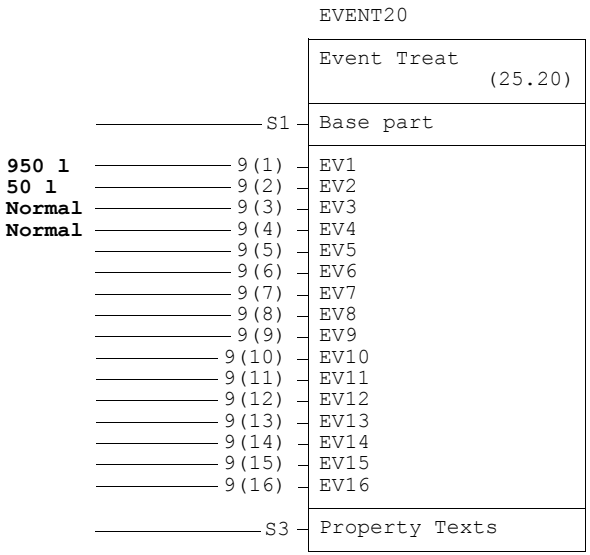


Figure 25. Population of Segment S2 in the Event Treat Element

8. Expand Section S3, Property Texts, and enter the text according to the following:

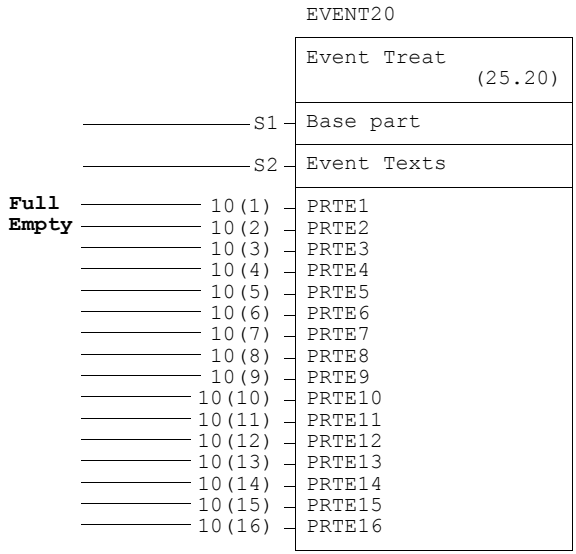


Figure 26. Population of Segment S3 in the Event Treat Element

9. Terminate the session with the following and close the window.
   - ! <CR>
   - END<CR>
   - TSESS<CR>
10. Save the RTA Board configuration according to the following steps:
    a. Go to the Control Structure, Network, Node, and RTA Board object.
    b. Select the RTA Board Control aspect and the **RTA Board Configuration** tab.
    c. Click the **Save Config** button.

The configuration of the event handling for the AI signal is now complete.

Defining Event Filter on Node Level

It is possible to filter alarm and events to be sent from selected controllers on a MB 300 network to certain connectivity servers. This can be useful when;

- Multiple connectivity servers are connected to the same MB 300 network.
- Only a selection of controllers connected to a MB 300 network are included into a 800xA system.

Instruction how to configure Event filter on Node level.

This function is configured on the RTA board in the connectivity servers.

Follow the steps below logged in as Application Engineer on the connectivity server:

1. Start the RTA board configuration from the RTA Board Control aspect.
2. Create one instance of the data base element NODE_DESCR for each controller node you want to prevent from sending events to this RTA board. Command: CRDB NODE_DESCR.
3. The following terminals should be defined:
   NETW_NO – Network number. For redundant networks, specify one of the two networks.
   NODE_NO – Node number for the controller node to be filtered.
   EVENT – 1 = Filter is active.
4. When all NODE_DESCR elements are defined, terminate the RTA board configuration session and start a RTA Board Backup (also from the RTA Board Control aspect).
5. Restart the RTA board from the RTA Board Control aspect.
6. Now the configuration is completed for this connectivity server. In case redundancy is configured for this you need to repeat the configuration on the redundant connectivity server.
7. Restart the concerned controllers to make the changes take effect. If not possible to restart the controllers, consult ABB Technical Support.

Advant Master Alarm Refresh

Using the Advant Master Alarm Refresh, it is possible to retrieve the latest alarm status from Advant Master and Safeguard Controllers, for example, after communication disturbance between Connectivity Server and Controller.

This function sends status requests called *Status Check* to the process objects in controllers. The controller replies with the actual alarm status. The Process Alarm Lists in the 800xA system is updated with new unacknowledged alarms. These alarms have the same Object name and Condition attributes as normal alarms, but the Message attribute is *Statuscheck*.

The Alarm Refresh function automatically starts when alarms from a Controller may have been lost. The following scenarios trigger Alarm Refresh:

- Connectivity server start up, for example, after a power cut or blackout.
- Event collector start up.
- RTA unit start up.
- Controller restart.
- After all types of communication failure between 800xA Connectivity Server and Controller.

Alarm Refresh for a specific node can also be manually invoked. For more information, refer to Configuring the Alarm Refresh on page 66.

All uploaded and implemented process objects with alarm support are handled by the Advant Master Alarm Refresh function. For active alarms, it scans the objects under the **Controller Node** object in **Control Structure** for process objects to be refreshed. The following controller types are tested and supported by this function.

- AC 400 Series
- MP200/1
- Safeguard 400
- Safeguard 9000

Configuring the Alarm Refresh

By default, the Alarm Refresh function is disabled. Execute the following steps to activate this function.

1. Open the file *AdvMbAeOPCServer.csv* from the location <*installation folder*>*bin* in the Connectivity Server, in a Notepad.
   a. Find the row beginning with *RefreshActDef*.
   b. Change the second argument on the line from 0 to 1.
   c. Save the file and restart the Connectivity Server, or just the Event Collector.

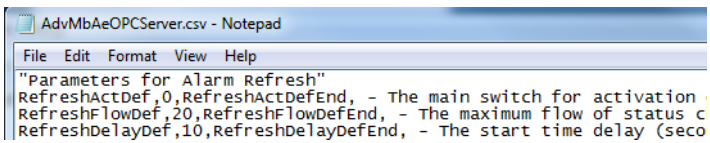


Figure 27. The AdvMbAeOPCServer.csv file

   d. Repeat this step on all Advant Master Connectivity Servers. The configuration file may be write protected. In that case, reset the Read Only attribute before saving the file.

2. The intensity of the refresh request sent to the controller can be customized for each Connectivity Server, to adapt the controller load for the actual circumstances. This is done using the **RefreshFlowDef** parameter in the configuration file. The maximum flow of requests per second can be set in the interval 4 to 200 (objects). By default, the flow of requests per second is 20. Restart the Connectivity Server or the concerned Event Collector Service Provider if the parameters in this file are changed.

   The time lagging (in seconds) before starting the Alarm Refresh, can also be set for each Connectivity Server to equalize the controller load. This is done using the **RefreshDelayDef** parameter in the configuration file. By default, the time is 10 seconds.

3. It is possible to select the **Support Refresh** check box in the **Alarm Collection Definition** aspect in **Library Structure > Advant Master Collection** to allow the Event Collectors to use the Alarm Refresh function at Connectivity Server failovers and when the Alarm Manager is restarted. This function erases the old alarms in an alarm list before a refresh is performed. However, the MMCX1, MMCX2, MMCX3, Engineered drive, Standard drive and MOTCONI object types in most Controller releases/revisions, do not support the Advant Master Alarm Refresh function. It is thereby not recommended to check **Support Refresh** if these object types are used.

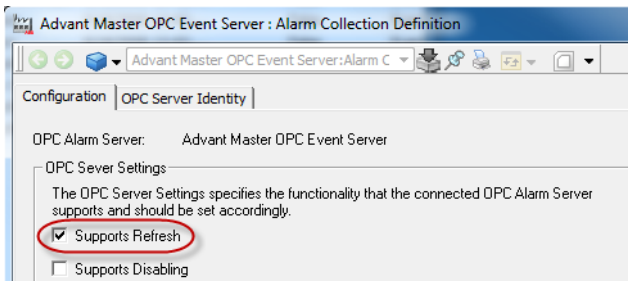


Figure 28. Alarm Refresh configuration

The text *Statuscheck* presented in the Message attribute is an NLS resource available in RTA unit, Logical file 31, record 132. It can be customized if needed. A maximum of 11 characters are allowed.

Alarm Refresh for a specific node can be manually invoked by setting the property **General Properties:FORCE_REFRESH** to *True* on the node object in the **Control Structure**.

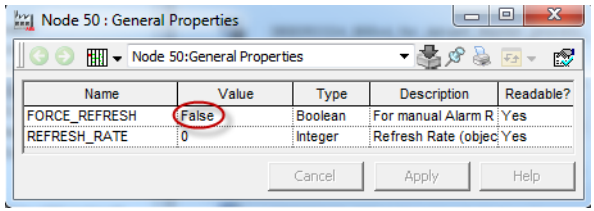


Figure 29. General Properties

The property will automatically be reset when refresh is completed. This aspect can be copied from **Object Type Structure** to the controller nodes in **Control Structure** if needed.

The **RefreshFlowDef** parameter in the configuration file can be overridden individually, for each controller node by setting the property **General Properties:REFRESH_RATE**. The following values can be set:

- = 0: Use default from the configuration file. This is the default setting.
- > 0 (positive value): Use this value as refresh rate (objects/s) for this controller.
- < 0 (negative value): Exclude this node from the Alarm Refresh function.

Controller behavior to consider

- The *Active Time* reported is the actual time of the *Status Check* response, because the controller cannot store event times.
- The refresh intensity impact on the controller has to be considered.

  A very high intensity can cause high load in the controller for a short time, but a very low value will delay the Alarm Refresh function considerably.

  The default value (20 object requests/second) is suitable for high loaded controllers. For AC 450 controllers with around 60% average CPU load, a value of 50 requests/second can be chosen.

  For further adjustment, adjust the time lagging (in seconds) before Alarm Refresh is started, for each Connectivity Server. For example, setting the **RefreshDelayDef** parameter to 0 in one Connectivity Server and 60 in the

other will minimize the risk of simultaneously loading influence of the controller.

- Only process object in alarm state responds on Alarm Refresh, that is, the conditions that changed to normal state during the time of communication failure may still be visible in alarm lists, although the alarm state has gone to normal during the time of communication failure.
- All Analog Input and Digital Input process objects with the DB element property NORM_TR set to 0, and Event treatment pointer (for example, LIM_1_TR) set to a value greater than 0, may generate new alarms as a result of *Status Check*. These alarms will remain in alarm list until manually removed. To avoid this, it is recommended to set Event treatment pointer to 0 for all objects having NORM_TR = 0 in controller database.
- AO and DO objects do not support Alarm Refresh, that is, signal error on these object types will not be refreshed.
- Objects blocked for alarm (with Alarm_Blk = True or Alarm_P_Blk = True) are not treated.
- All Controller versions/revisions do not support Alarm refresh for the following object types: MMCX1, MMCX2, MMCX3, Engineered drive, Standard drive and MOTCONI. Refer to the Controller release documentation for information if support has been added for any of these object types.

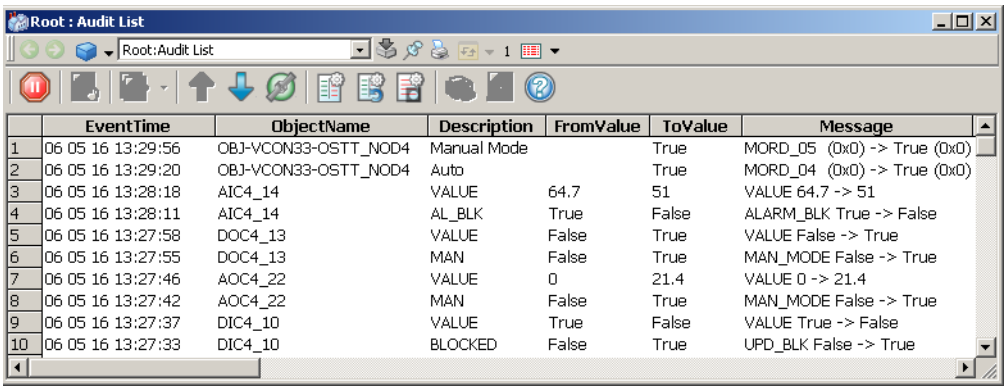
Alarm Hiding

800xA for Advant Master supports hiding of alarms. Hidden alarms does not appear in alarm lists. Hiding of individual conditions is possible if the Alarm Conditions aspect is used. The condition names added to Alarm Condition are dependent of the language selected on the RTA Board. Active hiding condition is indicated in the Alarm Control button in faceplates. For more information on alarm hiding, refer to *System 800xA, Configuration (3BDS011222*).*

Audit List for Operator Actions

An Audit List can display audit logs for operator actions. It can be a useful help to, for example, see what changes the last operator on shift made in the system.

The logs are displayed in an alarm and event list, which is configured to show operator actions. Lock operations are by default not logged in Audit Lists. For configuration and creation of an Operator Actions List see *System 800xA, Administration and Security (3BSE037410*)*. See Appendix D, Control Aspect for information about how to customize the description column.



	EventTime	ObjectName	Description	FromValue	ToValue	Message
1	06 05 16 13:29:56	OBJ-VCON33-OSTT_NOD4	Manual Mode		True	MORD_05 (0x0) -> True (0x0)
2	06 05 16 13:29:20	OBJ-VCON33-OSTT_NOD4	Auto		True	MORD_04 (0x0) -> True (0x0)
3	06 05 16 13:28:18	AIC4_14	VALUE	64.7	51	VALUE 64.7 -> 51
4	06 05 16 13:28:11	AIC4_14	AL_BLK	True	False	ALARM_BLK True -> False
5	06 05 16 13:27:58	DOC4_13	VALUE	False	True	VALUE False -> True
6	06 05 16 13:27:55	DOC4_13	MAN	False	True	MAN_MODE False -> True
7	06 05 16 13:27:46	AOC4_22	VALUE	0	21.4	VALUE 0 -> 21.4
8	06 05 16 13:27:42	AOC4_22	MAN	False	True	MAN_MODE False -> True
9	06 05 16 13:27:37	DIC4_10	VALUE	True	False	VALUE True -> False
10	06 05 16 13:27:33	DIC4_10	BLOCKED	False	True	UPD_BLK False -> True

Figure 30. Example of a customized Audit List for Operator Actions

External Alarm

The External Alarm function makes it possible to activate and deactivate an external alarm notification object in the controller. To prevent write conflicts when outputs are concurrently activated and deactivated, it is only supported to use object type DAT(B).

Events to 3’rd party clients

All access from 3’rd party OPC AE clients should be connected through “ABB OPC AE Server for 800xA”, ProgId ABB.OPCEventServer.1.

To enable sending of events to clients, please refer to Appendix H, Special Configuration.

Section 5 History Logs

When configuring Log Configurations in Basic History there are three collection methods to choose from:

- TTD Logging (See Time Tagged Data (TTD) on page 72.)
- OPC Direct Logging with controller generic cyclic (1, 3 and 9 seconds) subscription (See OPC Direct Logging - 1, 3 and 9 Seconds on page 94.)
- OPC Direct Logging without controller generic cyclic subscription (See OPC Direct Logging - Other Storage Intervals on page 94.)

Consider the following to optimize performance:

- Best availability, accuracy, and RTA performance is obtained when making use of TTD logs in the controller. The History Server can read from TTD logs much more efficiently compared to using direct OPC property subscriptions.
- Use OPC Direct Logging with cyclic rate 9, 3 or 1 seconds (preferably 9s) and only for properties that support controller generic cyclical subscriptions if TTD logging cannot be used. See Section 7, Data Transfer via OPC DA for more details and alternative subscription rates since the rules for data subscriptions also applies to OPC Direct Logging subscriptios.
- Do not read recent History values too frequent when the log's data source is TTD.

Properly configured time synchronization is a key to get stability and performance in logging values from the Advant Master controllers.

Time Tagged Data (TTD)

The configuration of TTD is based on the History Configuration Aspect System which is an integrated part of 800xA System. This section describes mainly the Aspect Categories for TTD. The History Configuration Aspect System user interfaces in general are described in detail in the *System 800xA, Configuration (3BDS011222*).*

The AC 400 Series Controllers have a Historical logging feature called Time Tagged Data (TTD). The framework for this, called log groups, must be defined during the Controller Engineering phase. Adding logs to the log groups can then be made from the workplace.

Before going into the details on how to create the TTD logs in the controllers, here is a short description of some of the terms and functionality of the AC 400 TTD logs:

A **log** is a collection of time stamped (tagged) data.

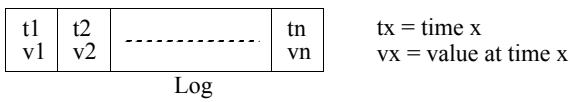


Figure 31. A Collection of Time Stamped (tagged) Data

The process or system object from which the data originates is called the data source.

A log also contains information about the data source.

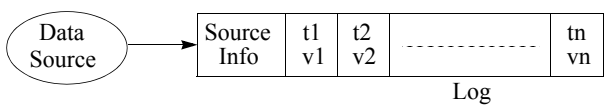


Figure 32. Information About the Data Source

A log is always part of a log group, one log group in TTD can manage up to 127 logs. There are up to 15 log groups. The log's sequence number within the log group is called log index. The log group keeps the important parameters, storage interval

and storage size (in controller documentation storage size is called log period) common to all logs in the log group.

The following History Configuration Aspect System user interfaces, which are specific for 800xA for Advant Master, are described below:

- The TTD Source Aspect
- The Log Template aspect, the TTD log tab
- The TTD Node Configuration aspect
- The Log Configuration aspect, the TTD log tab is the same view as the Log Template aspect.

History Primary Logging should preferably be done by using TTD as data source. TTD normally causes less impact on the system than OPC logging on a cyclic rate.

Too heavy History logging may cause unnecessary load in the control network and controllers in such a way that the operator interface may get slow, or even stops working properly.

The minimum length recommended for TTD Logs in the controller is 40 minutes.

TTD requires a time synchronization accuracy between the 800xA system and the MB300 network that is better than 20 seconds to work properly.

The TTD Source Aspect

The Configuration Wizard creates this aspect when adding an RTA board. The aspect defines the service groups for TTD Config Source and TTD Data Source. TTD Source is placed on the MB 300 Network object and does normally not need to be changed.

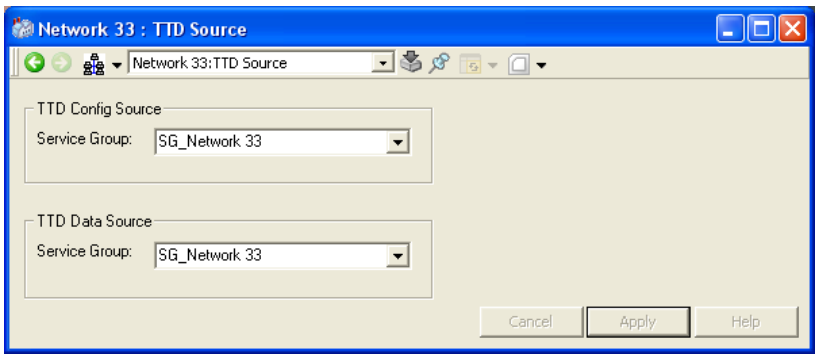


Figure 33. TTD Source Aspect

The Log Template Aspect

A Log Template is a re-usable template representing one log group. If you modify an existing Log Template, the changes will affect existing logs that have been created using that Log Template.You define the Log template via the Log Template aspect.

To create this aspect you preferably invoke an upload from the Synchronize View on TTD Node Configuration aspect. This creates a Log Template aspect for each TTD log Group on the corresponding AC 400 Series Controller object.

The aspect has two tabs:

- The **Log Definition** tab
- The **TTD Data Collection** tab

Property log Template area

You select the log to define in the **Property log Template** area on the left side.

You can add new Log templates by clicking the **Add** button and delete Log templates by clicking **Delete**. You quit the Log template definition without storing anything by clicking **Cancel**.

You save the Log template definition by clicking the **Apply** button. On-line help is available by clicking the **Help** button.

Log Definition Tab

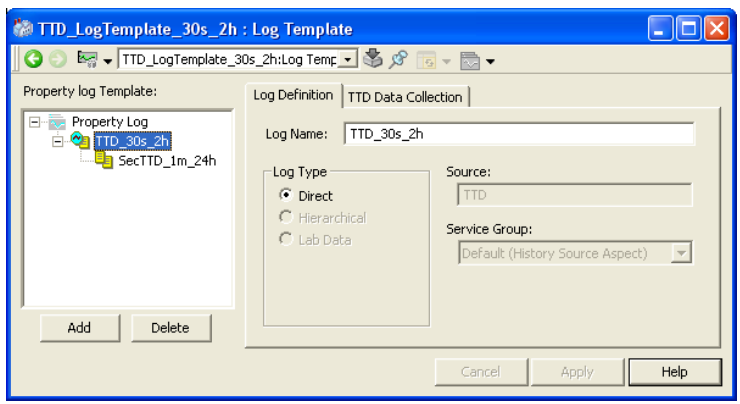


Figure 34. Log Template - Log Definition Tab

Log Name. Enter the log name in this field. We recommend that you include the following parts in the name:

- "TTD" for TTD logs
- the storage interval for the log (30s)
- the total log capacity (2h) in the log template name.

This makes it easier to find the correct log template.

Log Type. Log type is set to **Direct** for primary logs, and **Hierarchical** for hierarchical logs. The Log type and Source definitions are defined when you create the log template.

TTD Data Collection Tab

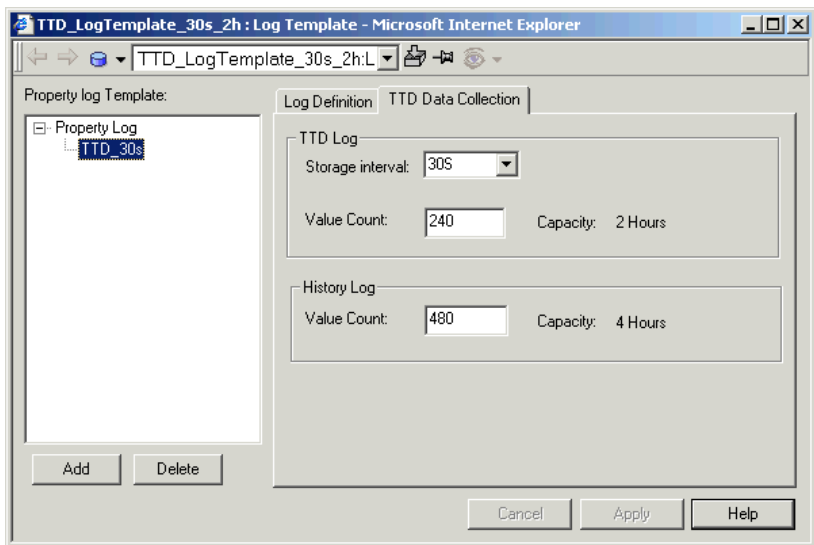


Figure 35. Log Template - TTD Data Collection Tab

TTD Log Storage Interval. This is the storage interval for the TTD log in the controller. Select the log storage interval for the TTD log in the drop-down menu.

TTD Log Value Count. This is the storage size (number of stored values) in the TTD log in the controller. The Capacity (in time) is automatically calculated and presented to the right of the entry field.

The TTD log storage interval and the TTD log value count must match a Log Group in the controller.

History Log Value Count. This is the storage size (number of stored values) in the history server. The default size of the History log is double the size of the TTD log. If you want to change the History log size, you enter the number of stored values in the History log in the **Value Count** field. The Capacity (in time) is automatically calculated and presented to the right of the entry field.

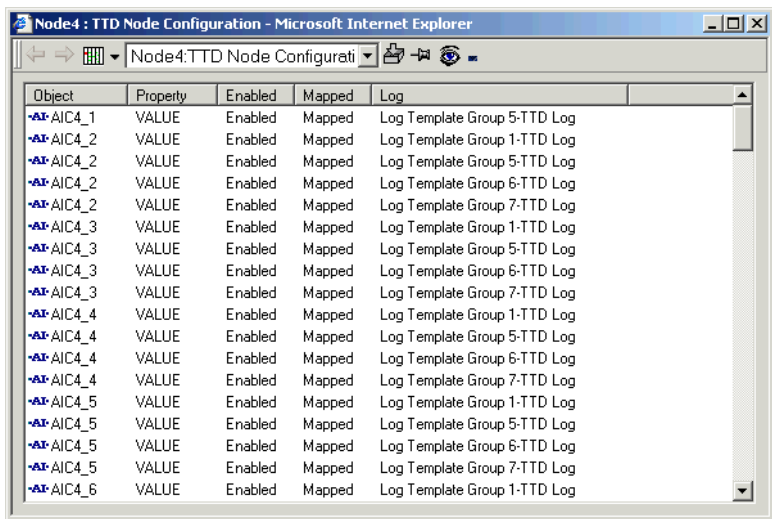
TTD Node Configuration Aspect

TTD Node Configuration is an aspect of the AC 400 Series Controller object in Control Structure. It has three different views:

- **Main View -** presents the TTD logs in the Control Structure
- **Controller View** - presents the structure of the Log Groups
- **Synchronize View** - used for upload and download of TTD configurations.

Main View

The main view of the TTD Node Configuration aspect presents the TTD logs in the Control Structure.s



Object	Property	Enabled	Mapped	Log
AIC4_1	VALUE	Enabled	Mapped	Log Template Group 5-TTD Log
AIC4_2	VALUE	Enabled	Mapped	Log Template Group 1-TTD Log
AIC4_2	VALUE	Enabled	Mapped	Log Template Group 5-TTD Log
AIC4_2	VALUE	Enabled	Mapped	Log Template Group 6-TTD Log
AIC4_2	VALUE	Enabled	Mapped	Log Template Group 7-TTD Log
AIC4_3	VALUE	Enabled	Mapped	Log Template Group 1-TTD Log
AIC4_3	VALUE	Enabled	Mapped	Log Template Group 5-TTD Log
AIC4_3	VALUE	Enabled	Mapped	Log Template Group 6-TTD Log
AIC4_3	VALUE	Enabled	Mapped	Log Template Group 7-TTD Log
AIC4_4	VALUE	Enabled	Mapped	Log Template Group 1-TTD Log
AIC4_4	VALUE	Enabled	Mapped	Log Template Group 5-TTD Log
AIC4_4	VALUE	Enabled	Mapped	Log Template Group 6-TTD Log
AIC4_4	VALUE	Enabled	Mapped	Log Template Group 7-TTD Log
AIC4_5	VALUE	Enabled	Mapped	Log Template Group 1-TTD Log
AIC4_5	VALUE	Enabled	Mapped	Log Template Group 5-TTD Log
AIC4_5	VALUE	Enabled	Mapped	Log Template Group 6-TTD Log
AIC4_5	VALUE	Enabled	Mapped	Log Template Group 7-TTD Log
AIC4_6	VALUE	Enabled	Mapped	Log Template Group 1-TTD Log

Figure 36. TTD Node Configuration, Main View

Object. Name of the object that has a property included in a TTD log.

Property. Object property that is included in the TTD log.

Enabled. Indicates if the TTD log is enabled or disabled in the History server.

Mapped. Indicates if the TTD log is mapped to a TTD Log Group (in the controller). This will normally be set to Mapped. Unmapped indicates that a download to the controller is required.

Log. TTD log name (defined in the Log Template aspect that represents the Log Group) that the object property is included in.

Controller View

The Controller View of the TTD Node Configuration aspect has two lists:

- one overview list with all groups in the controller node, see Figure 37.
- one list for a selected Log Group, see Figure 38.

The following columns are presented when you have selected the controller node in the left window:

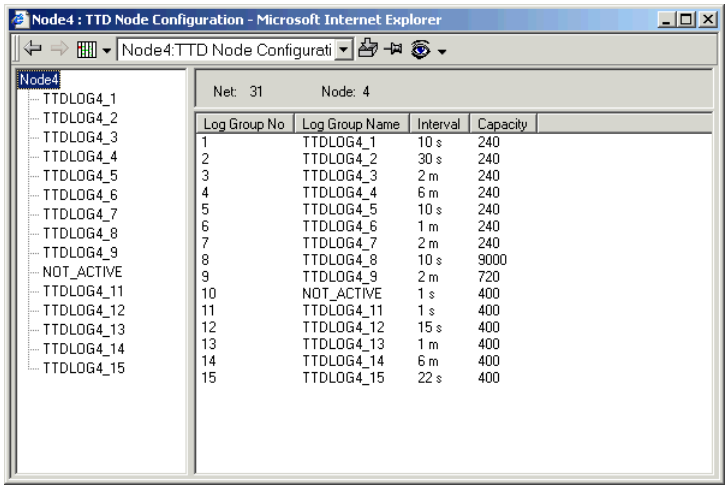


Figure 37. TTD Node Configuration: Controller View, Log Groups

- **Log Group No** - Log Group number 1 - 15.
- **Log Group Name** - The name of the TTD Log Group.
- **Interval** - The storage interval displayed in suitable time scale.
- **Capacity** - The number of stored data items before they are overwritten (circular log).

The second layout of the Controller View, showing the details for a selected Log Group has the following columns in the right window, see Figure 38.

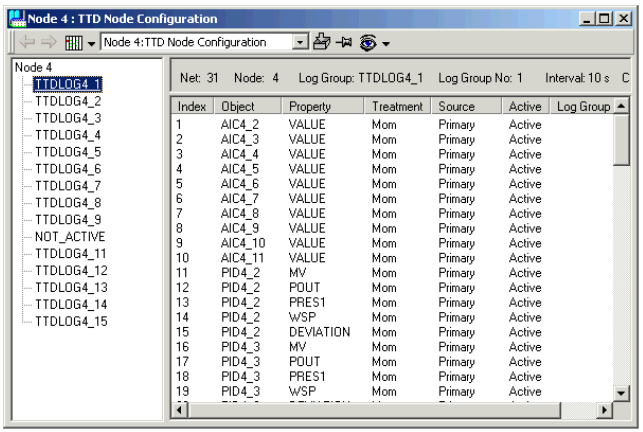


Figure 38. TTD Node Configuration: Controller View, Details About one Log Group

- **Index** - Index in the Log Group 1- 127.
- **Object** - The name of the logged object.
- **Property** - The name of the logged object property.
- **Treatment** - The logged value could be calculated from the real-time value. Four calculations (treatments) are available: **Mom** (momentary), **Mean** (average), **Max** (maximum) and **Min** (minimum). The treatment should always be **Mom** for primary logs.
- **Source** - Source shows if the log is primary or secondary. Only primary logs are handled by history server, secondary logs are ignored.
- **Active** - The state of the log - Active or Not active (shown as blank in the list).

The header above the list of logs contains information about Log group address, the storage interval and capacity.

Synchronize View

The Synchronize view of the TTD Node Configuration aspect has the following functionality:

- Upload of the current TTD groups and logs from the selected controller.
- Download of new TTD log configurations into the controller.

Upload of TTD Logs from Controller. To load the existing TTD logs into the History Configuration Aspect System,
you use the synchronization view of the TTD Node Configuration aspect.
The upload process creates all necessary Log Template and Log Configuration aspects. Click **Upload** and wait. The progress is presented in the text window, see Figure 39.

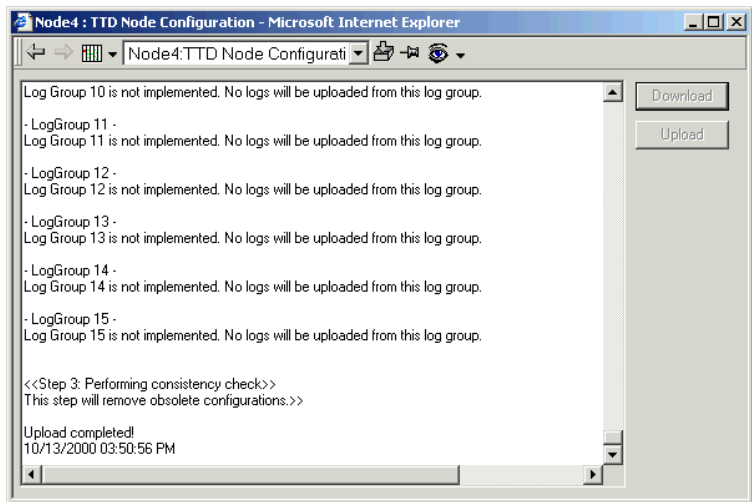


Figure 39. Upload of TTD Logs

The result of a completed Upload is as follows:

- A Log Template aspect for each type of TTD Log Group is created on the corresponding AC450 and AC410 Controller object.
- A Log Configuration aspect is created on each logged object. It has the primary log being a TTD log configured according to the information read from the controller and with an extension into a corresponding hierarchical log twice as large as the primary log. To extend it, change the corresponding Log template and it will affect all logs belonging to that Log Template.
- The hierarchical logs are disregarded.

Download of new log configurations into the Controller.

You can also do Download for an individual log from the Log configuration aspect on the process object.

To download and activate new TTD logs you can use the synchronization view of the TTD Node Configuration aspect. Click **Download** to the right and wait. The Match Existing dialog box appears, see Figure 40.

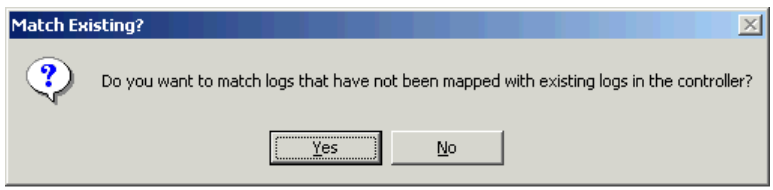


Figure 40. Match Existing

Click **Yes** for logs that have not been mapped to a TTD log in the controller. The download function will try to find a suitable log in the controller to map the downloaded log to. Click **No** when a new log will be created in the controller for each configured un-mapped log.

The progress is logged in the text window, see Figure 41.

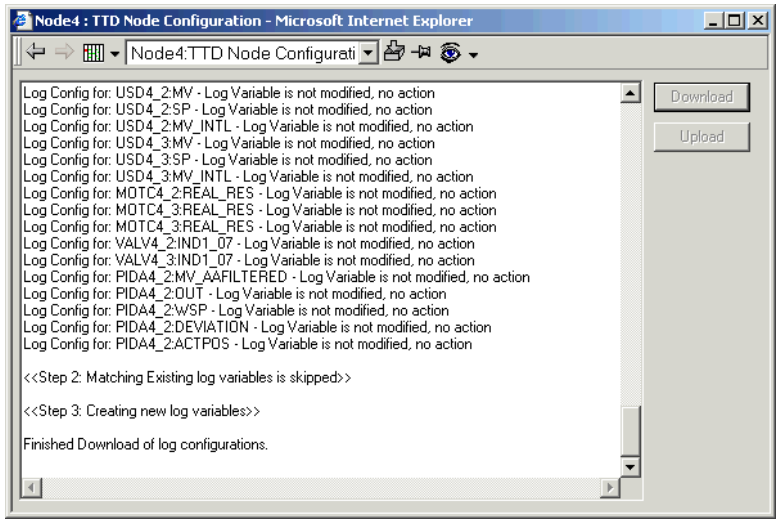


Figure 41. Downloading the new TTD logs

When the download is finished, you can check the result in the log on the screen but you can also bring up the Main View to check specific logs.

Working with TTD logs

We have now discussed each configuration aspect. In this section you will find three examples of how to make the configuration. The assumption is that you have a Control Structure for an AC 400 Series controller and that the controller has configured TTD log Groups with some logs.

1. The first example shows how to view the current TTD logs in the controller.
2. The second example shows how you prepare the configuration of new TTD logs by creating Log Templates for them.

3. The third example utilizes the Log templates when you create new logs and download them to the controllers.

The recommended method to configure the History Aspect System if the controller already is configured by terms of TTD groups and logs is however to perform Upload. This is done from the Synchronize View of the TTD Node Configuration aspect of the controller objects. This action creates all necessary objects and aspects.

Example 1: View the TTD logs in the Controllers

1. Go to the Control Structure in Plant Explorer.
2. Click on the object representing the controller you want to work with. Select the TTD Node Configuration aspect.
3. Select the Controller View. See Figure 42.

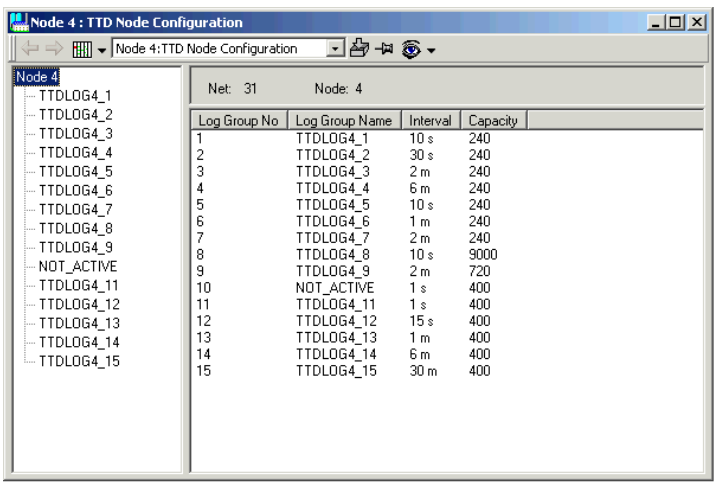


Figure 42. TTD Node Configuration: Controller View, Log Groups

To the left you see a list with the existing Log Groups, and to the right a storage interval (Interval) and storage size (Capacity) for each Log Group. If you select one of the groups you can see the existing logs in that group to the right, see also

Figure 43.

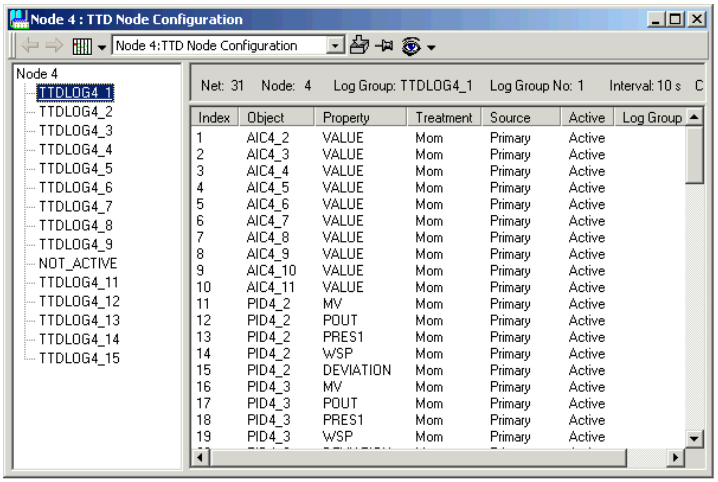


Figure 43. TTD Node Configuration: Controller View, details about one Log Group

Example 2: Create a Log Template for a TTD log

We want to create a log template for a TTD log with storage interval 30 seconds and a storage size of 2 hours (240 values). We already know (from example 1) that a suitable TTD log group exists in this controller and there are at least one free index in the group. The TTD log shall be stored and extended in the History Configuration Aspect System to a storage capacity of 4 hours.

Furthermore, a hierarchical log shall be created within the History Configuration Aspect System which aggregates the primary information into an average, stored with 1 minutes interval over 1 day (1440 values) aggregated as average.

1. Select the Library Structure in Plant Explorer.
2. Create a new object of the History Log Template type. Give it a proper Name and Description. Let the name reflect to the TTD log template and the storage interval and size. It makes it easier to select a log template during log creation.

It is recommended to save the log template in the Library Structure, History Log Template Object Type.

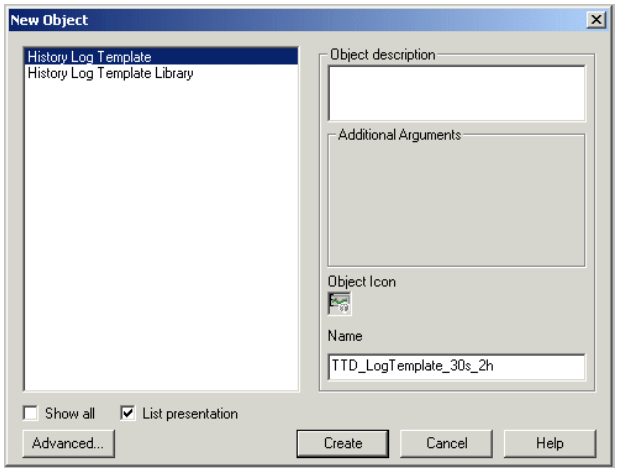


Figure 44. Creating the Log Template Object

3. An aspect of the category Log Template is created by default. Select it in the preview window or as a pop-up (double click on it).

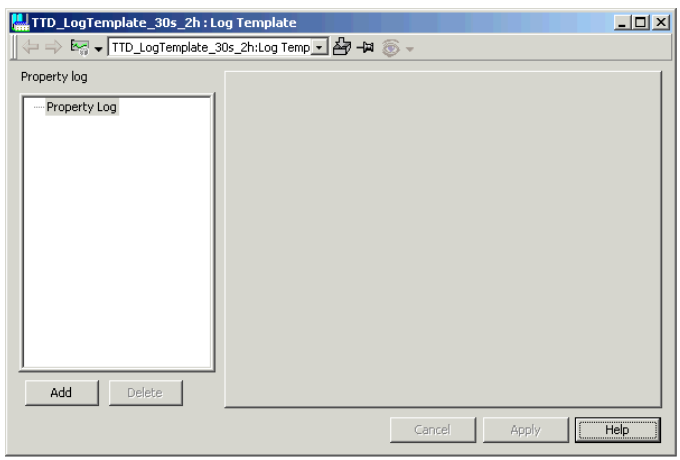


Figure 45. Log Template Aspect

4. Start creating the new TTD log. Select **Property Log** and click **Add**. See Figure 45. The New Log Template dialog is displayed, see Figure 46.

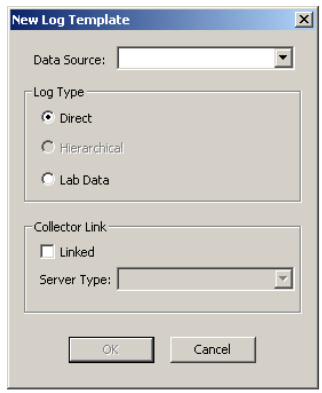


Figure 46. New Log Template

5. Select **TTD** in the **Data Source** drop-down menu. In the **Log Type** area, select the **Direct** radio button. Click **OK** to save the settings.

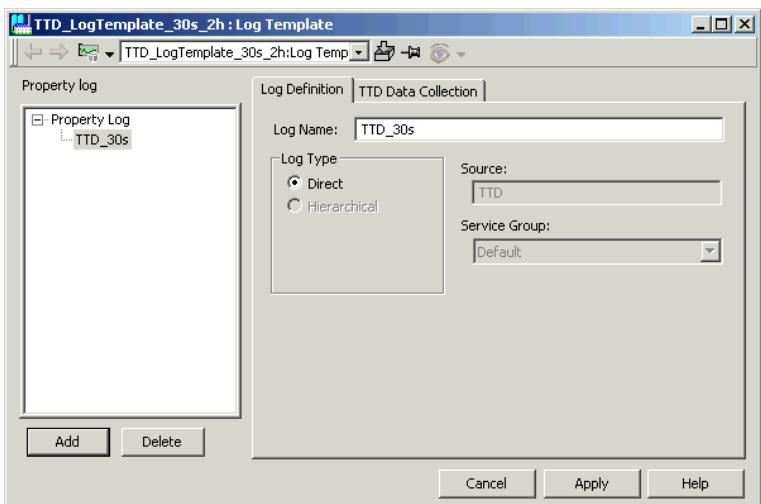


Figure 47. Creating TTD Log Templates

6. Enter the name of the new TTD log (TTD_30s in the example above). Name the logs according to the type of log (TTD in this case), and the storage interval (30s). See Figure 47.

7. Select the **TTD Data Collection** tab and enter the "TTD log storage interval" 30 seconds and a storage "TTD log value count" of 240 values (2 hours capacity), see Figure 48.

   The storage size in History Aspect System is defined by **History Log Value Count.** Choose twice the size of the TTD log, in this case 480 values (4 hours capacity).

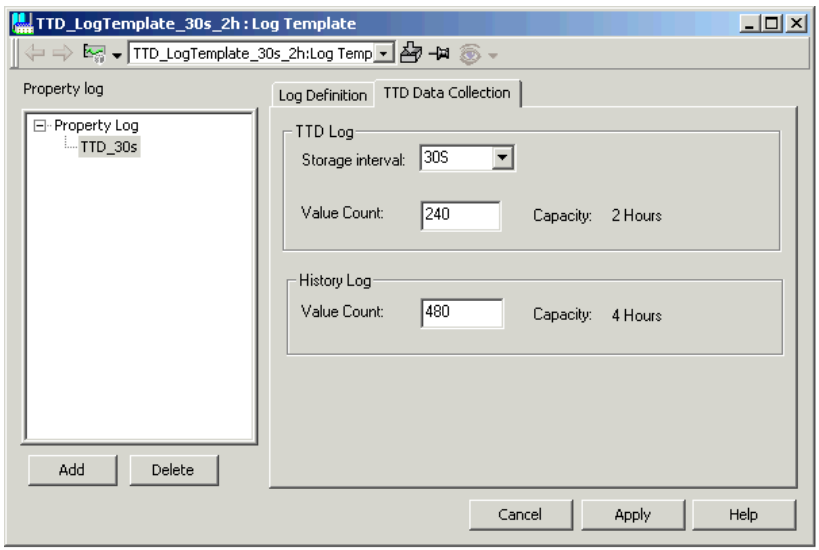


Figure 48. Log Configuration - TTD Data collection

8. Save the TTD Data Collection parameters (click **Apply**) and go back to the **Log Definition** tab to enter the definition of one hierarchical log.

9. Select the TTD_30s log and click the **Add** button and **OK** button.

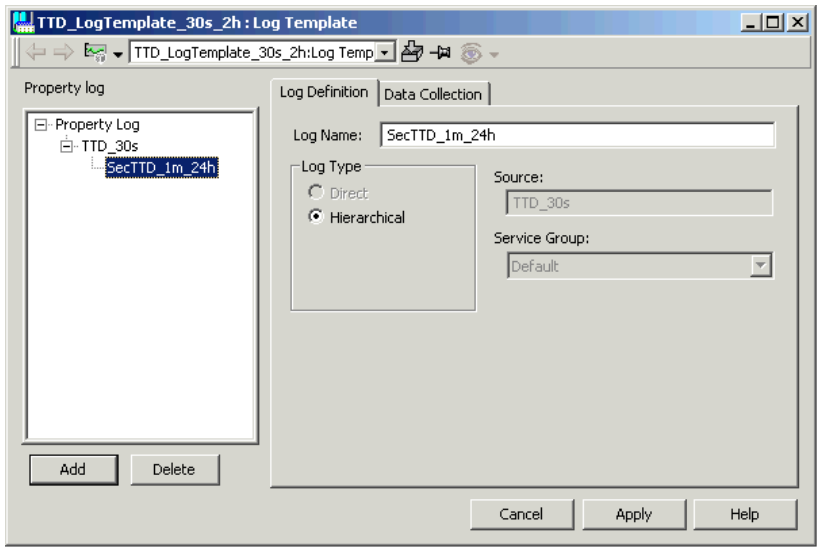


Figure 49. Creating the Hierarchical Log

10. Enter the following for the hierarchical log and save it:
    - Log name: SecTTD_1m_24h
    - Log type: Hierarchical

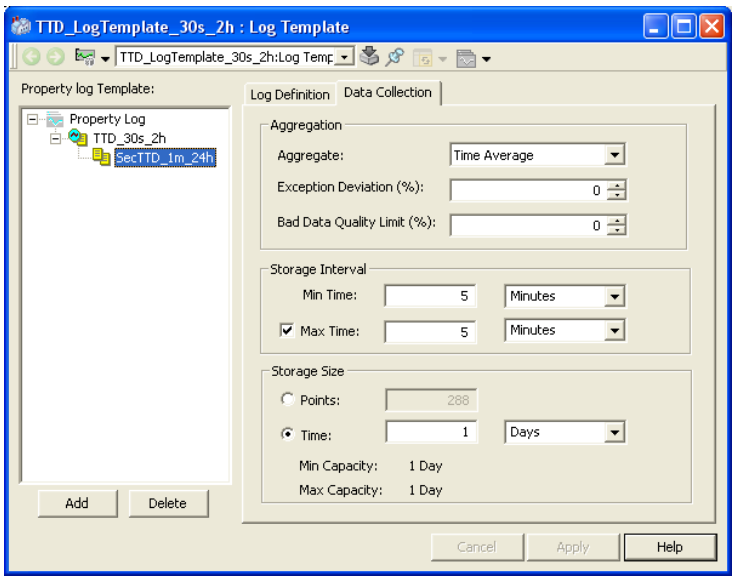


Figure 50. Hierarchical Log, Filling in Capacity

11. Select the **Data Collection** tab, enter the following data and click **Apply**.
    - Aggregate: Time Average
    - Exception Deviation = 0
    - Bad Data Quality Limit = 0
    - Min Time: 5 minutes
    - Max Time: 5 minutes
    - Storage Size (Time): 24 hours

The log template is now finished and ready to use.

Example 3: Create TTD logs from workplace and Download them

1. Go to the Control Structure in Plant Explorer.
2. Click on the object representing the controller that you want to work with. Browse down to the process object that shall be placed in the TTD log.
3. Select the process object. Add the Log Configuration aspect. Give it a proper Name and Description and save it.
4. Select the aspect and then - in the preview area - select the object and click **Add Property Log**.

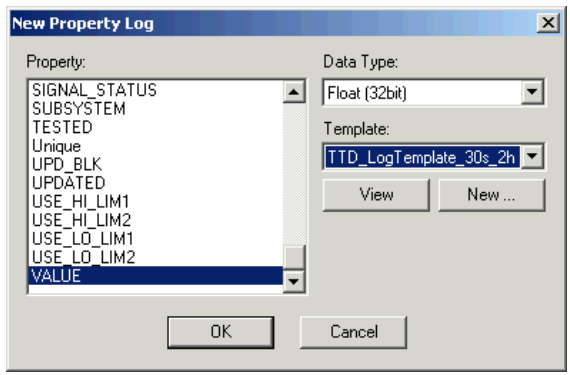


Figure 51. New Property Log Dialog

5. Select **VALUE** in the **Property** list and the **TTD_LogTemplate_30s_24h:Log Template** in the **Template** drop-down menu. Click **OK**.

6. The preview area now looks like Figure 52.

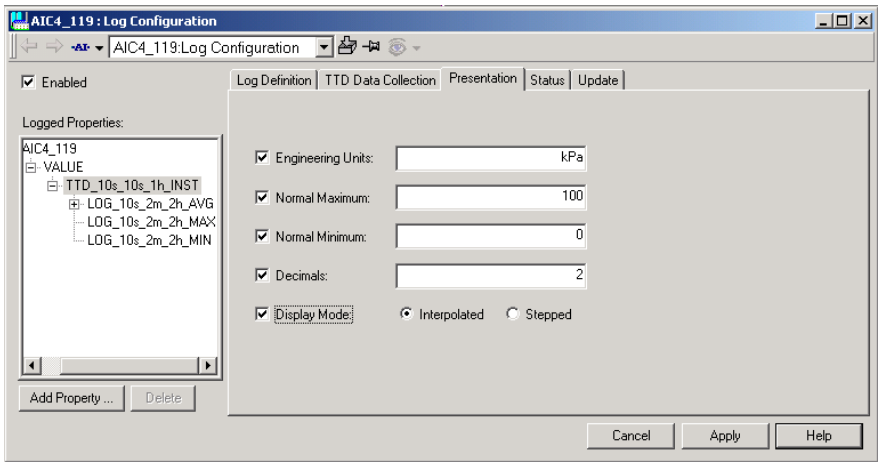


Figure 52. Log Configuration - presentation

7. Go to the **Presentation** tab and enter the following. Save it by clicking **Apply.**
   - Engineering Units: KPa
   - Normal Maximum: 200
   - Normal Minimum: 0
   - Decimals: 2
   - Display Mode: Interpolated.
8. Repeat Step 2 to Step 7 for each log that you want to create.
9. Now it is time to download all the new logs to the TTD logs in the controller. Click on the object representing the controller. Select the TTD Node Configuration aspect in the aspect list.

10. Select the Synchronize view and click **Download**. Select **No** in the Match Existing dialog box that appears. Study the results on the screen as the download proceeds:

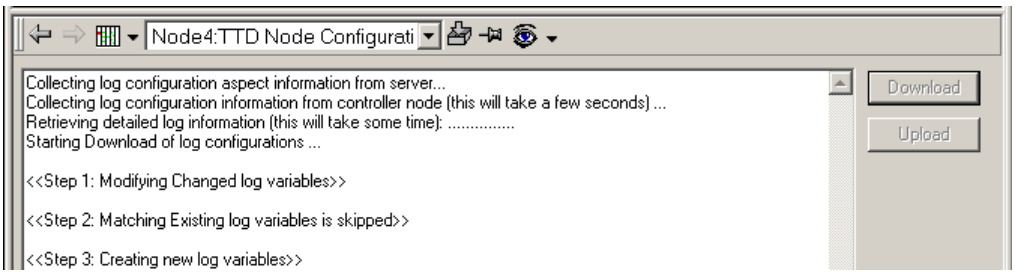


Figure 53. Downloading new TTD Log Definitions

11. If there were no error messages displayed during download, the TTD logs should now be mapped to a Log Group in the controller and started. You can bring up the Main view and study all TTD logs.
For more information, refer to TTD Node Configuration Aspect on page 77.

Direct OPC Logs

Configuration of Direct OPC Logs is described in *System 800xA, Configuration (3BSDS011222*)*.

Please note that if a communication failure occurs between controller and connectivity server, all log points in the failure period will be lost.

Always use 9s sample time for the primary log when a sample time faster than 2 min are wanted. Sample times from 2 min and upwards can be used for a limited number of primary OPC logs without causing load problems in the system.

OPC Direct Logging - 1, 3 and 9 Seconds

Direct OPC logging for 1, 3 and 9 seconds use the same method for data subscriptions as cyclic subscriptions in for example process displays. This type of subscriptions is controlled and optimized in the Connectivity Server, which for example will subscribe data only once if same data is requested by both logs and displays. The AC 400 Series controller is optimized for 1,3, and 9 second subscriptions.

Subscribe only to OPC properties that support cyclic subscriptions.
The properties that support cyclic subscriptions are listed in Appendix E, Cyclic OPC Properties.

The recommended Direct OPC Logging cyclic rate is 9 seconds.

OPC with controller generic cyclic subscriptions is the least costly alternative regarding CPU and memory resources in the controller and the data transfer between the controller and History Server is also optimal. If other clients (such as Process Graphics) subscribe to the same object property as the History Server, only one common subscription is setup in the controller.

OPC Direct Logging - Other Storage Intervals

Using OPC logs without controller generic cyclic subscriptions does not allocate resources permanently in the controller.

Please note that it can cause high overhead in data transfer between the controller and the connectivity server (require ten times more data transfer per log point/sec than 1,3 and 9 seconds).

For more detailed information about other subscriptions see Section 7, Data Transfer via OPC DA.

Section 6 Quick List

The Quick List can be used for searches. Some Quick Lists may be predefined in your system, but you can also configure your own Quick Lists. See *800xA for Advant Master, Operation (3BSE030352*)* for more information regarding Quick List configuration.

Quick List Data Source Aspect

If the Quick List aspect is created in the Control Structure, it is not necessary to create any Quick List Data Source aspect. In case it is created on an object in another structure e.g. the Functional Structure, a Quick List Data aspect need to be added to the same object. In the configuration view of the Quick List Data Source aspect you choose provider, see Figure 54.

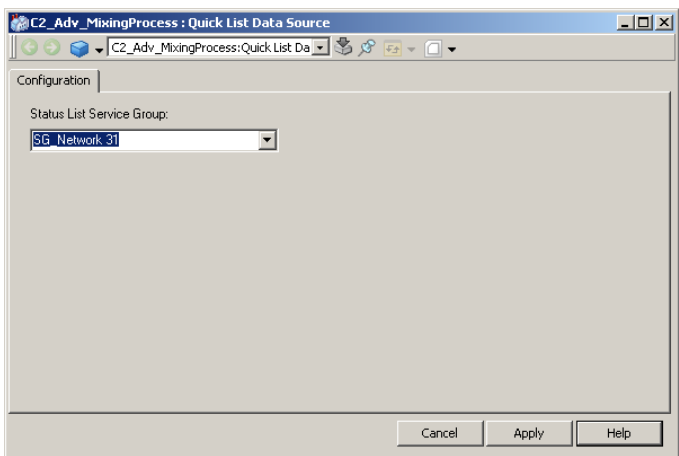


Figure 54. Quick List Data Source Aspect

Section 7 Data Transfer via OPC DA

800xA for Advant Master supports OPC (DA) OPCs. See Appendix D, Control Aspect for detailed information about CCA.

For detailed about the 800xA OPC DA interface see *System 800xA,Configuration (3BDS011222*).*

The OPC client application defines the Update Rate for the OPC Group in milliseconds unit (ms). If Update Rate 0 is specified, the default update rate defined in CCA is used. Negative Update Rate .

To optimize performance of the MB300 RTA Board:

- Subscribe cyclically, and use only 1, 3 and 9 second cyclic rates.
- Only subscribe to items that can be fed cyclically by the controllers.
- Avoid overriding the default UpdateRate in Process Graphics.
- Only write to MB300 DAT objects.

Do not read recent History values too frequent when the log's data source is TTD.

OPC Subscriptions

When configuring OPC subscription of properties from process object, it is important to understand the different types of available subscriptions. The following types exist:

- **Properties available in predefined cyclic subscription**.
  Properties in this group have the default update rate –9000 ms in CCA, and are also found in Appendix E, Cyclic OPC Properties.
  Supported update rates are 1000, 3000, 9000 ms for cyclic updates and –1000, –3000, –9000 ms for cyclic update with additional event driven updates.
- **Properties available in predefined event driven subscription**.
  Properties in this group have the default update rate –9000 ms in CCA, but not found in Appendix E, Cyclic OPC Properties.

Cyclic update is not possible.

- **Properties available in predefined demand update**.
  Properties in this group have the default update rate 9000 ms in CCA.
  Typically, static data such as text properties belong to this group.
  Cyclic subscription less than 10000 ms and Event driven subscription is not possible.
- **Other properties**.
  Properties in this group have the default update rate 20000 ms in CCA.
  The method used to retrieve data is cyclic polling, which is much more resource consuming compared with predefined subscriptions. Event driven subscription is not possible.

For all groups above the following is applicable:

- Update Rates greater than 10000 ms are possible. However, a cyclic polling method is used. This method is not optimized in terms of CPU load in RTA and in the Controller, and should therefore be used restrictive. There are no possibilities to get event driven updates with this method.
- Actual data from the Controller is always notified to the client once the subscription is started.

OPC Read Operations

To minimize load on the system, use 1000, 3000 or 9000 ms Subscriptions and do Synchronous cache read.

OPC Write Operations

It is only supported to do Write Operations towards DAT object marked .

The CPU load in the controller has to be taken in account.

Performance

The most visible ‘bottleneck’ in an 800xA System connected to Advant Master network is the CPU load of the RTA board in the connectivity server.

Consider the following when setting up the OPC Data Access subscriptions from the 3rd party client to minimize load:

- Set up the subscriptions as cyclic subscriptions for 1s, 3s or 9s. Preferably 9s. Only subscribe to OPC properties supporting cyclic subscriptions.
- Max 500 subscriptions are allowed with update rate greater than 10s per connectivity server (single or redundant pair).
- For 1s, 3s and 9s the following applies: The RTA CPU load is independent of how many properties you will request on the same tag, provided the same Update Rate is used.
- Do not load the RTA CPU higher than 50%. Use ‘RTA Board Config’ command ANPER - SYSTEM LOAD analyze to monitor the RTA CPU load. For systems with Connectivity Server redundancy, requiring fail-overs to be free of interference, a maximum RTA load of 40% is recommended. This limit is also recommended when only one of the two redundant Connectivity Servers is running.
- To minimize disturbance when the subscription is started, divide your subscriptions into several OPC groups. Not more than 1000 process objects in each group are recommended. When starting up or stopping the subscriptions, do it group by group in order not to overload the RTA and Controller.
- The average CPU load in the Controller should not exceed 80% after starting up the subscriptions.
- If 3’rd party clients are subscribing for many items, there is a risk for communication jam between the controller and connectivity server when client are started. See Appendix H, Special Configuration for how to set up Data Subscription flow control to minimize problems.

Calculation of RTA CPU load originating from data subscriptions is complicated. Here follows an instruction how to calculate load from 1, 3 and 9s subscriptions. Please note that the total RTA CPU load depends on many other functions, such as Time Tagged Data and Event handling.

- The load is directly related to the number of MB 300 signals (data package sent on MB 300) received per second. A flow of 1 MB 300 signal/second consumes 0.25% RTA CPU load when PU410 is used.
- Data from a process object is sent from the controller even if value is not changed (frozen), according to the chosen update rate (1, 3 or 9s).
- Each MB 300 signal consists of data from 1 up to 38 process objects in the controller. Use table below for maximum number of objects (instances) in each MB 300 signal dependant of object type.
- Each MB 300 signal only contains data from one node and from one object type with the same update rate. As a result of this optimization, data from 23 AI objects from one node utilize the same resources as 1 AI.

Calculation Example

3 s subscription from node A: 100 AI, 10 MOTCON and 5 PIDCONA
9 s subscription from node A: 400 AI
1 s subscription from node B: 1 AI, 25 AO

Average number of MB300 signals each second:
(5 + 2 + 2) / 3 + 18 / 9 + (1 + 2) / 1 = 8

RTA CPU load caused by subscription: 8 * 0.25% = 2.0%

Table 4. Object Type Specific Signal Packing

Object Type	Max Instances in One Signal
AI, AO, FI	23
DI, DO	38
DAT	35
GENOBJ	9
GROUP_ALARM	12
MANSTN	7

Table 4. Object Type Specific Signal Packing

Object Type	Max Instances in One Signal
MOTCON, VALVECON, TEXT_DATA	5
PIDCON, RATIOSTN	4
PIDCONA	3
SEQ	6
GI	14

Section 8 System Administration

System Diagnostics

The AC 400 Series Controllers produce a number of event and alarm messages to inform you about errors concerning the Advant Master system. These Advant Master specific system alarms are found in a System Alarm List aspect on the network object together with the 800xA system alarms. The System Alarms are presented in plain text.

Fault Finding and User Repair

Here are some hints on how to identify the cause of malfunctions within the Advant Master System. It could be difficult to decide whether the problem you see is a problem within the process, or a problem within the Advant System system. There are some tools you can use for your trouble shooting:

- System Status aspects
- System Alarm List aspects
- RTA Board aspects
- Communication aspects

System Status

The System Status presents the status of the control system. It presents the result from the control system internal supervision programs. The System Status is available for **Service Structure** and **Control Structure**.

System Status Viewer

The System Status Viewer shows status of different parts in the system. System status information can be available for both software processes and hardware units. The System Status Viewer aspect can be located on objects on different levels in all structures.

The System Status Viewer shows all objects that provide system status information. All status information in the System Status Viewer is updated dynamically when a change of status occurs.

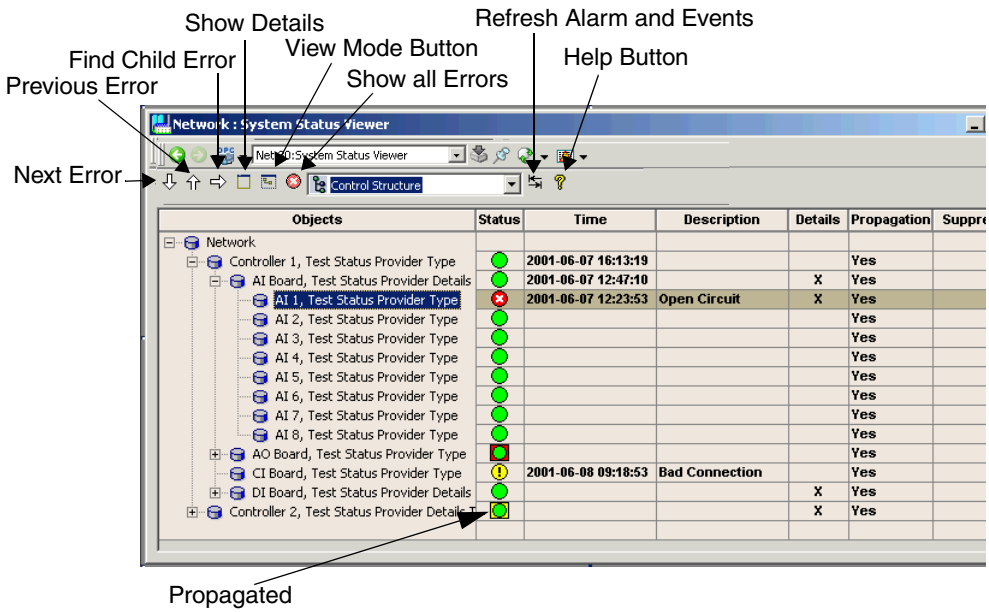


Figure 55. Example of a System Status Viewer on Network Object in the Control Structure

System Status - Service Structure

A System Status Viewer is available on the parent object in Service Structure, where the states of all services are presented. Many service providers also report their detailed status through a Service Status Object, also visible in the viewer. The MasterBus 300 RTA Management Service provider can report errors about the RTA

board, such as PU 410 redundant cable failure. A Service Event List is also available in Service Structure, where system messages about RTA are presented.

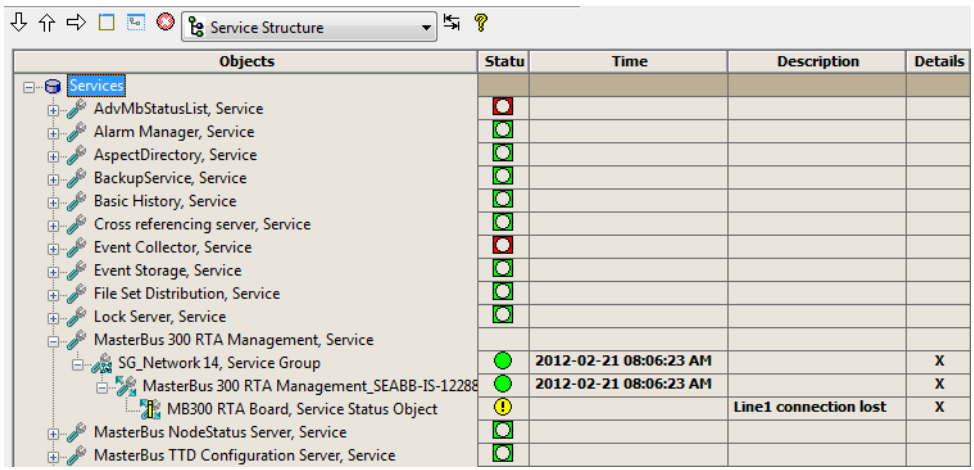


Figure 56. System Status Viewer Service

4		12-02-21 08:12:51:734	MasterBus 300 RTA Management_SEA		PU410 line 2 connected
4		12-02-21 08:11:20:498	MasterBus 300 RTA Management_SEA		PU410 RNRP connected
2		12-02-21 08:11:20:498	MasterBus 300 RTA Management_SEA		PU410 line 2 connection lost
4		12-02-21 08:11:20:498	MasterBus 300 RTA Management_SEA		PU410 line 1 connected
2		12-02-21 08:08:49:648	MasterBus 300 RTA Management_SEA		PU410 RNRP connection lost
2		12-02-21 08:08:49:648	MasterBus 300 RTA Management_SEA		PU410 line 2 connection lost
2		12-02-21 08:08:49:647	MasterBus 300 RTA Management_SEA		PU410 line 1 connection lost
4		12-02-21 08:06:49:397	MasterBus 300 RTA Management_SEA		RTA Board running
1	Inactive	12-02-21 08:06:23:127	SG_Network 14	Inoperative	Service in Operation

Figure 57. Service Event List

System Status - Control Structure

As a start you should have a look at the System Status for the entire Control Structure.

Go to the Control Structure in Plant Explorer, select the root object, and then the System Status Viewer aspect.

If you have several controllers it may cause a delay of the data update.

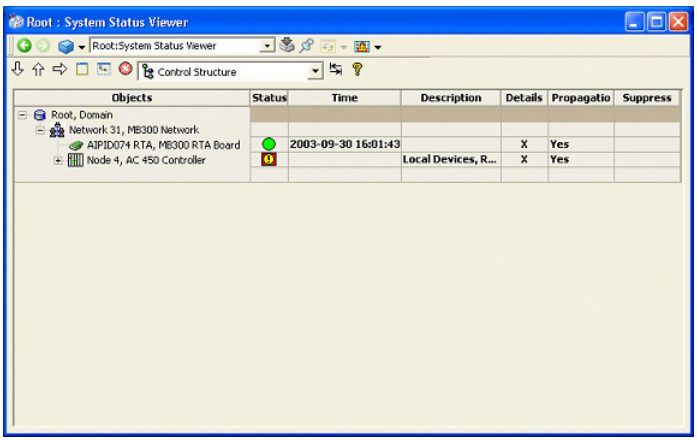


Figure 58. System Status for the Network

In this case you can see that there is something wrong with the AC 450 Controller. The next action should be to study the System Status for this controller.

System Status - MB 300 Control Network

The System Status for a MB 300 control network is presented if you select the Control Structure, the MB 300 Network object, and the System Status Viewer aspect.

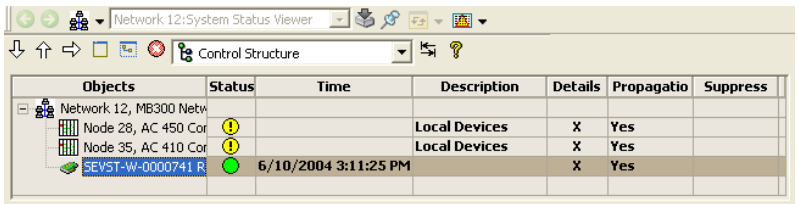


Figure 59. System Status View for MB 300 Network Object

When an RTA Board object is marked with a cross in the **Details** column, double-click in the column to show the Advant Master Connectivity Server Communication Display.

The display is also shown in the Advant Master CS Communication aspect of the MB 300 RTA board object in the Control Structure.

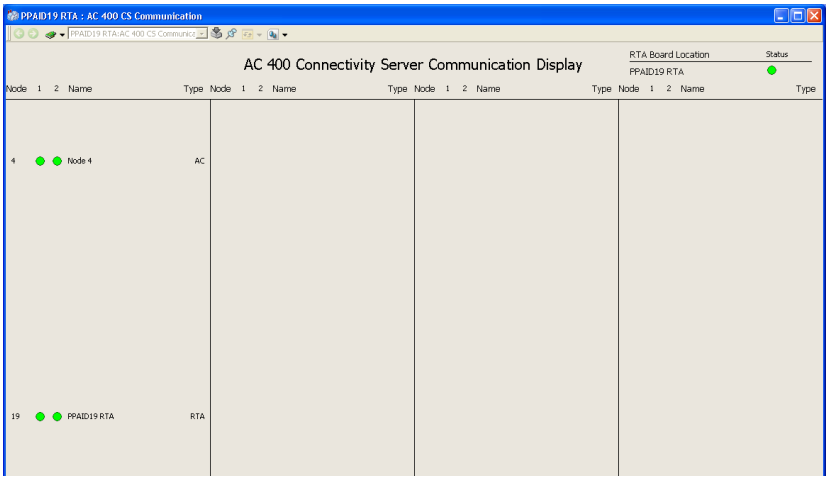


Figure 60. Advant Master Connectivity Server Communication Display

The display lists the communication status for each node the RTA Board can communicate with. The columns are:

- **Node**. It shows the node number.
- **Primary** and **Secondary**. They show the node communication status, and are further described in Table 5.
- **Name**. It shows the node name. The name originates from the node object in the control structure. If the node object is missing, then the object is not defined in the actual network.
- **Type**. It shows the node type, and are further described in Table 6.

For single network only one column is displaying information.

Table 5. Node Connectivity Status

Primary Column	Secondary Column	MB 300 Description
		Communication is OK.
		Communication error for an redundant MB 300 network.
		Communication error for an redundant MB 300 network.
		MB 300 communication error.

Table 6. Node Type

Type Column	Node Description
RTA	AC 400 Connectivity Server
AC	AC 400 Series Controller
MP	MP 200 Controller
OS	AdvaCommand Operator Station
IMS	Information Management System
MG	Master Gate
AC800	AC 800M Controller connected through MB 300 interface CI855
EXT	A generic MB300 node which is external to the 800xA system. For example, a standalone engineering station.

Right-click on any AC node in the communication display to get details from Context menu of its Local Devices object, such as Local Devices and Net Status Graphic displays.

In the top-right corner of the display, the status for the RTA Board Location is shown, see Figure 61.

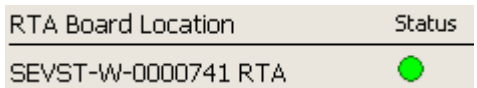


Figure 61. Status Indication

An error indicates that there is no communication with the RTA board, while a warning indicates a problem with the communication. If the status indication shows a warning or an error, restarting the RTA board may solve the problem. To restart the RTA board:

1. Right-click the status indication, and select RTA Board Control.

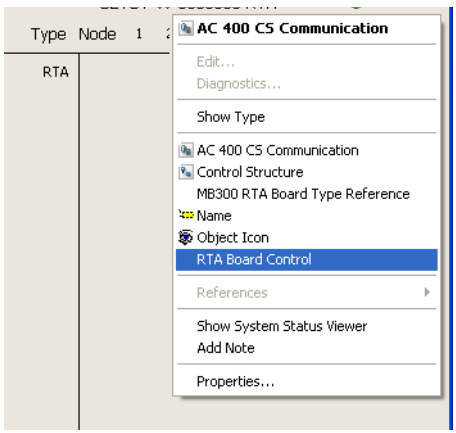


Figure 62. Context Menu

2. In the displayed aspect tab, click **Stop** and wait for the RTA Board to stop (check status).
3. Click on the **Start** button.

System Status - Controller Node

The System Status for a Controller node is presented if you select the Control Structure, the Controller node, and the System Status Viewer aspect. In the example below (Figure 63), the AC 450 Controller (Node 4) has a problem with an Analog input board no. 2.

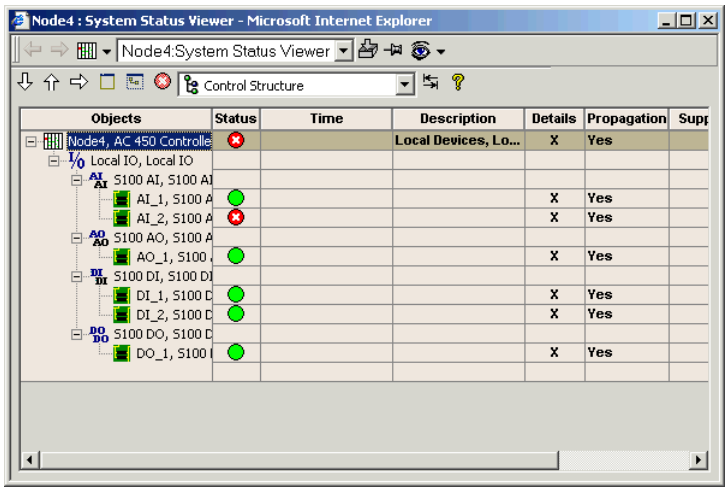


Figure 63. System Status View for Controller node

The next step in your fault finding is to study the system alarm list for the failing Controller, see System Alarms on page 120.

It is possible to archive detailed information about an object in System Status View by a double-click on the cross icon in **Details**. Some objects, such as AC Local Devices and AF100 units, show detailed information in a dedicated graphic display.

When an object is marked with a cross in the **Details** column, double-click in the column, and then select Local Devices, to show the Local Devices graphic display for the controller node.

System Status - Controller Node, Local Devices

This function is not supported for MP 200/1 controllers.

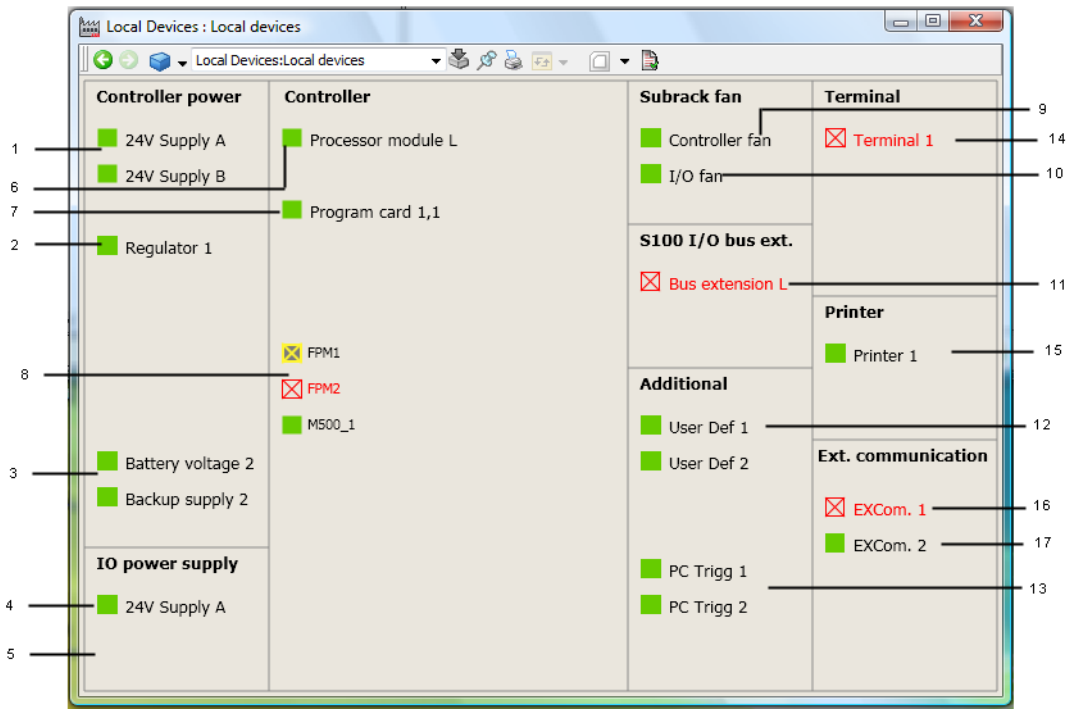


Figure 64. AC 400 Local Devices Display

No	Description	Default Presentation	Condition
1	Controller power supply header		
1.1	24V supply indication	Green filled	OK
		Red crossed	Error
1.2	24 supply name	Grey	OK
		Red	Error
2.1	Regulator indication	Green filled	OK
		Red crossed	Error
2.2	Regulator name	Grey	OK
		Red	Error
3.1	Battery voltage indication	Green filled	OK
		Red crossed	Error
3.2	Battery voltage name	Grey	OK
		Red	Error
3.3	Battery backup supply indication	Green filled	OK
		Red crossed	Error
3.4	Battery backup supply name	Grey	OK
		Red	Error
4	I/O power supply header		
4.1	I/O 24V supply indication	Green filled	OK
		Red crossed	Error

No	Description	Default Presentation	Condition
4.2	I/O 24V supply name	Grey	OK
		Red	Error
5.1	I/O regulator indication	Green filled	OK
		Red crossed	Error
5.2	I/O regulator name	Grey	OK
		Red	Error
6	Controller header		
6.1	Processor module indication	Green filled	OK
		Green empty	Standby
		Yellow crossed	Warning
		Red crossed	Error
6.2	Processor module name	Grey	OK
		Grey	Standby
		Grey	Warning
		Red	Error
7.1	Program card indication	Green filled	OK
		Yellow crossed	Warning
		Red crossed	Error
7.2	Program card name	Grey	OK
		Grey	Warning
		Red	Error

No	Description	Default Presentation	Condition
8.1	Free pgm card & M500 indication	Green filled	OK
		Yellow crossed	Warning
		Red crossed	Error
8.2	Free pgm card & M500 name	Grey	OK
		Grey	Warning
		Red	Error
9	Subrack fan header		
9.1	Fan indication	Green filled	OK
		Red crossed	Error
9.2	Fan name	Grey	OK
		Red	Error
10.1	I/O fan indication	Green filled	OK
		Red crossed	Error
	I/O fan name	Grey	OK
		Red	Error
11	S100 I/O bus extension header		
11.1	S100 I/O bus extension indication	Green filled	OK
		Red crossed	Error
11.2	S100 I/O bus extension name	Grey	OK
		Red	Error
12	Additional header		
12.1	User defined supervision indication	Green filled	OK
		Red crossed	Error

No	Description	Default Presentation	Condition
12.2	User defined supervision name	Grey	OK
		Red	Error
13.1	PC triggered supervision indication	Green filled	OK
		Red crossed	Error
13.2	PC triggered supervision name	Grey	OK
		Red crossed	Error
14	Terminal header		
14.1	Terminal indication	Green filled	OK
		Red crossed	Error
14.2	Terminal name	Grey	OK
		Red	Error
15	Printer header		
15.1	Printer indication	Green filled	OK
		Red crossed	Error
15.2	Printer name	Grey	OK
		Red	Error
16	External communication		
16.1	XCom indication	Green filled	OK
		Red crossed	Error
16.2	XCom name	Grey	OK
		Red	Error

No	Description	Default Presentation	Condition
17.1	RSCom indication	Green filled	OK
		Red crossed	Error
17.2	RSCom	Grey	OK

System Status - Controller Node, Net Status 1

This display shows the network status for the selected controller, that is, the actual communication status to other nodes connected through Master Bus 300, GCom or RCom.

The page shows up to 90 network connections. If required, page 2 shows 91 to 180 connections.

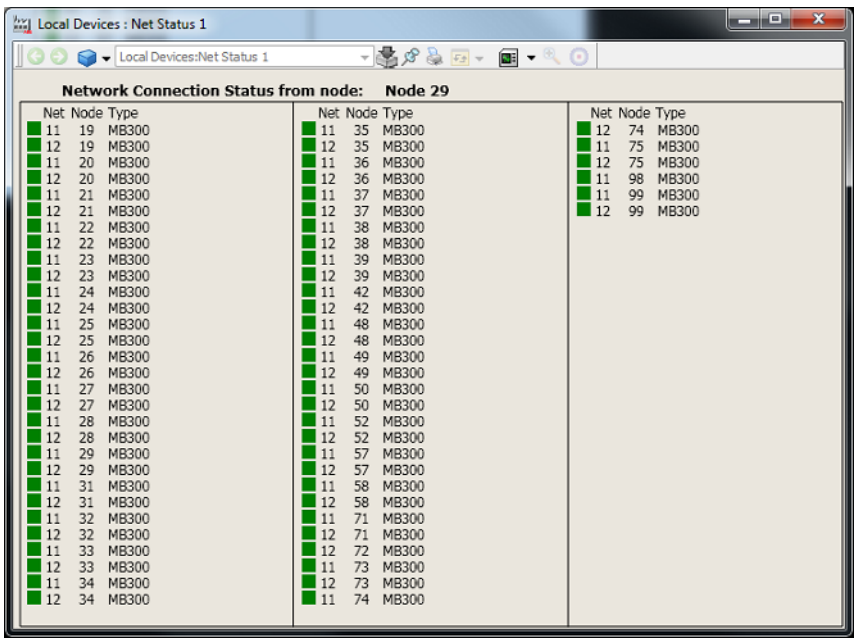


Figure 65. Local Devices Net Status 1

System Status - AF 100 unit detailed status

800xA for Advant Master offers a number of status displays for AF 100 buses, stations and I/O modules. To open the displays, double-click  in the Details column in **System Status Viewer** in the **Control Structure.**

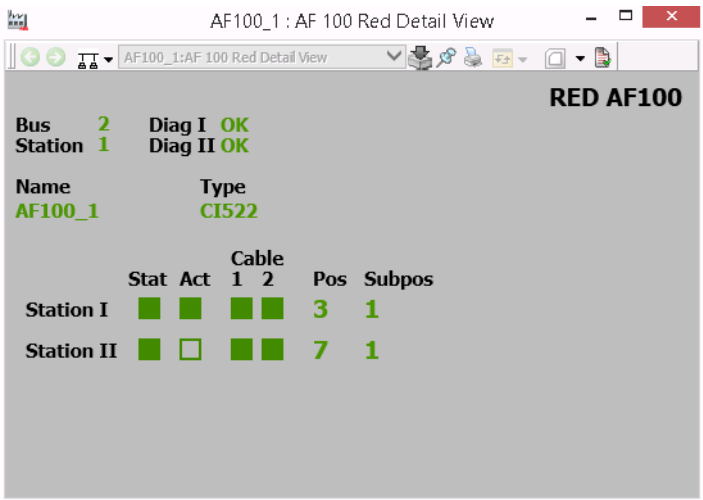


Figure 66. Example of AF 100 Bus Detailed View

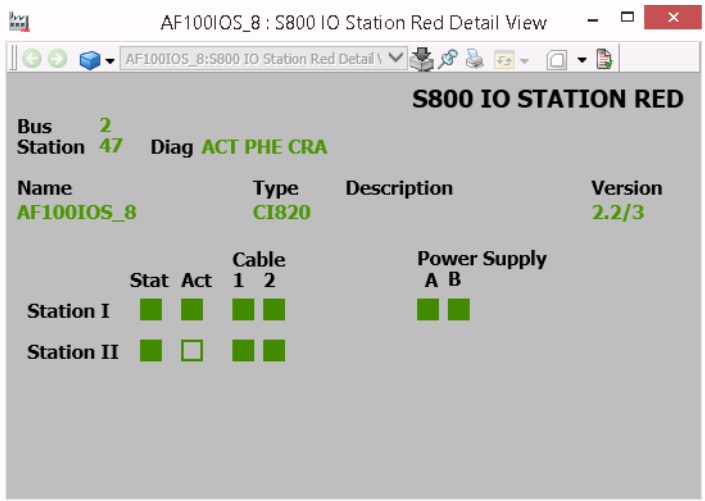


Figure 67. Example of AF 100 Station Detailed View

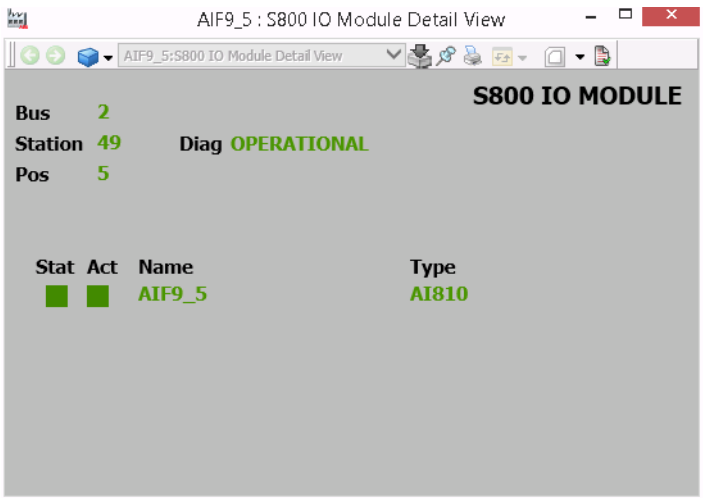


Figure 68. Example of S800 I/O Module Unit Detailed View

System Alarms

System Alarms provide valuable information for fault tracing. By default, you have a System Alarm aspect on the Network object in your Control Structure. You should use it since it will give you an overview list for all system alarms in your system.

Two types of Advant Master System Alarms exist:

- System alarms from MB300 nodes
- System alarms from Services

System Alarms from MB300 nodes

System Alarms from MB300 nodes do not indicate the current state of the alarm if the alarm is active or inactive and are always removed from the alarm list after acknowledged. All these system alarms have the priority level three. An example of the Object Name and Condition for a System Alarm from MB300 nodes is “MN STATUS Conn. lost netw/node 11 52”.

Do not delete objects related to Advant system alarms which may be created in **Control Structure > Lost&Found** if any character conversion other than English is selected in RTA Board Control aspect or if the Advant system alarms are rephrased.

System Alarms from Services

The most important System Alarms from services in 800xA for Advant Master are described in the table below.

- The Message Description column lists short descriptions of system events.
- The Extended Description column explains system events further.

Table 7. System Alarms from Services

Message Description	Extended Description
Failed to create RTA Management Handler	The 800xA for Advant Master data service failed to connect to the RTA board Server. This indicates that no data from the controller(s) will be available through this Connectivity Server Node.
Failed to initialize	Could not start the 800xA for Advant Master data service due to severe problems in the system. This indicates that no data from the controller(s) will be available through this Connectivity Server Node.
No nodes available on MB 300 network	No contact with the controller(s) on the MB 300 network. This indicates that no data from the controller(s) will be available through this Connectivity Server Node.
Failed to send data to client	The 800xA for AC 400 TTD Server has lost contact with a History Server. This indicates that no history data from the controller(s) will be logged from this Connectivity Server Node.

Table 7. System Alarms from Services

Message Description	Extended Description
RTA Board failed (stall) or RTA Board failed (connection lost)	The RTA Management Server has lost contact with the RTA board. This indicates that no data from the controller(s) will be available through this Connectivity Server Node.
RTA Board startup failed	This indicates that no data from the controller(s) will be available through this Connectivity Server Node.
RTA Board startup failed (Network address not set)	This indicates that the Node and network address has not been configured on the RTA Board. No data from the controller(s) will be available through this Connectivity Server Node unless the address is set on the RTA Board.

For more information regarding System Alarms see *System 800xA, Configuration (3BDS011222*)*.

Backup/Restore Procedures

General backup procedures are described in *System 800xA, Maintenance (3BSE046784*)*.

Backup

Backup the following as described below:

- RTA Board Configuration
- Network Configuration Settings
- Time Server Setting

RTA Board Configuration

Changing language settings (character conversion) will change to the default configuration.

You use the RTA Board Control aspect to save changes to the configuration of the RTA board. Use Plant Explorer to locate the Control Structure, the MB300 Network object, the RTA Board object, select the RTA Board Control aspect, and then the **RTA Board Configuration** tab.

This view contains a **RTA Board backup** button. When you have made changes to configuration data on the RTA Board, you shall save the new configuration (if not, the changes will be lost after the next restart of the RTA Board or PC). Click on the **RTA Board backup** button and check the log messages that are presented in the text window. If there are no errors, the new configuration has been saved (and will automatically be loaded to the RTA Board at startup).

The RTA board configuration files are locally stored on the computer with the RTA board. If you want to export the files for backup or for importing to another machine with an RTA board, you can do as follows:

Path: \ProgramData\ABB\AC400Connect\AdvantBase\Data\RTA\Init

In order to view and access the ProgramData folder, in the Windows Explorer, go to **Tools > Folder**. The **Folder Options** dialog appears, select the **View** tab and choose **Show hidden files and folders** option.

Save these three files on a safe location, such as a CD or other server.

- \DATHR1.CD
- \DATHR2.CD
- \DATHR3.CD

Network Configuration Settings

To backup the network configuration settings for an RTA Board:

1. Select the Control Structure, a Server node object, and then an RTA Board object.

2. Select the RTA Board Control aspect, and then the Network/Node Configuration tab.
3. Relevant settings are the values of:
   - Network Number 1
   - Network Number 2
   - Node Number
   - Time Server Setting
4. Write down these settings and store the information on a safe location, such as a CD or other server.

Restore

When you make a restore, you first have to make a general 800xA restore as described in *System 800xA, Maintenance (3BSE046784*).*

RTA Board Configuration

To restore the configuration for an RTA Board, copy the three files you saved during backup from the safe location to the folder\..\ProgramData\ABB\AC400Connect\AdvantBase\Data\RTA\Init. (If prompted, answer Yes to replace existing files).

Before you restore these three files you have to make sure that the system to which you restore to, is running on the same “Character conversion language” as the machine from where you took the dump.

In order to view and access the ProgramData folder, in the Windows Explorer, go to **Tools > Folder.** The **Folder Options** dialog appears, select the **View** tab and choose **Show hidden files and folders** option.

RTA Network Settings

To restore the network configuration settings, setup network and node numbers using the settings you wrote down during the backup. The settings are performed in the RTA Board Control aspect, in the Network/Node Configuration tab. See Figure 5 for an example.

RTA Board Backup

Saves the RTA Board configuration. The saved configuration will be loaded on the RTA Board after the board is started.

The window also contains a progress log (information about the communication with the RTA Board) and the RTA Board status.

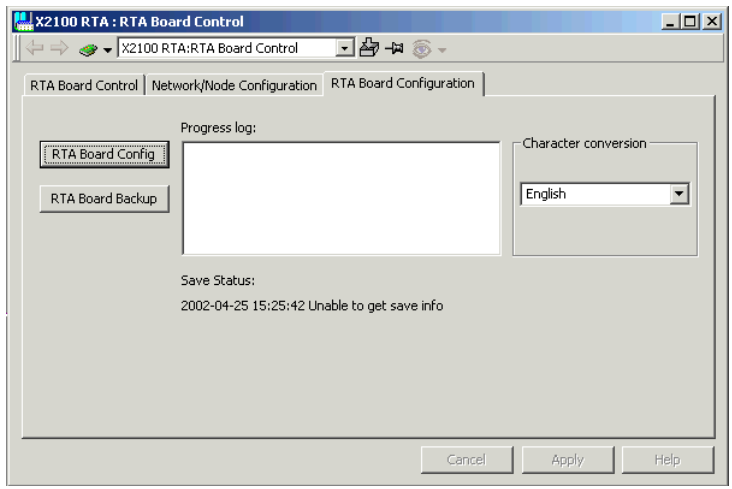


Figure 69. RTA Configuration

Advanced Connectivity Server Configuration

There are two configuration possibilities if it is desired to connect more than 20 controllers to one system, or if it is needed for performance reasons:

- A second Connectivity Server (single or redundant pair) can be connected to support up to 20 additional controllers.
- Split the MB 300 Control Network into two or several networks (maximum of six). The multiple Connectivity Servers function allows several MB 300 Control Networks to be connected from a common 800xA System.

Connect Several MB 300 Networks to an 800xA System

Multiple Connectivity Servers allow you to connect several MB 300 control networks from a common 800xA System.

A Connectivity Server (single or a redundant pair) can only be connected to one MB 300 Control Network. This means that several Connectivity Servers are required if several MB 300 Control Networks shall be connected.

Add Multiple Connectivity Servers to the Same MB 300 Network

A maximum of four pair of connectivity servers can be configured on the same MB300 network, to reduce the load on existing connectivity server(s). The steps specified below describe how to add a second connectivity server pair. The third or fourth pair are added correspondingly based on the described steps.

1. Follow the steps below to configure a second single connectivity server for a MB 300 network:
   a. Add a connectivity server by performing the **Add Connectivity Server** task in the Configuration Wizard.
   b. Select **Add RTA** in the Configuration Wizard. Assign the same network tens as the primary network, but change the last digit to something different, e.g. by adding 2 to the number (enter 13 if primary network number is 11). To avoid confusion, do not use the next consecutive number as this normally indicates network redundancy (11 and 12).
   c. Select **MB300 RTA Settings** in the Configuration Wizard to set network and node numbers for RTA board(s).

Assign the RTA Boards with real network and node numbers, not the network number used in previous step.

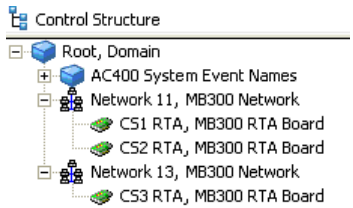


Figure 70. Network Object with Redundant RTA Board Objects in Control Structure

d. Configure the RTA Board event filtering to avoid duplicated alarms and events. See Defining Event Filter on Node Level on page 64. Remember to backup the RTA Board configurations.

e. Add controller objects to the networks. Use the real network and node numbers. Perform an MB 300 Upload for each controller. Optionally, perform a TTD Synchronization (Upload) to create log configurations.

2. Follow the steps to configure redundancy for the second Connectivity Server:

   a. Add a redundant connectivity server via the **Add Redundant Server** in the Configuration Wizard.

   b. Perform procedure b-d in Step1.

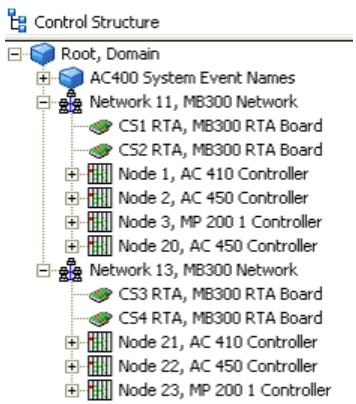


Figure 71. Controller Objects added to the Network Objects in the Control Structure

3. Follow the steps below to move controllers already assigned to the first Connectivity Server, without having problem with history logging and data subscriptions:

   a. Move selected controllers from the first network to the other network by using drag and drop.

   b. Make a dummy modification in the Adapter Data Source Definition aspect of the MB 300 Network object from which the node is moved.

   c. Restart all Basic History Services on both connectivity servers involved in the move.

   d. Restart all services with active subscriptions towards the moved controller node in all servers.

4. Verify the things below:

   – Each network shall have one Service Group for the following services: *AdvMbStatusList*, *Alarm Logger, Basic History*, *Event Collector, MasterBus TTD Configuration Server*, *MasterBus TTD Data Server* and *OpcDA_Connector.*

   – The following aspects of the MB 300 Network objects shall refer to its corresponding Service Group: Adapter Data Source Definition, Quick List Data Source, History Source and TTD Source. For example, the Adapter Data Source Definition of Network 13 shall refer to “SG_Network 13”.

   – Use the System Status Viewer in the Service Structure to verify that all related services have started up and are running without error.

   – Verify that it is possible to subscribe for live data from all controllers.

5. Renaming of Network Objects and Service Groups:
   In order to get a more user friendly naming it is possible to rename Network objects and Service Groups in the Plant Explorer. For example, “Network 13” can be renamed to “Network 11_2” and corresponding Service Groups can be renamed to “SG_Network 11_2” and “Basic 11_2”.

Required Settings to Make an Upload from a Client

Follow the steps below to set up the required settings for making an upload from a client:

1. Run **dcomcnfg** in the Run dialog on the connectivity server.
2. Right-click on **My Computer** and select **Properties**.

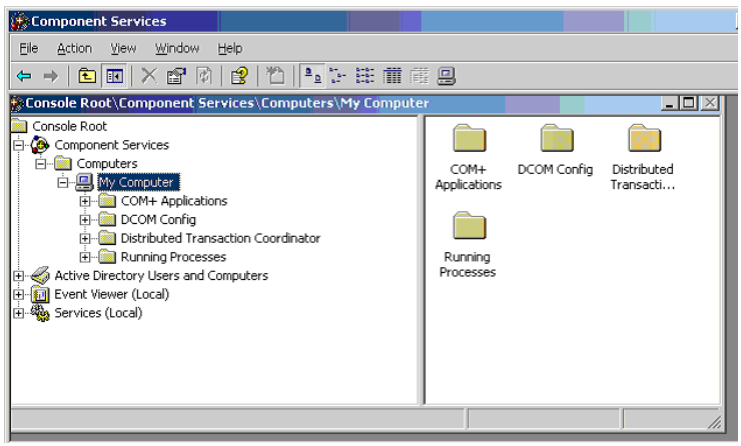


Figure 72. My Computer

3. Select the **COM Security** tab and press the **Edit Limits** button in the **Launch and Activation Permissions** area.

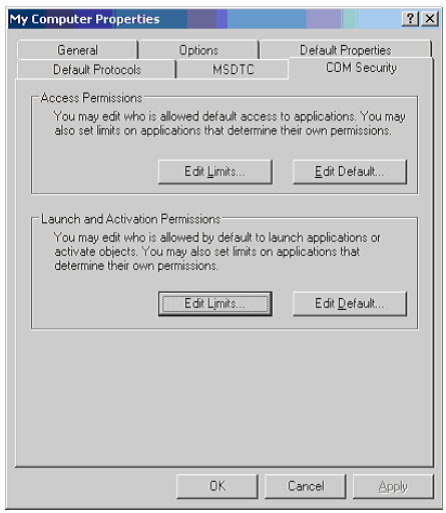


Figure 73. My Computer Properties

4. Add the engineering user, in this case Max Johansson, and give him/her the permissions according to Figure 74.

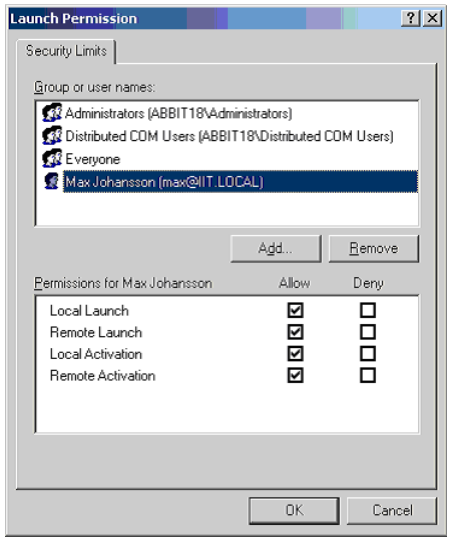


Figure 74. Launch Permissions Dialog Box

Section 9 Advant Master Central Backup

Advant Master Central Backup offers services for Backup and Restore of AMPL Applications of Advant Master Controller nodes (such as Advant Controller 450, Advant Controller 410, Master Piece 200/1, Safeguard 400, Safeguard 3000, and Safeguard 9000) connected to the 800xA system through Master Bus 300.

Central Backup is based on the 800xA framework for Service Backup. The backup data is stored on the same location as the other backup files in the 800xA system, that is, in an Aspect Server or a shared disk connected to it. There may exist several versions of a backup. The format of backup files is the standard On-line Builder command DUAP format.

The user creates **Full Backup Definition** objects in the **Maintenance Structure**. These objects define the scope of backup, that is, the controllers for which the backup must be taken. The Backup can then be invoked manually or scheduled at any time. For more information on the System 800xA backup and restore service, refer to *System 800xA, Maintenance (3BSE046784*)*.

On starting the Backup operation, the **Full Backup** object is created. This object holds the information about the specific backup. Restore can then be invoked from the **Full Backup** object.

Configuration and Operation

All Controller nodes available in the **Control Structure** can be configured for Central Backup. A **Full Backup Definition** can include one or more Controller nodes available in the **Control Structure**, in spite of the Connectivity Servers to which the Controller nodes are connected to. A Controller node can be included in several backup definitions.

The backup function is controlled by the Backup Service Provider, typically located to an Aspect Server. The real backup is performed by a Connectivity Server connected to the Controller. It is not possible to configure which of two redundant Connectivity Servers will perform the backup.

Advant Master Central Backup shares resources with the functions, **RTA Board Configuration**, **RTA Board Backup**, and **MB300 Upload**. The Central Backup can not be started while these functions are running. It is not allowed to run the Backup and Restore operation simultaneously.

If a backup operation is started before the completion of the previous backup, the job will be delayed until the first job is finalized. In such scenarios, it is possible to set several Scheduling Definitions to the same start time.

The backup function can handle several MB300 networks. The restore is always done towards same network or node number. To load the applications to the respective controller, it is recommended to have unique network or node addresses for all nodes in the MB300 networks.

Configuring the Advant Master Central Backup

Execute the following steps to configure the Advant Master Central Backup:

1. Right-click the **Backup Definitions** object in the **Maintenance Structure** and select **New Object** from the context menu.
2. Enter a name for the object of category **Full Backup Definition** and click **Create**.
3. Select the **Backup Definition** aspect of this object.
4. In the tab **Scope**, select the **Backup Type** as **External Services**.
5. In **Configuration**, select the **Advant Master Central Backup** check box.
6. In **Advant Master Central Backup**, select the controller nodes for which the backup must be taken.

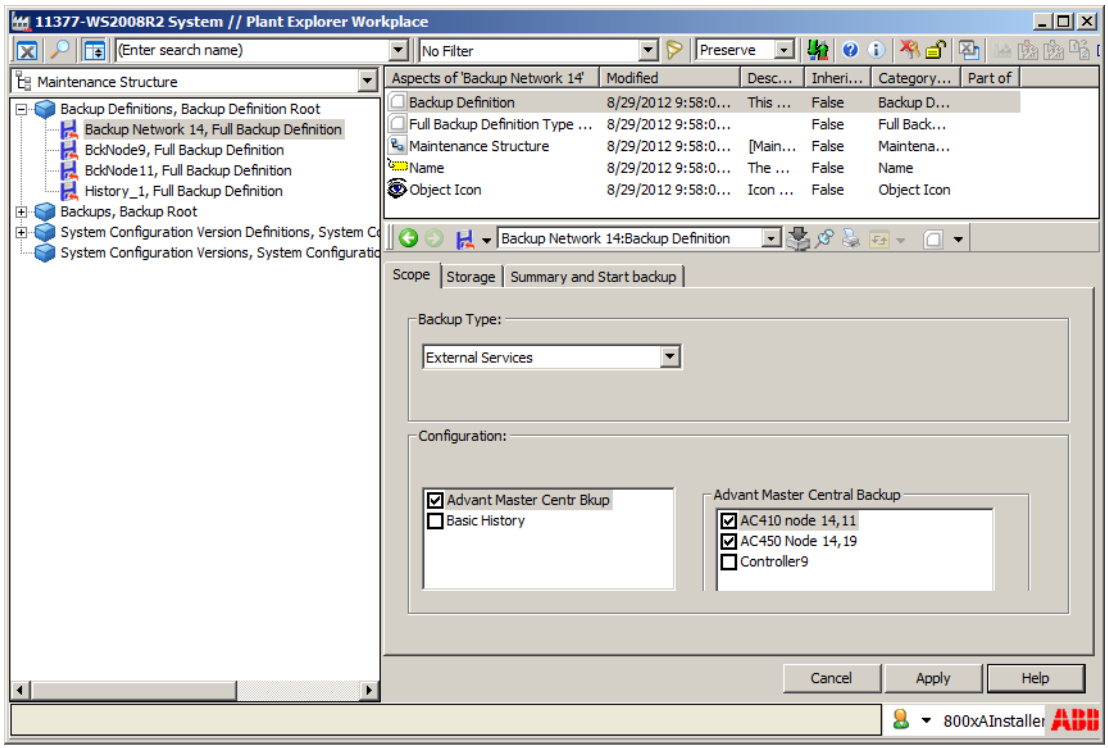


Figure 75. Backup Definition aspect - Scope

7. Click **Apply** to save the changes.

Starting a backup manually

Execute the following steps to start a backup manually:

1. Select the **Backup Definition** aspect of the configured **Full Backup Definition** object from the **Maintenance Structure**.
2. In the tab **Summary and Start backup**, click **Start Backup** to start the backup operation.

When the backup is started, an object <*Backup Definition Name*>; <*Date*>; <*Time*> of the category **Full Backup** is created in the **Maintenance Structure > Backups**.

Backup Info aspect

The **Backup Info** aspect appears in the **Maintenance Structure > Backups > Full Backup** object.

The **Backup Info** aspect contains information about the actual backup.

The following are the tabs included in the **Backup Info** aspect:

- **Status**

  This tab displays the progress of the backup operation. The detailed progress information is available through the **View Log** button.

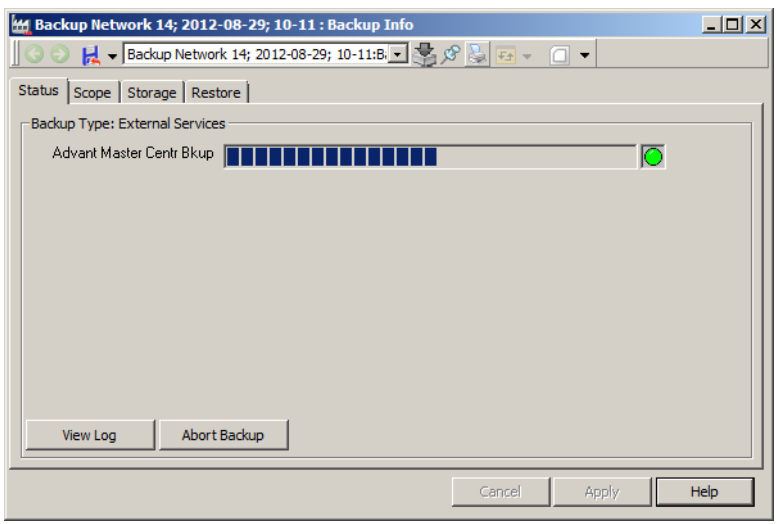


Figure 76. Backup Info aspect-Status tab

- **Scope**

  This tab presents the configured list of nodes for the backup.

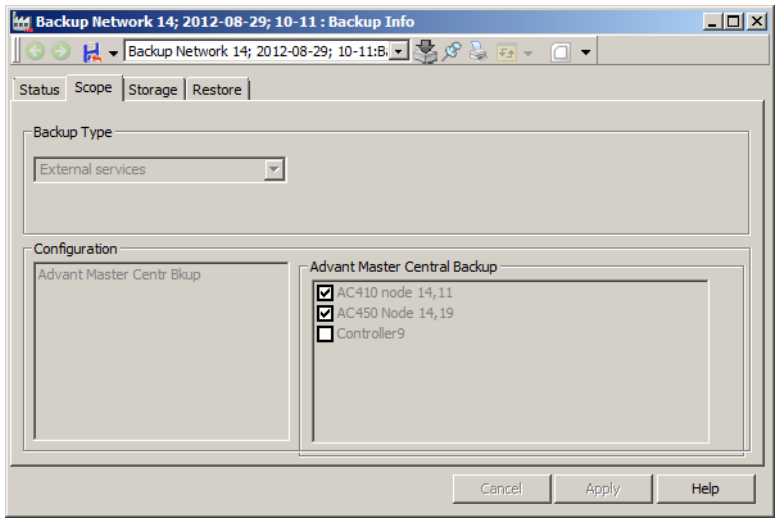


Figure 77. Backup Info aspect-Scope tab

- **Storage**

  This tab presents information about the backup, such as folder location and size.

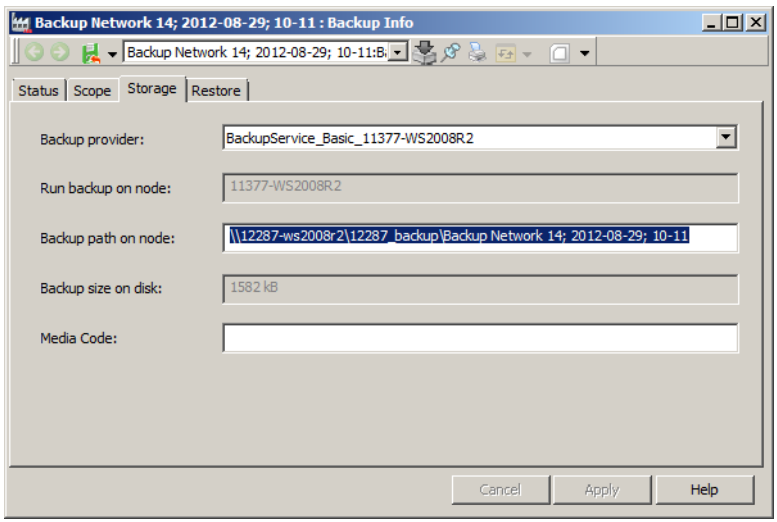


Figure 78. Backup Info aspect-Storage tab

- **Restore**

  This tab displays the content of the backup. The restore operation is started using this tab.

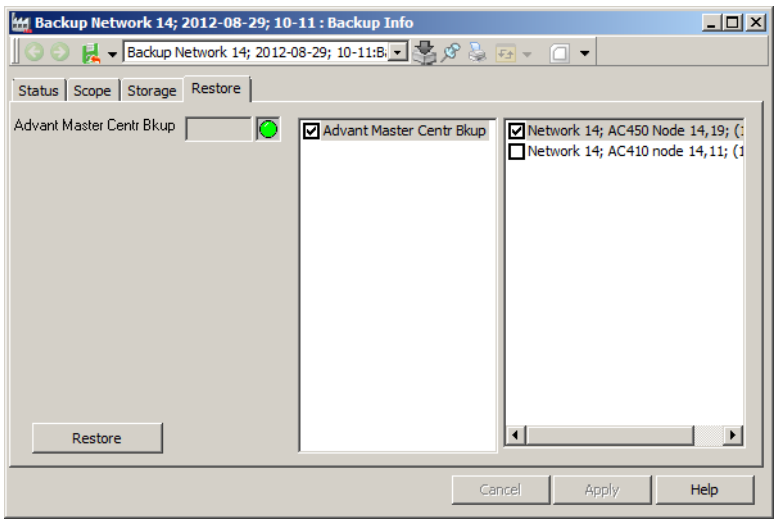


Figure 79. Backup Info aspect-Restore tab

Configuring a scheduled backup

The objects of category **Full Backup Definition** can be scheduled.

To perform the scheduling operation, the system extension **ABB Inform IT Scheduler** that must be installed. For more information on setting up the scheduler, refer to the *Scheduling Reports* section in *System 800xA, Administration and Security (3BSE037410*).*

1. Create an object of the **Job Description** category in **Scheduling Structure > Schedules and Jobs > Job Descriptions**.
2. Select the newly created object and add an **Action** aspect.
3. In the **Action** aspect, select **Backup Action** from the list of actions.

4. Select the object to be scheduled from the list of other **Full Backup Definition** objects.
5. Click **Apply** to confirm the selection.
6. Select the **Scheduling Definition** aspect and set up the schedule parameters.

Restoring a backup

Execute the following to restore a backup:

1. Select **Backup Info** aspect of an object of the **Full Backup** category from the **Maintenance Structure** that requires a restore.
2. In the **Restore** tab, select one or more controller nodes for which the saved AMPL application backup must be restored.
3. Click **Apply** to save the changes. See Figure 79.
4. Before beginning the restore operation, perform a cold start for the controller. The controller must be running in the P2 (configuration) mode.
5. Click **Restore**.
6. When the restore is completed, the controller remains in the P2 mode. To start the controller after restore, start it in the P1 (operational) mode using the On-Line Builder command DICONFIG.

The Safeguard controllers require a manual restore procedure because the same backup shall be used for both the dual nodes. For more information, refer to *Safeguard 400 Series Safety Manual (3BNP000432*)*. Central Backup only supports restore to the same node number. The recommended procedure for Safeguard is to copy the backup files to the engineering station and run the On-line Builder command LOAP with U option (this allows loading even if network or node numbers differ in dump and target).

Maintenance

System events are generated when a Central Backup job is started and finished. It is therefore possible to get the status of backup jobs from the system event list.

Prio	EventTime	ObjectName	Message
4	12-08-29 09:55:39:084	node 11; 2012-08-29; 09-54	Backup node 11; 2012-08-29; 09-54 aborted
4	12-08-29 09:54:10:254	node 11; 2012-08-29; 09-54	Backup node 11; 2012-08-29; 09-54 started

Figure 80. Example of system events from a failed Central Backup job

The detailed information of each backup operation is available in the log file by using the **View Log** button in the **Backup Info** aspect of the **Full Backup** object. See Figure 76.

The log file is stored together with the backup files. The path of this folder is available in the **Storage** tab of the **Backup Info** aspect. Each restore attempt also creates a detailed log file in same folder.

To perform a Restore operation, the network object parent to the related Controller objects in the **Control Structure** must not be deleted or renamed from the time the backup was taken.

Periodic Scheduled Backup jobs functions as desired until the scheduler is not restarted. A restart of the server invokes all Job Descriptions with periodic schedule to take an immediate backup. This is not always desired. If the Connectivity Server is restarted at the same time as the backup server, it is likely that the RTA board is not up and running when the first backup is invoked. This results in a failed backup. A better alternative is to select Weekly Scheduling, which is executed during a specific day and time of the week.

Appendix A Event Treat Elements

Introduction

When an event occurs it is analyzed in the Controller. As a result of that an information package is sent to the Connectivity Server. Using your Operator Workplaces and functionality you can study the alarm and event information in various formats. The presentation of an Alarm or Event generated in an AC 400 Series Controller is made according to the specified event treatment.

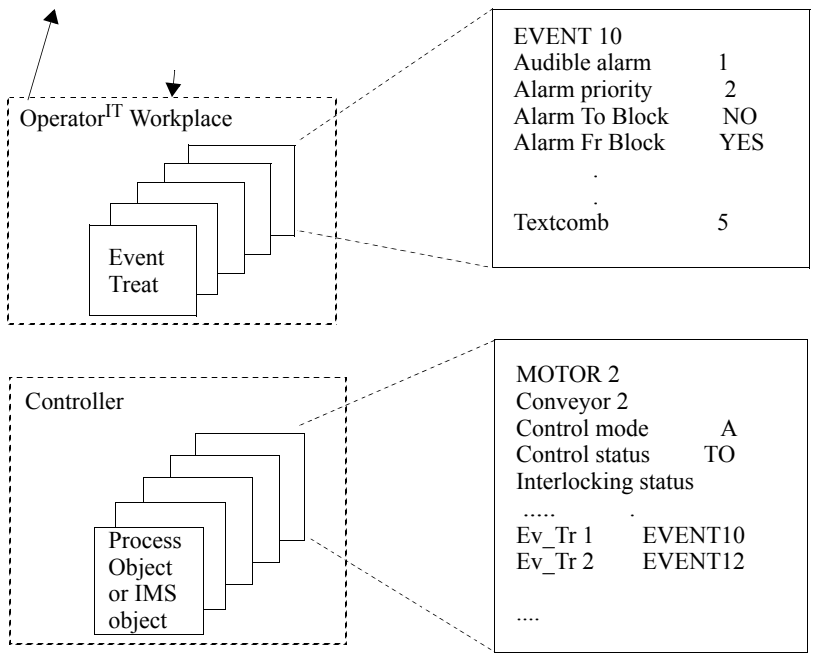


Figure 81. Alarm and event handling

The alarm and event message includes:

- NAME of the object, for example “MOTOR 2”.
- DESCRIPTION of the object, for example “Conveyor 2”.
- A set of status flags for the internal control.
- Pointers to elements (EVENT) in the Event Treat data base, where the event is described.

Each Process and System object contains at least one pointer to the Event Treat data base. A digital input. DI for example, contains two Event Treat pointers, ERR_TR (Error Treat) and VALUE_TR (Value Treat).

ERR_TR is used for internal signal errors when the ERR flag changes status.

VALUE_TR is used for process events and process alarms when the signal value (VALUE) changes status, possibly together with the process error flag (DISTURBANCE).

The Event Treat Database Element

The contents of the Event Treat data base, and ways to vary the event handling, are described below:

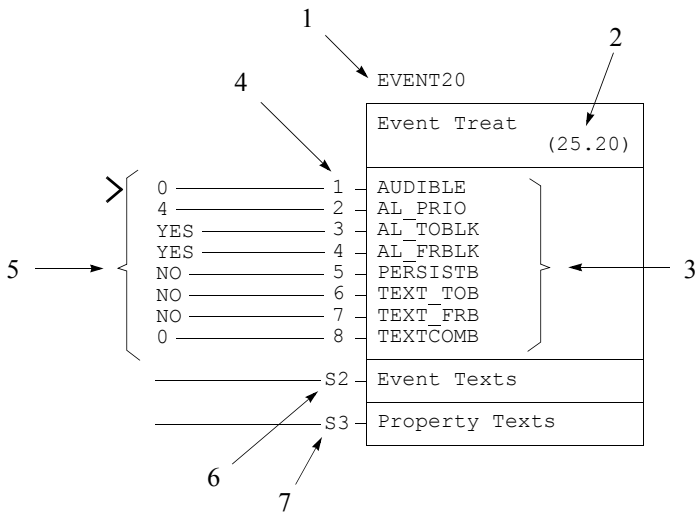


Figure 82. Data Base Element for Event Handling

1. Name and element number in the Event Treat data base.
2. Logic file number and element number.
3. Properties whose values can be changed.
4. Property numbers.
5. Start values applied.
6. Segment S2 containing User defined event texts for use instead of the standard system texts (applies only to EVENT20 - EVENT300).
7. Segment S3 containing User defined property texts for use instead of the standard system texts (applies only to EVENT20 - EVENT300).

The properties in the EVENT element are described below:

AUDIBLE. Must be 0 for events and 1 for alarms. See *System 800xA, Configuration (3BDS011222*)* for configuration of Audible Alarms.

AL_PRIO. Alarm Priority defines the Alarm priority in the presentation. Read the *System 800xA, Operations, Operator Workplace Configuration (3BSE030322*)* and *System 800xA, Configuration (3BDS011222*)* for information about how to configure text color and font, sound effects etc. The priority you define in the Event Treat elements are converted to a corresponding 800xA Priority Level according to the following:

Table 8. Default Mapping Event Treatment Alarm Priority to 800xA Priority Level

Event Treatment Priority	800xA Priority Level
1	1
2	2
3	3
4	4
5	4
6	4
7	4

AL_TOBLK - Alarm To Block. Blocks alarm handling of a supervised alarm signal/flag when it makes a 0 -> 1 transition. Since most alarm signals/flags are active high, AL_TOBLK should be set = N (No) to invoke alarm handling for 0 -> 1 transitions and = Y (Yes) to disable it.

AL_FRBLK - Alarm From Block. Blocks alarm handling of a supervised alarm signal/flag when it makes a 1 -> 0 transition. Since most alarm signals/flags are active high, and since alarm handling on return to normal makes little sense, AL_FRBLK should normally be set = Y (Yes).

PERSISTB - Persistent Alarm Block. Should be set = N (No) when alarm handling is required and = Y (yes) when it is not.

TEXT_TOBLK - Text To Block. Flag that blocks generation of text in lists and printouts when the alarm/event changes from 0 ->1.

TEXT_FRBLK - Text From Block. Flag that blocks generation of text in lists and printouts when the alarm/event changes from 1 -> 0.

TEXTCOMB - Text Combination code. Integer to select if standard or User defined property text and event text shall be used.

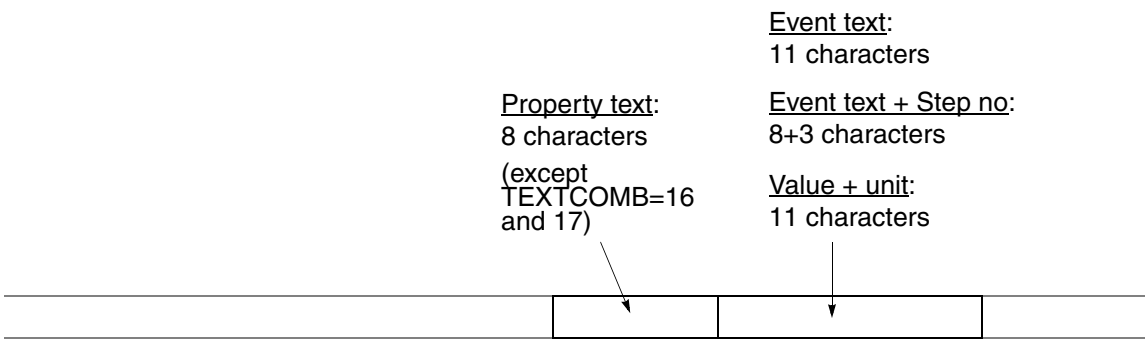


Figure 83. Maximum number of characters for the property and event texts

Table 9. Formats for Property and Event Texts

TEXT COMBination code	Property text	Event text	
0	Standard	Standard	
1		User defined	
2 (1)	Standard		Value + unit
3	Standard	User defined	
4	User defined	Standard	
5	User defined	User defined	
16	User defined (2)	Standard	
17	User defined (2)	User defined	
18 (3)	Standard	Standard	+ Step no
19 (3)	Standard	User defined	+ Step no
20 (3)	User defined	Standard	+ Step no
21 (3)	User defined	User defined	+ Step no
24 (1) (4)	Standard		Value + unit

(1) Suitable for analog values only.
(2) Suitable for long property texts (up to 20 characters).
(3) Specially adapted for the function unit Sequence (SEQ).
(4) Similar to TEXTCOMB=2, but only events caused by a process event will be entered in the list.

Description can NOT be displayed if long Property texts are used (TEXTCOMB 16 and 17).

As mentioned earlier, EVENT20 - EVENT300 can accommodate your own event texts as an alternative to the standard texts in the system. You can enter these by expanding segment S2.

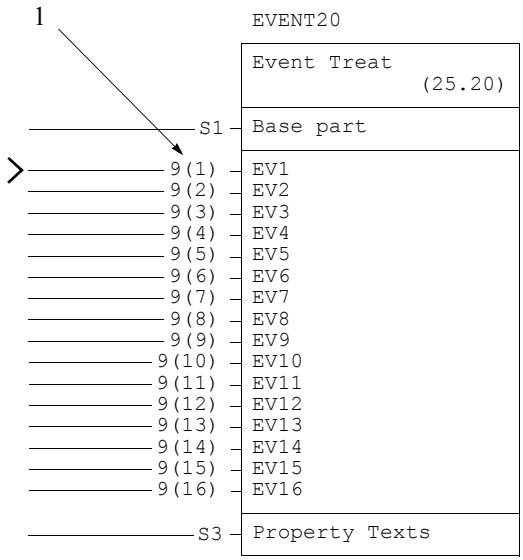


Figure 84. Segment S2 in Event Treat

Line numbers for **16** event texts. Each text can have a **maximum** of **11 characters**. Each event points out a certain line number. It is important when you use a user defined event text that you put the text into the correct line number.

You enter the user defined property texts in EVENT20 - EVENT300 by expanding segment S3.

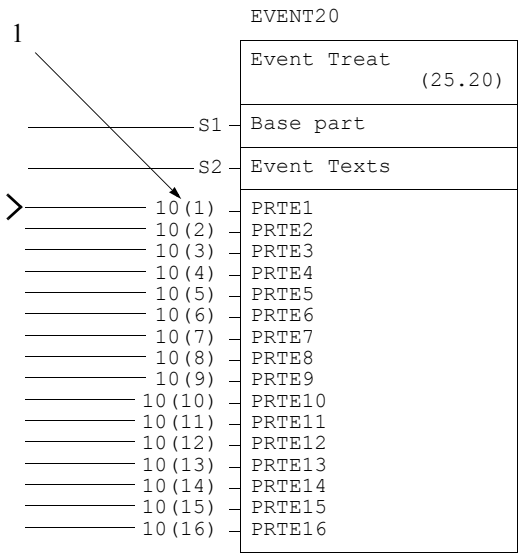


Figure 85. Segment S3 in Event Treat

The relative line numbers for **16** property texts. Each text has a **maximum** length of **20 characters**.

TEXTCOMB decides max number of characters to be presented in the lists.

The coupling between the event and the line number is the same as it is for the event texts. For this reason, you must put the different texts on the correct lines. Appendix C, Event Texts describes the events that generate different line number pointers.

There are **300** EVENT elements available, as described below:

Table 10. Event Elements

EVENT1-18	Predefined for standard event handling (see Table 8). **You are not allowed to change these**
EVENT19	Reserved for future extensions of the standard handling. **You are not allowed to change it.**
EVENT20-94	Available for user defined event handling. You can change all properties.
EVENT95-100	Reserved for Batch applications.
EVENT101-300	Available for user defined event handling. You can change all properties.
EVENT301	Reserved for other applications.

Property	**EVENT**																	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**
AUDIBLE	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**1**	**0**	**0**	**0**	**0**	**1**	**0**	**1**
AL_PRIO	**4**	**2**	**4**	**2**	**4**	**2**	**2**	**2**	**4**	**2**	**2**	**4**	**4**	**4**	**4**	**2**	**3**	**2**
AL_TOBLK	**Y**	**N**	**Y**	**N**	**Y**	**N**	**N**	**N**	**Y**	**N**	**N**	**Y**	**Y**	**Y**	**N**	**N**	**Y**	**N**
AL_FRBLK	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**N**	**Y**	**Y**	**Y**
PERSISTB	**Y**	**N**	**Y**	**N**	**Y**	**N**	**Y**	**N**	**Y**	**N**	**N**	**Y**	**Y**	**Y**	**N**	**N**	**Y**	**N**
TEXT_TOB	**N**	**N**	**N**	**N**	**N**	**N**	**Y**	**N**	**N**	**N**	**N**	**N**	**N**	**N**	**N**	**N**	**N**	**N**
TEXT_FRB	**N**	**N**	**N**	**N**	**N**	**N**	**Y**	**N**	**N**	**N**	**N**	**N**	**N**	**Y**	**N**	**N**	**N**	**N**
TEXTCOMB	**0**	**0**	**24**	**24**	**18**	**18**	**0**	**5**	**5**	**5**	**5**	**5**	**5**	**5**	**0**	**5**	**17**	**17**

Table 11. Recommended Usage of the EVENT Elements

EVENT Number	Usage
EVENT1, EVENT2	Are suitable for binary process signals and objects, for example DI, DO, GENBIN.
EVENT3, EVENT4	Are suitable for analog process signals and objects, for example AI, AO, GENCON and PIDCON.
EVENT5, EVENT6	Are specially adapted for SEQUENCE. EVENT1,EVENT3 and EVENT5 block all alarm handling whereas EVENT2, EVENT4 and EVENT6 permit alarm handling.
EVENT7	Not recommended to use in 800xA.
EVENT8	Is specially adapted for use as I1_TR for GROUP.
EVENT9	Is specially adapted for use as I2_TR for GROUP.
EVENT10	Is specially adapted for use as I1_TR for MOTCON.
EVENT11	Is specially adapted for use as I1_TR for VALVECON.
EVENT12	Is specially adapted for use as I2_TR for MOTCON.
EVENT13	Is specially adapted for use as I2_TR for VALVECON.
EVENT14	Is specially adapted for use for texts generated from the AMPL element EVENT in the Controller.
EVENT15	Not recommended to use in 800xA.
EVENT16	Is specially adapted for use as ADAP_TR for PIDCONA.
EVENT17	Is specially adapted for SPC (Statistical Process Control).
EVENT18	Is specially adapted for SPC (Statistical Process Control).

Appendix B MB300 Uploader

Using the From FCB Tab to Upload

The From FCB Tab in the MB300 Uploader aspect makes it possible to request an upload of object data based on information from the Function Chart Builder Engineering tool.

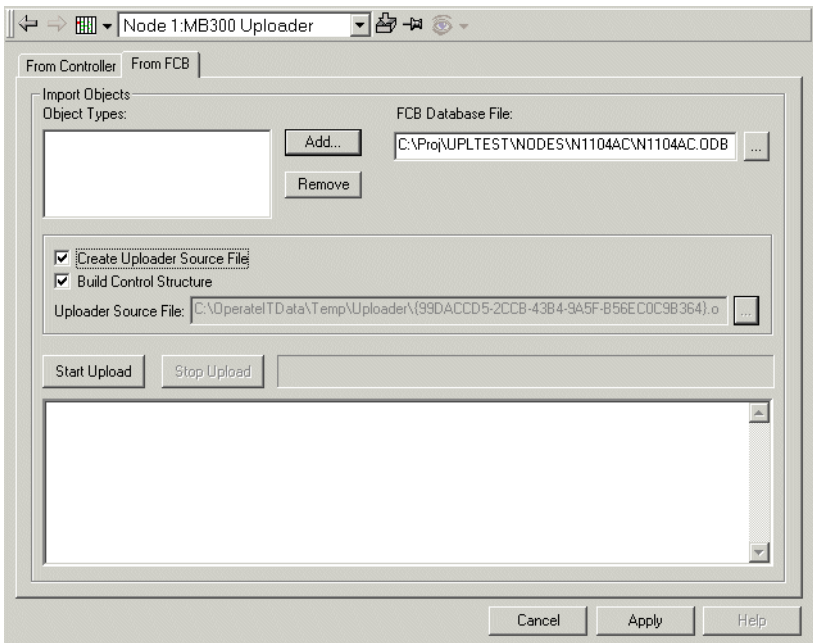


Figure 86. MB 300 Uploader, From FCB

Adding or Removing Object Types to Upload

Click **Add** or **Remove** to modify the list with object types that you want to include in the Upload. If you leave the list blank, all object types are included (this is the default).

The following object types are valid for Upload:

- All object types located in MB 300 Process Objects group.
- Bus and Station object types under Advant Fieldbus 100, Lonbus and Profibus.
- S400 MasterFieldbus object type.

Selecting Type of Upload

You can select three types of upload from this view.

- Check the **Create Uploader Source File** and **Build Control Structure** check boxes to build the Control Structure based on information from the engineering software FCB (Function Chart Builder). You have to fill in the name of the **FCB Database File** (ODB file) to retrieve information from. This choice requires that FCB is installed at the same node.
- Uncheck the **Create Uploader Source File** check box and check **Build Control Structure** check box to build the Control Structure from a previously created **Uploader Source File** (OCD file). You have to enter the name of the Uploader Source File. This choice does not require that FCB is installed at the same node. See Building the Control Structure On-line on page 39.
- Check **Create Uploader Source File** and uncheck **Build Control Structure** to create a Uploader Source file based on the information from the engineering software FCB (Function Chart Builder). You have to fill in the name of the **FCB Database File** (ODB file) to retrieve information from, and the name of the **Uploader Source File** (OCD file) to create. This option does not update or create the Control Structure. Can be used to produce the OCD file for other 800xA systems using the same controllers. This choice requires that FCB is installed at the same node.

Starting the Upload

Click on the **Start Upload** button when you are ready for the upload. The window in the lower part of the aspect view presents messages describing the progress, see Figure 86.

Stopping the Upload

Click on the **Stop Upload** button when you want to interrupt the upload.

Workaround 1

S800 I/O modules are not created correctly from FCB upload which results that the I/O channels are not created in the Control Structure.

To create the S800 I/O modules, perform the following:

1. From FCB tab, check **Create Uploader Source File** and uncheck **Build Control Structure** check box. Click on the **Start Upload** button.
2. Open the **Uploader Source file** (.OCD file) in an editor. The path and file name are found at Uploader Source file in MB300 Uploader aspect.
3. Edit the lines containing second argument "S800_INTF" as described in the example below and save the file. The 8th argument in:

   AI800_1, S800_INTF, -5, 2, 125, 325, -1, 0, 0 should be changed to the used type. The first argument is the name of the DB element where the used type is found.

   Example:

   AI800_1, S800_INTF, -5, 2, 125, 325, -1, AI810, 0, 0, 0

   DO800_2,S800_INTF,-5,2,125,325,1,DO821,0,0,0

   DRISTD_3,S800_INTF,-5,2,125,325, 1,ACS600 STD,0,0,0
4. From FCB tab, uncheck **Create Uploader Source File** and check **Build Control Structure** check box. Click on the **Start Upload** button.

Workaround 2

S400 I/O modules are not created correctly from FCB upload which results that the I/O channels are not created in the Control Structure.

To create the S400 I/O modules, perform the following:

1. From FCB tab, check **Create Uploader Source File** and uncheck **Build Control Structure** check box. Click on the **Start Upload** button.

2. Open the **Uploader Source file** (.OCD file) in an editor. The path and file name are found at Uploader Source file in MB300 Uploader aspect.
3. Edit the lines containing second argument "SIO_INTERFACE" as as described in the example below and save the file. The 4th argument (-1) in:

   SIO_INTERFACE_1, SIO_INTERFACE, 2, -1, 1, 181, 1, 0, 0, 0 should be changed to a value as given below depending on the used type of basic unit and expansion unit. The first argument is the name of the DB element where used types can be found.

   1 DSAX 452

   17 DSDX 452

   18 DSDX 454

   21 DSDI 452

   22 DSDI 454

   1717 DSDX 452 + DSDX 451

   2117 DSDX 452+ DSDI 451

   1721 DSDI 452 + DSDX 451

   2121 DSDI 452 + DSDI 451

   1818 DSDX 454 + DSDX 453

   2218 DSDX 454 + DSDI 453

   1822 DSDI 454 + DSDX 453

   2222 DSDI 454 + DSDI 453
4. From FCB tab, uncheck Create Uploader Source File and check Build Control Structure check box. Click on the Start Upload button.

Appendix C Event Texts

The events defined for the various signal/object types are described in the tables below with the following layout:

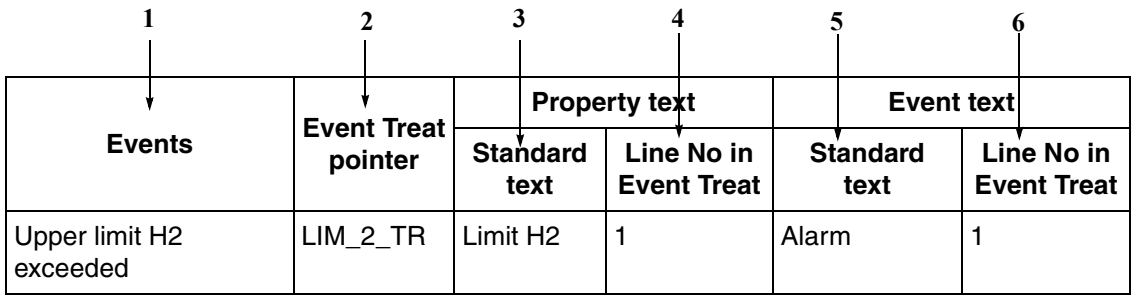


Figure 87. Event Text Table Layout

1. Description of the event.
2. Pointer to data base elements in Event Treat. All pointers for each signal/object type and the recommended values for standard event handling are described in the manuals *Functional Units, Part 1* to *Functional Units, Part 9*.
3. Standard property text that describes the event (applies to EVENT1-EVENT18). EVENT20- EVENT300 have user-defined texts.
4. Line number in the EVENT data base elements where your own text should be placed (applies to EVENT20- EVENT300).
5. Standard event text that describes the state of the event, alarm-normal, blocked-deblocked, etc. If EVENT20-EVENT300 are chosen, you can define the appropriate text yourself.
6. Line number in the EVENT database elements where your own text should be placed (applies to EVENT20- EVENT300). The line number is the same for standard event texts (applies to EVENT1- EVENT18).

Analog Input Signal AI

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Upper limit H2 exceeded	LIM_2_TR	Lim H2	1	Alarm	1
Upper limit H2 re-entered	LIM_2_TR	Lim H2	1	Normal	3
Upper limit H1 exceeded	LIM_1_TR	Lim H1	1	Alarm	1
Upper limit H1 re-entered	LIM_1_TR	Lim H1	1	Normal	3
Lower limit L1 exceeded	LIM_1_TR	Lim L1	2	Alarm	2
Lower limit L1 re-entered	LIM_1_TR	Lim L1	2	Normal	4
Lower limit L2 exceeded	LIM_2_TR	Lim L2	2	Alarm	2
Lower limit L2 re-entered	LIM_2_TR	Lim L2	2	Normal	4
Measured value entered	LIM_1_TR	Value	4	ValueChg	8
H2-value entered	LIM_2_TR	Lim H2	1	ValueChg	5
H1-value entered	LIM_1_TR	Lim H1	1	ValueChg	6
L1-value entered	LIM_1_TR	Lim L1	2	ValueChg	7
L2-value entered	LIM_2_TR	Lim L2	2	ValueChg	6
Signal error On	ERR_TR	Sig.Err	1	Alarm	1
Signal error Off	ERR_TR	Sig.Err	1	Normal	2
Blocking of event handling	ERR_TR	Alarm	2	Blocked	3
Deblocking of event handling	ERR_TR	Alarm	2	Deblocked	4
Blocking of printout	ERR_TR	Printout	3	Blocked	3
Deblocking of printout	ERR_TR	Printout	3	Deblocked	4

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Blocking of DB-update	ERR_TR	Update	4	Blocked	3
Deblocking of DB-update	ERR_TR	Update	4	Deblocked	4

Analog Output Signal AO

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Manual control mode	ERR_TR	Manual	3	On	5
Auto control mode	ERR_TR	Auto	3	On	4
Block output signal against process	ERR_TR	Output sign	6	Blocked	6
Deblock output signal against process	ERR_TR	Output sign	6	Deblocked	7
Measured value entered	ERR_TR	Output sign	2	ValueChg	3
Max limit value entered	ERR_TR	Max limit	7	ValueChg	3
Min limit value entered	ERR_TR	Min limit	8	ValueChg	3

Digital Input Signal DI

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Abnormal position On	VALUE_TR	Value	1	Alarm	1
Normal position On	VALUE_TR	Value	1	Normal	2
Abnormal value entered	VALUE_TR	Value	3	ValueChg	5
Normal value entered	VALUE_TR	Value	3	ValueChg	6
Signal error On	ERR_TR	Sig.Err	1	Alarm	1
Signal error Off	ERR_TR	Sig.Err	1	Normal	2
Blocking of event handling	ERR_TR	Alarm	2	Blocked	3
Deblocking of event handling	ERR_TR	Alarm	2	Deblocked	4
Blocking of printout	ERR_TR	Printout	3	Blocked	3
Deblocking of printout	ERR_TR	Printout	3	Deblocked	4
Blocking of data base update	ERR_TR	Update	4	Blocked	3
Deblocking of data base update	ERR_TR	Update	4	Deblocked	4

Digital Output Signal DO

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Order On	ERR_TR	Output sign	2	On	3
Order Off	ERR_TR	Output sign	2	Off	4
Manual control mode	ERR_TR	Manual	3	On	6
Auto control mode	ERR_TR	Auto	3	Off	5
Blocking of output signal against process	ERR_TR	Output sign	6	Blocked	7
Deblocking of output signal against process	ERR_TR	Output sign	6	Deblocked	8

Sequence SEQ

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Change the step	POS_TREAT	StepChg	1	ValueChg	1
Step number for jump entered	POS_TREAT	Jump mode	2	Enter	2
Number of turns entered	POS_TREAT	Turn no	3	Enter	2
Interval time entered	POS_TREAT	IntervT	4	Enter	2
Indication Active On	IND_TREAT	Active	1	On	1
Indication Active Off	IND_TREAT	Active	1	Off	2
Indication Last Step On	IND_TREAT	Last Step	2	On	3
Indication Last Step Off	IND_TREAT	Last Step	2	Off	4

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Indication Auto control mode On	IND_TREAT	Auto Ind	3	On	5
Indication Manual control mode On	IND_TREAT	Man Ind	4	On	5
Indication step Hold On	IND_TREAT	Hold Ind	5	On	5
Indication Unconditional control mode On	IND_TREAT	Uncond Ind	6	On	5
Indication Next step On	IND_TREAT	Next	7	On	5
Position error On	FAULT_TR	Jump error	1	Alarm	1
Position error Off	FAULT_TR	Jump error	1	Normal	2
Sequence error On	FAULT_TR	SeqAlarm	3	Alarm	1
Sequence error Off	FAULT_TR	SeqAlarm	3	Normal	2
Step error On	FAULT_TR	StepAlarm	4	Alarm	1
Step error Off	FAULT_TR	StepAlarm	4	Normal	2
Blocking of event handling for position change	FAULT_TR	PosAlarm	7	Blocked	3
Deblocking of event handling for position change	FAULT_TR	PosAlarm	7	Deblocked	4
Blocking of printout for position change	FAULT_TR	PosPrint	11	Blocked	3
Deblocking of printout for position change	FAULT_TR	PosPrint	11	Deblocked	4
Blocking of event handling for disturbance	FAULT_TR	Disturb	5	Blocked	3
Deblocking of event handling for disturbance	FAULT_TR	Disturb	5	Deblocked	4

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Blocking of printout for disturbance	FAULT_TR	DistPrint	9	Blocked	3
Deblocking of printout for disturbance	FAULT_TR	DistPrint	9	Deblocked	4
Blocking of event handling for type error	FAULT_TR	TEAlarm	6	Blocked	3
Deblocking of event handling for type error	FAULT_TR	TEAlarm	6	Deblocked	4
Blocking of printout for type error	FAULT_TR	TEPrint	10	Blocked	3
Deblocking of printout for type error	FAULT_TR	TEPrint	10	Deblocked	4
Blocking of event handling for ind. changes	FAULT_TR	IndAlarm	8	Blocked	3
Deblocking of event handling for ind. changes	FAULT_TR	IndAlarm	8	Deblocked	4
Blocking of printout for ind. changes	FAULT_TR	IndPrint	12	Blocked	3
Deblocking of printout for ind. changes	FAULT_TR	IndPrint	12	Deblocked	4
Blocking of operator order	FAULT_TR	OprOrder	13	Blocked	3
Deblocking of operator order	FAULT_TR	OprOrder	13	Deblocked	4

Process Controller PIDCON

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Lower limit L1 for measured value exceeded	H1L1_TR	MV<L1	1	<Value> <Unit>	1
Lower limit L1 for measured value re-entered	H1L1_TR	MV<L1	1	<Value> <Unit>	2
Upper limit H1 for measured value exceeded	H1L1_TR	MV>H1	2	<Value> <Unit>	1
Upper limit H1 for measured value re-entered	H1L1_TR	MV>H1	2	<Value> <Unit>	2
Lower limit (1) for deviation exceeded	H1L1_TR	DEV<L	3	<Value> <Unit>	1
Lower limit (1) for deviation re-entered	H1L1_TR	DEV<L	3	<Value> <Unit>	2
Upper limit (1) for deviation exceeded	H1L1_TR	DEV>H	4	<Value> <Unit>	1
Upper limit (1) for deviation re-entered	H1L1_TR	DEV>H	4	<Value> <Unit>	2
Lower limit L2 for measured value exceeded	H2L2_TR	MV<L2	1	<Value> <Unit>	1
Lower limit L2 for measured value re-entered	H2L2_TR	MV<L2	1	<Value> <Unit>	2
Upper limit H2 for measured value exceeded	H2L2_TR	MV>H2	2	<Value> <Unit>	1
Upper limit H2 for measured value re-entered	H2L2_TR	MV>H2	2	<Value> <Unit>	2

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
PC-gate LOCALFL decides control mode	MODE_TR	LOCAL !!	1	On	1
PC-gate MANFL decides control mode	MODE_TR	MAN !!	2	On	1
PC-gate AUTOFL decides control mode	MODE_TR	AUTO !!	3	On	1
PC-gate E1 decides control mode	MODE_TR	E1 !!	4	On	1
PC-gate E2 decides control mode	MODE_TR	E2 !!	5	On	1
PC-gate E3 decides control mode	MODE_TR	E3 !!	6	On	1
AI-error On	ERR_TR	AISign Err	1	Alarm	1
AI-error Off	ERR_TR	AISign Err	1	Normal	2

Process Controller PIDCONA

Lower limit L1 for measured value exceeded	H1L1_TR	MV<L1	1	<Value> <Unit>	1
Lower limit L1 for measured value re-entered	H1L1_TR	MV<L1	1	<Value> <Unit>	2
Upper limit H1 for measured value exceeded	H1L1_TR	MV>H1	2	<Value> <Unit>	1
Upper limit H1 for measured value re-entered	H1L1_TR	MV>H1	2	<Value> <Unit>	2
Lower limit (1) for deviation exceeded	H1L1_TR	DEV<L	3	<Value> <Unit>	1
Lower limit (1) for deviation re-entered	H1L1_TR	DEV<L	3	<Value> <Unit>	2
Upper limit (1) for deviation exceeded	H1L1_TR	DEV>H	4	<Value> <Unit>	1
Upper limit (1) for deviation re-entered	H1L1_TR	DEV>H	4	<Value> <Unit>	2
Lower limit L2 for measured value exceeded	H2L2_TR	MV<L2	1	<Value> <Unit>	1
Lower limit L2 for measured value re-entered	H2L2_TR	MV<L2	1	<Value> <Unit>	2
Upper limit H2 for measured value exceeded	H2L2_TR	MV>H2	2	<Value> <Unit>	1
Upper limit H2 for measured value re-entered	H2L2_TR	MV>H2	2	<Value> <Unit>	2
PC-gate LOCALFL decides control mode	MODE_TR	LOCAL !!	1	On	1

PC-gate MANFL decides control mode	MODE_TR	MAN !!	2	On	1
PC-gate AUTOFL decides control mode	MODE_TR	AUTO !!	3	On	1
PC-gate E1 decides control mode	MODE_TR	E1 !!	4	On	1
PC-gate E2 decides control mode	MODE_TR	E2 !!	5	On	1
PC-gate E3 decides control mode	MODE_TR	E3 !!	6	On	1
AI-error On	ERR_TR	AISign Err	1	Alarm	1
AI-error Off	ERR_TR	AISign Err	1	Normal	2
Autotuning aborted	ADAP_TR	AT Abort	1	Alarm	1
Autotuning aborted	ADAP_TR	AT Abort	1	Normal	2
Autotuning failed	ADAP_TR	AT Fail	2	Alarm	1
Autotuning failed	ADAP_TR	AT Fail	2	Normal	2
Autotuning. Question to operator	ADAP_TR	AT Quest	3	Decision	3
Autotuning. Question to operator answered	ADAP_TR	AT Quest	3	Answered	4
There is an invalid sample rate	ADAP_TR	TS Fault	4	Alarm	1
There is an invalid sample rate	ADAP_TR	TS Fault	4	Normal	2
Adaptation failed	ADAP_TR	Ad Fail	5	Alarm	1
Adaptation failed	ADAP_TR	Ad Fail	5	Normal	2

Manual Station MANSTN

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Lower limit L1 for measured value exceeded	H1L1_TR	MV<L1	1	<Value> <Unit>	1
Lower limit L1 for measured value re-entered	H1L1_TR	MV<L1	1	<Value> <Unit>	2
Upper limit H1 for measured value exceeded	H1L1_TR	MV>H1	2	<Value> <Unit>	1
Upper limit H1 for measured value re-entered	H1L1_TR	MV>H1	2	<Value> <Unit>	2
Lower limit L2 for measured value exceeded	H2L2_TR	MV<L2	1	<Value> <Unit>	1
Lower limit L2 for measured value re-entered	H2L2_TR	MV<L2	1	<Value> <Unit>	2
Upper limit H2 for measured value exceeded	H2L2_TR	MV>H2	2	<Value> <Unit>	1
Upper limit H2 for measured value re-entered	H2L2_TR	MV>H2	2	<Value> <Unit>	2
AI-error On	ERR_TR	AISign Err	1	Alarm	1
AI-error Off	ERR_TR	AISign Err	1	Normal	2

Ratio Station RATIOSTN

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Lower limit L1 for measured value exceeded	H1L1_TR	MV<L1	1	<Value> <Unit>	1
Lower limit L1 for measured value re-entered	H1L1_TR	MV<L1	1	<Value> <Unit>	2
Upper limit H1 for measured value exceeded	H1L1_TR	MV>H1	2	<Value> <Unit>	1
Upper limit H1 for measured value re-entered	H1L1_TR	MV>H1	2	<Value> <Unit>	2
Lower limit L2 for measured value exceeded	H2L2_TR	MV<L2	1	<Value> <Unit>	1
Lower limit L2 for measured value re-entered	H2L2_TR	MV<L2	1	<Value> <Unit>	2
Upper limit H2 for measured value exceeded	H2L2_TR	MV>H2	2	<Value> <Unit>	1
Upper limit H2 for measured value re-entered	H2L2_TR	MV>H2	2	<Value> <Unit>	2
AI-error On	ERR_TR	AISign Err	1	Alarm	1
AI-error Off	ERR_TR	AISign Err	1	Normal	2

User Defined Controller GENCON

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
AI-error On	AL_TR	AISign Err	1	Alarm	1
AI-error Off	AL_TR	AISign Err	1	Normal	2
Limit for control deviation exceeded	AL_TR	Big Dev	4	Alarm	1
Limit for control deviation re-entered	AL_TR	Big Dev	4	Normal	2
Upper limit H2 for measured value exceeded	AL_TR	MV>H2	5	Alarm	1
Upper limit H2 for measured value re-entered	AL_TR	MV>H2	5	Normal	2
Upper limit H1 for measured value exceeded	AL_TR	MV>H1	6	Alarm	1
Upper limit H1 for measured value re-entered	AL_TR	MV>H1	6	Normal	2
Lower limit L1 for measured value exceeded	AL_TR	MV<L1	7	Alarm	1
Lower limit L1 for measured value re-entered	AL_TR	MV<L1	7	Normal	2
Lower limit L2 for measured value exceeded	AL_TR	MV<L2	8	Alarm	1
Lower limit L2 for measured value re-entered	AL_TR	MV<L2	8	Normal	2
Max limit for output sign On	LIM_TR	SpMaxLim	1	On	1
Max limit for output sign Off	LIM_TR	SpMaxLim	1	Off	2
Min limit for output sign On	LIM_TR	SpMinLim	2	On	1

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Min limit for output sign Off	LIM_TR	SpMinLim	2	Off	2
Max limit for output sign On	LIM_TR	OpMaxLim	3	On	1
Max limit for output sign Off	LIM_TR	OpMaxLim	3	Off	2
Min limit for output sign On	LIM_TR	OpMinLim	4	On	1
Min limit for output sign Off	LIM_TR	OpMinLim	4	Off	2
Op. mode Balance On	IND_TR	Balance	1	On	1
Op. mode Balance Off	IND_TR	Balance	1	Off	2
Op. mode Manual Forced On	IND_TR	ManFrced	2	On	1
Op. mode Manual Forced Off	IND_TR	ManFrced	2	Off	2
Op. mode Manual On	IND_TR	Manual	3	On	1
Op. mode Manual Off	IND_TR	Manual	3	Off	2
Op. mode Auto On	IND_TR	Auto	4	On	1
Op. mode Auto Off	IND_TR	Auto	4	Off	2
Op. mode E1 On	IND_TR	E1	5	On	1
Op. mode E1 Off	IND_TR	E1	5	Off	2
Op. mode E2 On	IND_TR	E2	6	On	1
Op. mode E2 Off	IND_TR	E2	6	Off	2
Controller output On	IND_TR	Controller	9	On	1
Controller output Off	IND_TR	Controller	9	Off	2
Controller On Binary control	IND_TR	BinCtrl	10	On	1
Controller Off Binary control	IND_TR	BinCtrl	10	Off	2
Blocking of alarm	BLK_TR	Alarm	1	Blocked	1

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Deblocking of alarm	BLK_TR	Alarm	1	Deblocked	2
Blocking of event	BLK_TR	Event	2	Blocked	1
Deblocking of event	BLK_TR	Event	2	Deblocked	2
Blocking of printout	BLK_TR	Printout	3	Blocked	1
Deblocking of printout	BLK_TR	Printout	3	Deblocked	2

Binary Object GENBIN

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Signal error On	AL_TR	SignErr	1	Alarm	1
Signal error Off	AL_TR	SignErr	1	Normal	2
Feedback error On	AL_TR	FBErr	2	Alarm	1
Feedback error Off	AL_TR	FBErr	2	Normal	2
Upper limit H2 for measured value exceeded	AL_TR	MV>H2	5	Alarm	1
Upper limit H2 for measured value re-entered	AL_TR	MV>H2	5	Normal	2
Upper limit H1 for measured value exceeded	AL_TR	MV>H1	6	Alarm	1
Upper limit H1 for measured re-entered	AL_TR	MV>H1	6	Normal	2
Lower limit L1 for measured value exceeded	AL_TR	MV<L1	7	Alarm	1
Lower limit L1 for measured value re-entered	AL_TR	MV<L1	7	Normal	2
Lower limit L2 for measured value exceeded	AL_TR	MV<L2	8	Alarm	1
Lower limit L2 for measured value re-entered	AL_TR	MV<L2	8	Normal	2
Op. mode Interlock INTMODE On	INTL_TR	ModeIntl	1	On	1
Op. mode Interlock INTMODE Off	INTL_TR	ModeIntl	1	Off	2

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Process Interlock INTPROC On	INTL_TR	ProcIntl	2	On	1
Process Interlock INTPROC Off	INTL_TR	ProcIntl	2	Off	2
Switchgear Interlock INTSWGR On	INTL_TR	SwgrIntl	3	On	1
Switchgear Interlock INTSWGR Off	INTL_TR	SwgrIntl	3	Off	2
Emergency Stop Interlock INTEMTP On	INTL_TR	EmTrpInt	4	On	1
Emergency Stop Interlock INTEMTP Off	INTL_TR	EmTrpInt	4	Off	2
Op. mode Central On	IND_TR	Central	1	On	1
Op. mode Central Off	IND_TR	Central	1	Off	2
Op. mode Local On	IND_TR	Local	2	On	1
Op. mode Local Off	IND_TR	Local	2	Off	2
Op. mode Remote On	IND_TR	Remote	3	On	1
Op. mode Remote Off	IND_TR	Remote	3	Off	2
Op. mode Manual On	IND_TR	Manual	5	On	1
Op. mode Manual Off	IND_TR	Manual	5	Off	2
Op. mode Auto On	IND_TR	Auto	4	On	1
Op. mode Auto Off	IND_TR	Auto	4	Off	2
Op. mode Blocked On	IND_TR	CtrlBlk	6	On	1
Op. mode Blocked Off	IND_TR	CtrlBlk	6	Off	2
Op. mode Standby On	IND_TR	Standby	7	On	1

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Op. mode Standby Off	IND_TR	Standby	7	Off	2
ON-indication On	IND_TR	On/Open	9	On	1
ON-indication Off	IND_TR	On/Open	9	Off	2
Indication changing On	IND_TR	PosChnge	10	On	1
Indication changing Off	IND_TR	PosChnge	10	Off	2
Blocking of alarm	BLK_TR	Alarm	1	Blocked	1
Deblocking of alarm	BLK_TR	Alarm	1	Deblocked	2
Blocking of event	BLK_TR	Event	2	Blocked	1
Deblocking of event	BLK_TR	Event	2	Deblocked	2
Blocking of printout	BLK_TR	Printout	3	Blocked	1
Deblocking of printout	BLK_TR	Printout	3	Deblocked	2

User Defined Object GENUSD

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Alarm ALQ1 On	AL_TR	Quality 1	1	Alarm	1
Alarm ALQ1 Off	AL_TR	Quality 1	1	Normal	2
Alarm ALQ2 On	AL_TR	Quality 2	2	Alarm	1
Alarm ALQ2 Off	AL_TR	Quality 2	2	Normal	2
Alarm ALF1 On	AL_TR	Fault 1	3	Alarm	1
Alarm ALF1 Off	AL_TR	Fault 1	3	Normal	2
Alarm ALF2 On	AL_TR	Fault 2	4	Alarm	1
Alarm ALF2 Off	AL_TR	Fault 2	4	Normal	2
Alarm ALF3 On	AL_TR	Fault 3	5	Alarm	1
Alarm ALF3 Off	AL_TR	Fault 3	5	Normal	2
Alarm ALF4 On	AL_TR	Fault 4	6	Alarm	1
Alarm ALF4 Off	AL_TR	Fault 4	6	Normal	2
Alarm ALF5 On	AL_TR	Fault 5	7	Alarm	1
Alarm ALF5 Off	AL_TR	Fault 5	7	Normal	2
Alarm ALF6 On	AL_TR	Fault 6	8	Alarm	1
Alarm ALF6 Off	AL_TR	Fault 6	8	Normal	2
Interlock INTLU1 On	INTL_TR	Interl 1	1	On	1
Interlock INTLU1 Off	INTL_TR	Interl 1	1	Off	2
Interlock INTLU2 On	INTL_TR	Interl 2	2	On	1
Interlock INTLU2 Off	INTL_TR	Interl 2	2	Off	2
Interlock INTLU3 On	INTL_TR	Interl 3	3	On	1

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Interlock INTLU3 Off	INTL_TR	Interl 3	3	Off	2
Interlock INTLU4 On	INTL_TR	Interl 4	4	On	1
Interlock INTLU4 Off	INTL_TR	Interl 4	4	Off	2
Interlock INTLU5 On	INTL_TR	Interl 5	5	On	1
Interlock INTLU5 Off	INTL_TR	Interl 5	5	Off	2
Interlock INTLU6 On	INTL_TR	Interl 6	6	On	1
Interlock INTLU6 Off	INTL_TR	Interl 6	6	Off	2
Op. mode M1 M1 On	IND_TR	OperPos1	1	On	1
Op. mode M1 M1 Off	IND_TR	OperPos1	1	Off	2
Op. mode M2 M2 On	IND_TR	OperPos2	2	On	1
Op. mode M2 M2 Off	IND_TR	OperPos2	2	Off	2
Op. mode M3 M3 On	IND_TR	OperPos3	3	On	1
Op. mode M3 M3 Off	IND_TR	OperPos3	3	Off	2
Op. mode Manual On MAN	IND_TR	Manual	5	On	1
Op. mode Manual Off MAN	IND_TR	Manual	5	Off	2
Op. mode Auto On AUTO	IND_TR	Auto	4	On	1
Op. mode Auto Off AUTO	IND_TR	Auto	4	Off	2
Op. mode Blocked On BLK	IND_TR	CtrlBlk	6	On	1
Op. mode Blocked Off BLK	IND_TR	CtrlBlk	6	Off	2
Op. mode Standby On STBY	IND_TR	Standby	7	On	1
Op. mode Standby Off STBY	IND_TR	Standby	7	Off	2
Indication IND1 On	IND_TR	Indic.1	9	On	1

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Indication IND1 Off	IND_TR	Indic.1	9	Off	2
Indication IND2 On	IND_TR	Indic.2	10	On	1
Indication IND2 Off	IND_TR	Indic.2	10	Off	2
Indication IND3 On	IND_TR	Indic.3	11	On	1
Indication IND3 Off	IND_TR	Indic.3	11	Off	2
Indication IND4 On	IND_TR	Indic.4	12	On	1
Indication IND4 Off	IND_TR	Indic.4	12	Off	2
Blocking of alarm	BLK_TR	Alarm	1	Blocked	1
Deblocking of alarm	BLK_TR	Alarm	1	Deblocked	2
Blocking of event	BLK_TR	Event	2	Blocked	1
Deblocking of event	BLK_TR	Event	2	Deblocked	2
Blocking of printout	BLK_TR	Printout	3	Blocked	1
Deblocking of printout	BLK_TR	Printout	3	Deblocked	2

Motor Control MOTCON

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Control voltage absent	I1_TR	ControlV	1	Alarm	1
Control voltage restored	I1_TR	ControlV	1	Normal	2
Bimetal relay has tripped	I1_TR	BimetalR	2	Alarm	1
Bimetal relay triggered	I1_TR	BimetalR	2	Normal	2
Local Stop	I1_TR	LStop	3	Alarm	1
Local Stop ended	I1_TR	LStop	3	Normal	2
Safety Monitor tripped	I1_TR	SafeMon	4	Alarm	1
Safety Monitor triggered	I1_TR	SafeMon	4	Normal	2
Main contactor Error	I1_TR	MainCErr	5	Alarm	1
Main contactor Error acknowledged	I1_TR	MainCErr	5	Normal	2
Monitor Low tripped	I1_TR	MonLow	6	Alarm	1
Monitor Low restored	I1_TR	MonLow	6	Normal	2
Monitor High tripped	I1_TR	MonHigh	7	Alarm	1
Monitor High restored	I1_TR	MonHigh	7	Normal	2
Position A reached	I1_TR	Pos A	8	Alarm	1
Position A left	I1_TR	Pos A	8	Normal	2
Current limit 100 % exceeded	I1_TR	HighCurr	11	Alarm	1
Current limit 100 % re-entered	I1_TR	HighCurr	11	Normal	2
Position B reached	I1_TR	Pos B	12	Alarm	1
Position B left	I1_TR	Pos B	12	Normal	2

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Control mode Local On	I2_TR	Local	2	On	1
Control mode Local Off	I2_TR	Local	2	Off	2
Control mode Test On	I2_TR	Test	3	On	1
Control mode Test Off	I2_TR	Test	3	Off	2
Control mode Standby On	I2_TR	Standby	4	On	1
Control mode Standby Off	I2_TR	Standby	4	Off	2
Control mode Sequence On	I2_TR	Sequence	5	On	1
Control mode Sequence Off	I2_TR	Sequence	5	Off	2
Op. mode Auto On	I2_TR	Auto	6	On	1
Op. mode Auto Off	I2_TR	Auto	6	Off	2
Interlock IC1 On	I2_TR	CInterl1	9	On	1
Interlock IC1 Off	I2_TR	CInterl1	9	Off	2
Interlock IC2 On	I2_TR	CInterl2	10	On	1
Interlock IC2 Off	I2_TR	CInterl2	10	Off	2
Interlock IB1 On	I2_TR	BInterl1	13	On	1
Interlock IB1 Off	I2_TR	BInterl1	13	Off	2
Interlock IB2 On	I2_TR	BInterl2	14	On	1
Interlock IB2 Off	I2_TR	BInterl2	14	Off	2
Interlock IB3 On	I2_TR	BInterl3	11	On	1
Interlock IB3 Off	I2_TR	BInterl3	11	Off	2
Interlock IB4 On	I2_TR	BInterl4	12	On	1
Interlock IB4 Off	I2_TR	BInterl4	12	Off	2

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Interlock IA On	I2_TR	AInterl	15	Off	1
Interlock IA Off	I2_TR	AInterl	15	Off	2
Control mode Local	ORD_TR	Local	1	Order	1
Control mode Test	ORD_TR	Test	2	Order	1
Control mode Standby	ORD_TR	Standby	3	Order	1
Control mode Central	ORD_TR	Central	4	Order	1
Control mode Sequence	ORD_TR	Sequence	13	Order	1
Op. mode Auto	ORD_TR	Auto	5	Order	1
Op. mode Manual	ORD_TR	Manual	6	Order	1
Blocking of IB1 and IB3 Interlock	ORD_TR	Blk.BI	7	Order	1
Deblocking of IB1 and IB3 Interlock	ORD_TR	Norm.BI	8	Order	1
Order Forward	ORD_TR	Forward	9	Order	1
Order Reverse	ORD_TR	Reverse	10	Order	1
Order Start	ORD_TR	Start	11	Order	1
Order Stop	ORD_TR	Stop	12	Order	1

Valve Control VALVECON

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
User defined Fault 1 coming	I1_TR	F1	1	Alarm	1
User defined Fault 1 passing	I1_TR	F1	1	Normal	2
User defined Fault 2 coming	I1_TR	F2	2	Alarm	1
User defined Fault 2 passing	I1_TR	F2	2	Normal	2
Position error open coming	I1_TR	PosErrO	3	Alarm	1
Position error open passing	I1_TR	PosErrO	3	Normal	2
Position error closed coming	I1_TR	PosErrC	4	Alarm	1
Position error closed passing	I1_TR	PosErrC	4	Normal	2
Valve in open position	I1_TR	Pos O	8	Alarm	1
Valve changes from open position	I1_TR	Pos O	8	Normal	2
Valve in closed position	I1_TR	Pos C	12	Alarm	1
Valve changes from closed position	I1_TR	Pos C	12	Normal	2
Valve in intermediate position	I1_TR	IntPos	16	Alarm	1
Valve not in intermediate position	I1_TR	IntPos	16	Normal	2
Control mode Local On	I2_TR	Local	2	On	1
Control mode Local Off	I2_TR	Local	2	Off	2
Control mode Test On	I2_TR	Test	3	On	1
Control mode Test Off	I2_TR	Test	3	Off	2
Control mode Standby On	I2_TR	Standby	4	On	1

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Control mode Standby Off	I2_TR	Standby	4	Off	2
Control mode Sequence On	I2_TR	Sequence	5	On	1
Control mode Sequence Off	I2_TR	Sequence	5	Off	2
Op. mode Auto On	I2_TR	Auto	6	On	1
Op. mode Auto Off	I2_TR	Auto	6	Off	2
Interlock IC1 On	I2_TR	CInterl1	9	On	1
Interlock IC1 Off	I2_TR	CInterl1	9	Off	2
Interlock IC2 On	I2_TR	CInterl2	10	On	1
Interlock IC2 Off	I2_TR	CInterl2	10	Off	2
Interlock IB1 On	I2_TR	BInterl1	11	On	1
Interlock IB1 Off	I2_TR	BInterl1	11	Off	2
Interlock IB2 On	I2_TR	BInterl2	12	On	1
Interlock IB2 Off	I2_TR	BInterl2	12	Off	2
Interlock IB3 On	I2_TR	BInterl3	13	On	1
Interlock IB3 Off	I2_TR	BInterl3	13	Off	2
Interlock IB4 On	I2_TR	BInterl4	14	On	1
Interlock IB4 Off	I2_TR	BInterl4	14	Off	2
Interlock IA On	I2_TR	AInterl	15	On	1
Interlock IA Off	I2_TR	AInterl	15	Off	2
Op. mode Local	ORD_TR	Local	1	Order	1
Op. mode Test	ORD_TR	Test	2	Order	1
Control mode Standby	ORD_TR	Standby	3	Order	1

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Control mode Central	ORD_TR	Central	4	Order	1
Control mode Sequence	ORD_TR	Sequence	11	Order	1
Op. mode Auto	ORD_TR	Auto	5	Order	1
Op. mode Manual	ORD_TR	Manual	6	Order	1
Blocking of IB1 and IB3 Interlock	ORD_TR	Blk.BI	7	Order	1
Deblocking of IB1 and IB3 Interlock	ORD_TR	Norm.BI	8	Order	1
Order Open	ORD_TR	Open	9	Order	1
Order Close	ORD_TR	Close	10	Order	1

Group Start GROUP

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Timeout sequence	I1_TR	Timeout G	1	Alarm	1
Timeout sequence acknowledged	I1_TR	Timeout G	1	Normal	2
Timeout step	I1_TR	Timeout S	2	Alarm	1
Timeout step acknowledged	I1_TR	Timeout S	2	Normal	2
User defined Fault 3 coming	I1_TR	Fault 3	3	Alarm	1
User defined Fault 3 passing	I1_TR	Fault 3	3	Normal	2
User defined Fault 4 coming	I1_TR	Fault 4	4	Alarm	1
User defined Fault 4 passing	I1_TR	Fault 4	4	Normal	2
User defined Fault 5 coming	I1_TR	Fault 5	5	Alarm	1
User defined Fault 5 passing	I1_TR	Fault 5	5	Normal	2
Position A reached	I1_TR	Pos A	8	Alarm	1
Position A left	I1_TR	Pos A	8	Normal	2
Position B reached	I1_TR	Pos B	12	Alarm	1
Position B left	I1_TR	Pos B	12	Normal	2
Control mode Local On	I2_TR	Local	2	On	1
Control mode Local Off	I2_TR	Local	2	Off	2
Control mode Test On	I2_TR	Test	3	On	1
Control mode Test Off	I2_TR	Test	3	Off	2
Control mode Standby On	I2_TR	Standby	4	On	1
Control mode Standby Of	I2_TR	Standby	4	Off	2f

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Control mode Sequence On	I2_TR	Sequence	5	On	1
Control mode Sequence Off	I2_TR	Sequence	5	Off	2
Op. mode Auto On	I2_TR	Auto	6	On	1
Op. mode Auto Off	I2_TR	Auto	6	Off	2
Interlock IC1 On	I2_TR	CInterl1	9	On	1
Interlock IC1 Off	I2_TR	CInterl1	9	Off	2
Interlock IC2 On	I2_TR	CInterl2	10	On	1
Interlock IC2 Off	I2_TR	CInterl2	10	Off	2
Interlock IB1 On	I2_TR	BInterl1	13	On	1
Interlock IB1 Off	I2_TR	BInterl1	13	Off	2
Interlock IB2 On	I2_TR	BInterl2	14	On	1
Interlock IB2 Off	I2_TR	BInterl2	14	Off	2
Interlock IB3 On	I2_TR	BInterl3	11	On	1
Interlock IB3 Off	I2_TR	BInterl3	11	Off	2
Interlock IB4 On	I2_TR	BInterl4	12	On	1
Interlock IB4 Off	I2_TR	BInterl4	12	Off	2
Interlock IA On	I2_TR	AInterl	15	On	1
Interlock IA Off	I2_TR	AInterl	15	Off	2

Group Alarm GRPALARM

Events	Event Treat pointer	Property text		Event text	
		Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
Abnormal position Off	DISTR_TR	Disturb	1	Alarm	1
Abnormal position On	DISTR_TR	Disturb	1	Normal	2
Blocking of event handling	DISTR_TR	Alarm	2	Blocked	3
Deblocking of event handling	DISTR_TR	Alarm	2	Deblocked	4
Blocking of printout	DISTR_TR	Printout	3	Blocked	3
Deblocking of printout	DISTR_TR	Printout	3	Deblocked	4
Blocking master On	PREV_TR	Blk Master	1	On	1
Blocking master Off	PREV_TR	Blk Master	1	Off	2
First error On	PREV_TR	First Err	2	On	1
First error Off	PREV_TR	First Err	2	Off	2
Prevent event handling On	PREV_TR	Event	3	Blocked	3
Prevent event handling Off	PREV_TR	Event	3	Deblocked	4
Blocking of the members event handling	PREV_TR	MembAlarm	4	Blocked	3
Deblocking of the members event handling	PREV_TR	MembAlarm	4	Deblocked	4
Blocking of the members event printout	PREV_TR	MembPrint	5	Blocked	3
Deblocking of the members event printout	PREV_TR	MembPrint	5	Deblocked	4

Motor Control MOTCONI

Alarms and events in this context are divided in three groups:

- Warnings, alarms from the Motor Controller Unit (MCU) which does not trip the motor. Acknowledge by operator. Color: Yellow
- Alarms, alarms from the MCU which trips the motor. Acknowledge by operator. Color: Red.
- Events, appears only in the event list. Not able to acknowledge. Color: Green.

Some alarms are reported from the PC element MOTCON directly, such as communication error. Most events and alarms are however generated and time stamped in the Motor Control Unit itself, for best possible time resolution, and sent via the LONWORKS network to the Controller.

Figure 88 below shows the principles for how the events are translated and associated to proper texts in the Operator Station.

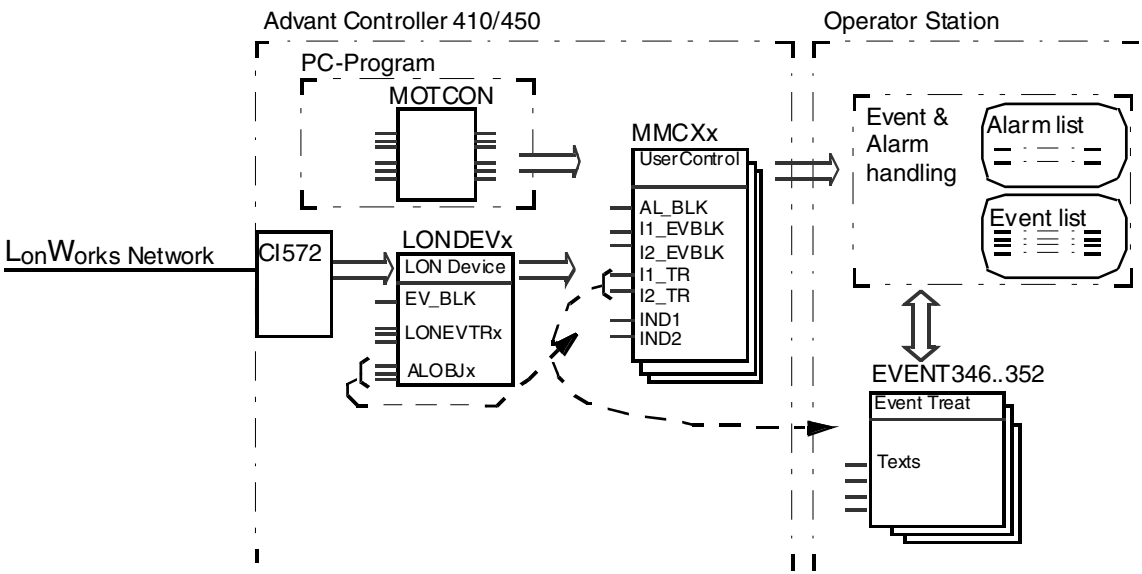


Figure 88. Event Handling from LONWORKS Network to Operator Station in Principles

In principle, the following processing is done in the Controller for every event message received from the MCU via network variable 2040:

- The LONDEV data base element that represents the sending device (MCU) is recognized.
- The LONDEV data base element points out three objects, of type MMCX. One object for warnings (ALOBJ1), one for alarms (ALOBJ2) and one for events(ALOBJ3). Each MMCX object can deal with up to 32 different warning/alarm/event indications. The event message is then forwarded to each alarm object.
- The MMCX object in it's turn forwards all state transitions to the Operator stations for presentation according to defined treatment and texts defined by terminals I1_TR or I2_TR. The treatment is divided in groups represented by data base element Event Treat. For each group different handling can be selected for status changes 0 to 1 and 1 to 0 respectively see Table E-6.

It is possible to block the update of events and alarms in different manners. It could be done either from the operators dialog affecting all alarms and events from each MCU. Separate alarms could be blocked by the bit masks I1_EVBLK and I2_EVBLK in MMCX data base element. The terminal EV_BLK in LONDEV data base element could also be used to block events received from each MCU.

The translation of events and alarms for MOTCONI is described below. They are divided into four groups. For each group one data base element MMCX is needed. The groups are:

- Alarm from MOTCON PC element
- Warnings directly from MCU
- Alarm directly from MCU
- Events directly from MCU

See table below for network variable bindings that has to be done to correctly receive events and alarms from MCU.

Table 12. Network Variable bindings needed for event and alarm handling from MCU

Description	SNVT	CI572		MCU	
		NV-Index	NV-Name	NV-Index	NV-Name
Clock Warning, broadcast from CI572 before the clock synchronization message	0	2042	nvoClockWrng	8	nviClockWrng
Clock synchronization broadcast message. The send interval is defined by LONCHAN database element	0	2041	nvoClock	7	nviClock
Alarm Report message with warnings, alarms, events and time stamp	0	2040	nviAlarmReport	51	nvoAlarmReport

Standard Alarm Text from MOTCON PC Element

Table 13. Alarms reported from MOTCON PC element, translated via data base element EVENT352

Description	IND bit in MMCX	Property text	Event Text
Communication error	IND1_00	CommErr	Warning
			Normal
Over load	IND1_01	Overload	Warning
			Normal
Local Stop	IND1_02	LStop	Warning
			Normal

Table 13. Alarms reported from MOTCON PC element, translated via data base element EVENT352 (Continued)

Emergency stop	IND1_03	EStop	Warning
			Normal
Main contactor error	IND1_04	MainCErr	Warning
			Normal
Monitor low	IND1_05	MonLow	Warning
			Normal
Monitor high	IND1_06	MonHigh	Warning
			Normal

Standard Warning Texts from MCU

Table 14. Warnings reported from MCU, translated via data base element EVENT346 or EVENT347

Description	IND bit in MMCX	Event Treat element	Property text	Event Text
Thermal Overload	IND1_00	EVENT346	ThermOL	Warning
				Normal
Phase Loss L1	IND1_01	EVENT346	PhLossL1	Warning
				Normal
Phase Loss L2	IND1_02	EVENT346	PhLossL2	Warning
				Normal
Phase Loss L3	IND1_03	EVENT346	PhLossL3	Warning
				Normal
Under load	IND1_04	EVENT346	UndLoad	Warning
				Normal
No load	IND1_05	EVENT346	NoLoad	Warning
				Normal
Stall	IND1_06	EVENT346	Stall	Warning
				Normal
Feedback CFA	IND1_07	EVENT346	FeedbCFA	Warning
				Normal
Feedback CFB	IND1_08	EVENT346	FeedbCFB	Warning
				Normal

Table 14. Warnings reported from MCU, translated via data base element EVENT346 or EVENT347 (Continued)

Feedback CFC	IND1_09	EVENT346	FeedbCFC	Warning
				Normal
Earth Fault	IND1_10	EVENT346	EarthFlt	Warning
				Normal
Unbalance	IND1_11	EVENT346	UnBalanc	Warning
				Normal
U/L Cosphi	IND1_12	EVENT346	Cosphi	Warning
				Normal
Rotation	IND1_13	EVENT346	Rotation	Warning
				Normal
PTC temperature	IND1_14	EVENT346	PTCtemp	Warning
				Normal
Under Voltage	IND1_15	EVENT346	UnderVlt	Warning
				Normal
Start limitation	IND2_00	EVENT347	StartLim	Warning
				Normal
Autoreclosuer	IND2_01	EVENT347	AutoRecl	Warning
				Normal
Device temperature	IND2_02	EVENT347	Dev temp	Warning
				Normal
O/L	IND2_03	EVENT347	Overload	Warning
				Normal

Table 14. Warnings reported from MCU, translated via data base element EVENT346 or EVENT347 (Continued)

Maintenance A	IND2_04	EVENT347	MaintenA	Warning
				Normal
Maintenance B	IND2_05	EVENT347	MaintenB	Warning
				Normal
Maintenance C	IND2_06	EVENT347	MaintenC	Warning
				Normal
Maintenance hour run	IND2_07	EVENT347	MaintMot	Warning
				Normal
Fail-safe activated	IND2_08	EVENT347	FailSafe	Warning
				Normal
No external memory	IND2_09	EVENT347	NoExtROM	Warning
				Normal
ROM write fail	IND2_10	EVENT347	ROMWrFai	Warning
				Normal
ROM read fail	IND2_11	EVENT347	ROMRdFai	Warning
				Normal
Drawer location	IND2_12	EVENT347	DrawLoc	Warning
				Normal
PTC short circuit	IND2_13	EVENT347	PTCshort	Warning
				Normal
PTC open circuit	IND2_14	EVENT347	PTCopen	Warning
				Normal

Table 14. Warnings reported from MCU, translated via data base element EVENT346 or EVENT347 (Continued)

Startup inhibit	IND2_15	EVENT347	StrtInhb	Warning
				Normal

Standard Alarm Text from MCU

Table 15. Alarms reported from MCU, translated via data base element EVENT348 or EVENT349

Description	IND bit in MMCX	Event Treat element	Property text	Event Text
Thermal Overload trip	IND1_00	EVENT348	ThermOL	Warning
				Normal
Phase Loss trip L1	IND1_01	EVENT348	PhLossL1	Warning
				Normal
Phase Loss trip L2	IND1_02	EVENT348	PhLossL2	Warning
				Normal
Phase Loss trip L3	IND1_03	EVENT348	PhLossL3	Warning
				Normal
Under load trip	IND1_04	EVENT348	UndLoad	Warning
				Normal
No load trip	IND1_05	EVENT348	NoLoad	Warning
				Normal
Stall trip	IND1_06	EVENT348	Stall	Warning
				Normal

Table 15. Alarms reported from MCU, translated via data base element EVENT348 or EVENT349

Feedback trip CFA	IND1_07	EVENT348	FeedbCFA	Warning
				Normal
Feedback trip CFB	IND1_08	EVENT348	FeedbCFB	Warning
				Normal
Feedback trip CFC	IND1_09	EVENT348	FeedbCFC	Warning
				Normal
Earth fault trip	IND1_10	EVENT348	EarthFlt	Warning
				Normal
Unbalance trip	IND1_11	EVENT348	UnBalanc	Warning
				Normal
U/L Cosphi trip	IND1_12	EVENT348	Cosphi	Warning
				Normal
Rotation trip	IND1_13	EVENT348	Rotation	Warning
				Normal
PTC temperature trip	IND1_14	EVENT348	PTCtemp	Warning
				Normal
Under Voltage trip	IND1_15	EVENT348	UnderVlt	Warning
				Normal
Start limitation trip	IND2_00	EVENT349	StartLim	Warning
				Normal
PTC short circuit trip	IND2_01	EVENT349	PTCshort	Warning
				Normal

Table 15. Alarms reported from MCU, translated via data base element EVENT348 or EVENT349

PTC open circuit trip	IND2_02	EVENT349	PTCopen	Warning
				Normal
Torque trip	IND2_03	EVENT349	TorqTrip	Warning
				Normal
Parametering failure	IND2_04	EVENT349	ParamFail	Warning
				Normal
External trip	IND2_05	EVENT349	ExtTrip	Warning
				Normal
Test mode failure trip	IND2_06	EVENT349	TestTrip	Warning
				Normal
EM stop activated	IND2_07	EVENT349	EMstop	Warning
				Normal
Internal fault trip	IND2_08	EVENT349	InterFlt	Warning
				Normal
External trip command	IND2_09	EVENT349	ExTrpCmd	Warning
				Normal
Main switch OFF	IND2_10	EVENT349	MainSwOf	Warning
				Normal
spare	IND2_11	EVENT349	spare	Warning
				Normal
MCB trip	IND2_12	EVENT349	MCBtrip	Warning
				Normal

Table 15. Alarms reported from MCU, translated via data base element EVENT348 or EVENT349

Motor still running	IND2_13	EVENT349	MotIsRun	Warning
				Normal
Start Interlock trip	IND2_14	EVENT349	StrtIntl	Warning
				Normal

Standard Event Text from MCU

Table 16. EventsreportedfromMCU,translatedviadatabaseelementEVENT350 or EVENT351

Description	IND bit in MMCX	Event Treat element	Property text	Event Text
TOL reset level reached	IND1_00	EVENT350	TolResLv	McuEvent
Motor started 1	IND1_01	EVENT350	MotStrt1	McuEvent
Motor started 2	IND1_02	EVENT350	MotStrt2	McuEvent
Motor stopped	IND1_03	EVENT350	MotorStp	McuEvent
Control mode ‘local’	IND1_04	EVENT350	Local	McuEvent
Main switch in Test-position	IND1_05	EVENT350	TestPos	McuEvent
Trip reset	IND1_06	EVENT350	TripRes	McuEvent
Device set to off-line	IND1_07	EVENT350	Off-line	McuEvent
Motor started CW by RCU-switch	IND1_08	EVENT350	RcuStrt1	McuEvent
Motor stopped by RCU-switch	IND1_09	EVENT350	RcuStop	McuEvent

Table 16. EventsreportedfromMCU,translatedviadatabaseelementEVENT350 or EVENT351 (Continued)

Motor started CCW by RCU-switch	IND1_10	EVENT350	RcuStrt2	McuEvent
Trip bypass activated	IND1_11	EVENT350	TrpBypas	McuEvent
Motor stopped by limit position 1	IND1_12	EVENT350	StopLim1	McuEvent
Motor stopped by limit position 2	IND1_13	EVENT350	StopLim2	McuEvent
Start interlock alarm	IND2_01	EVENT351	StrtIntl	McuEvent

Event Treatment in Operator Station

The treatment of events and alarms in Operator Station (such as color of event line, audible alarm or not, text line appearance) is defined in the EVENT Treat database element. The table below describes the predefined settings for different Event Treat elements for MOTCONI.

Table 17. Predefined Event treatment for MOTCONI

Terminal	Index of EVENT treatment data base element						
	352	346	347	348	349	350	351
AUDIBLE	1	1	1	1	1	0	0
AL_PRIO	2	3	3	2	2	4	4
AL_TOBLK	No	No	No	No	No	Yes	Yes
AL_FRBLK	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PERSISTB	No	No	No	No	No	Yes	Yes
TEXT_TOB	No	No	No	No	No	No	No
TEXT_FRB	No	No	No	No	No	Yes	Yes
TEXTCOMB	5	5	5	5	5	5	5

Engineered Drive, DRICONE

This section describes the principles for events and alarms, the standard event texts for DRICONE main data base element and DRICONE fault and alarm data base elements. The section contains also a layout of the Event Treat configuration for DRICONE.

Figure 89 shows diagrammatically the relation between indications and the parameters which can be used to block event handling for certain indications and points out a required handling in the operator's station.

The possibilities of the operator to block event printouts and alarms are shown under the heading 'Event and alarm blocking' below.

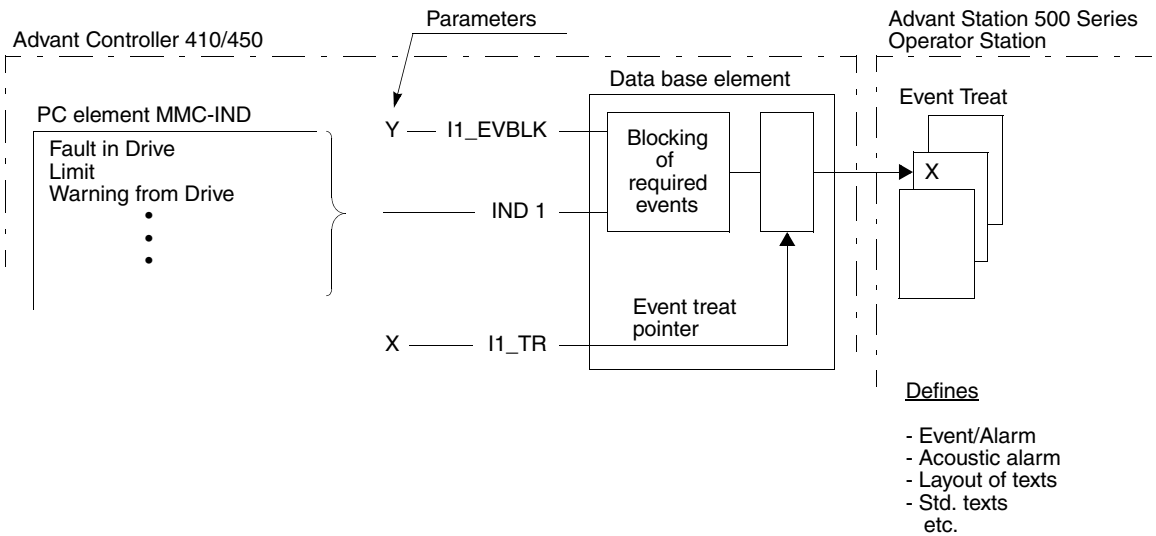


Figure 89. Event Handling Principles

The events are divided into groups. One pointer is allocated to each group which means that the individual events of the group are, in several respects, handled in the same way in Advant Station 500 Series Operator Station. For each group different handling can be selected for status changes 0-->1 and 1-->0 respectively. The texts which are written out in different lists are individual for each event. The group

division - which events are associated with the pointer - is given in ‘Event description with standard texts’ below. This also specifies which event handling is obtained as default and which alternative standard function.

Table 18. Event Text for Main MMCX Data Base Element, EVENT Treat element 334 - 335

Signal	Events	Event Treat pointer	Property text		Event Text	
			Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
IND1_01	Out of window	I1_TR	Speed Outside Window	2	Alarm	1
IND1_01	Within window	I1_TR	Speed Outside Window	2	Normal	2
IND1_02	Emergency stop	I1_TR	Emergency Stop Fault	3	Alarm	1
IND1_02	No Emergency stop	I1_TR	Emergency Stop Fault	3	Normal	2
IND1_04	Run disabled	I1_TR	Run Disabled	5	Alarm	1
IND1_04	Run enabled	I1_TR	Run Disabled	5	Normal	2
IND1_06	First start not done	I1_TR	First start not done	7	Alarm	1
IND1_06	First start done	I1_TR	First start not done	7	Normal	2
IND1_07	Alarm from drive	I1_TR	Alarm from drive	8	Alarm	1
IND1_07	No alarm	I1_TR	Alarm from drive	8	Normal	2
IND1_08	Start inhibition	I1_TR	False Start Inhibit	9	Alarm	1

Table 18. Event Text for Main MMCX Data Base Element, EVENT Treat element 334 - 335 (Continued)

IND1_08	No start inhibition	I1_TR	False Start Inhibit	9	Normal	2
IND1_09	Limit exceeded	I1_TR	Limit	10	Alarm	1
IND1_09	No Limit	I1_TR	Limit	10	Normal	2
IND1_11	Fault from drive	I1_TR	Fault in Drive	12	Alarm	1
IND1_11	No fault	I1_TR	Fault in Drive	12	Normal	2
IND1_12	Communication fault	I1_TR	Communication Error	13	Alarm	1
IND1_12	Communication fault	I1_TR	Communication Error	13	Normal	2
IND2_00	In service ready to switch on	I2_TR	In service	1	On	1
IND2_00	Not in service	I2_TR	In service	1	Off	2
IND2_01	Drive ready to start	I2_TR	Ready to Start	2	On	1
IND2_01	Drive is not ready to start	I2_TR	Ready to Start	2	Off	2
IND2_02	Running with selected reference	I2_TR	Run	3	On	1
IND2_02	Stopped	I2_TR	Run	3	Off	2
IND2_03	Motor is magnetized	I2_TR	Motor Magnetized	4	On	1
IND2_03	Motor is not magnetized	I2_TR	Motor Magnetized	4	Off	2

Table 18. Event Text for Main MMCX Data Base Element, EVENT Treat element 334 - 335 (Continued)

IND2_04	OFF 2	I2_TR	Emergency	5	On	5
IND2_04	No OFF 2	I2_TR	Emergency	5	Off	6
IND2_05	OFF 3	I2_TR	Emergency stop	6	On	5
IND2_05	No OFF 3	I2_TR	Emergency stop	6	Off	6
IND2_06	Switch on is inhibited	I2_TR	Inhibit Switch On	7	Yes	3
IND2_06	Switch on is not inhibited	I2_TR	Inhibit Switch On	7	No	4
IND2_07	Drive runs at setpoint	I2_TR	At Setpoint	8	On	1
IND2_07	Drive is not at setpoint	I2_TR	At Setpoint	8	Off	2
IND2_08	C-Interlock	I2_TR	C-Interlock	9	On	1
IND2_08	C-Interlock	I2_TR	C-Interlock	9	Off	2
IND2_09	Drive runs remotely controlled	I2_TR	Remote	10	On	1
IND2_09	Drive runs locally controlled	I2_TR	Remote	10	Off	2
IND2_10	B-Interlock 1	I2_TR	B-Interlock 1	11	On	1
IND2_10	B-Interlock 1	I2_TR	B-Interlock 1	11	Off	2
IND2_11	B-Interlock 2	I2_TR	B-Interlock 2	12	On	1
IND2_11	B-Interlock 2	I2_TR	B-Interlock 2	12	Off	2
IND2_12	B-Interlock 3	I2_TR	B-Interlock 3	13	On	1
IND2_12	B-Interlock 3	I2_TR	B-Interlock 3	13	Off	2

Table 18. Event Text for Main MMCX Data Base Element, EVENT Treat element 334 - 335 (Continued)

IND2_13	B-Interlock 4	I2_TR	B-Interlock 4	14	On	1
IND2_13	B-Interlock 4	I2_TR	B-Interlock 4	14	Off	2
IND2_14	A-Interlock	I2_TR	A-Interlock	15	On	1
IND2_14	A-Interlock	I2_TR	A-Interlock	15	Off	2
IND2_15	Override interlock	I2_TR	Override Interlock	16	On	1
IND2_15	Don't override interlock	I2_TR	Override Interlock	16	Off	2

Table 19. Event Text for Fault MMCX Data Base Element ACS type of Drive, EVENT Treat element 336 - 337

Signal	Events	Event Treat pointer	Property text		Event Text	
			Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
IND1_00	Short circuit	I1_TR	Short Circuit	1	Fault	1
IND1_00	Short circuit	I1_TR	Short Circuit	1	Normal	2
IND1_01	Over current	I1_TR	Over Current	2	Fault	1
IND1_01	Over current	I1_TR	Over Current	2	Normal	2
IND1_02	DC over voltage fault	I1_TR	DC Over Voltage	3	Fault	1
IND1_02	No DC over voltage fault	I1_TR	DC Over Voltage	3	Normal	2

Table 19. Event Text for Fault MMCX Data Base Element ACS type of Drive, EVENT Treat element 336 - 337 (Continued)

IND1_03	Power plate over temperature fault	I1_TR	Over Temp Pow Plate	4	Fault	1
IND1_03	No power plate over temperature fault	I1_TR	Over Temp Pow Plate	4	Normal	2
IND1_04	Earth fault	I1_TR	Earth Fault	5	Fault	1
IND1_04	No earth fault	I1_TR	Earth Fault	5	Normal	2
IND1_05	Motor over temperature fault	I1_TR	Over Temp Motor	6	Fault	1
IND1_05	No motor over temperature fault	I1_TR	Over Temp Motor	6	Normal	2
IND1_06	Motor over load fault	I1_TR	Over Load Motor	7	Fault	1
IND1_06	No motor over load fault	I1_TR	Over Load Motor	7	Normal	2
IND1_07	System fault	I1_TR	System Fault	8	Fault	1
IND1_07	No system fault	I1_TR	System Fault	8	Normal	2
IND1_08	Under load fault	I1_TR	Under Load	9	Fault	1
IND1_08	No under load fault	I1_TR	Under Load	9	Normal	2
IND1_09	Over speed fault	I1_TR	Over Speed	10	Fault	1
IND1_09	No over speed fault	I1_TR	Over Speed	10	Normal	2
IND1_10	Supply Section fault	I1_TR	Supply Section	11	Fault	1
IND1_10	No Supply Section fault	I1_TR	Supply Section	11	Normal	2
IND1_11	Master/Follower fault	I1_TR	Master/Follower	12	Fault	1
IND1_11	No Master/Follower fault	I1_TR	Master/Follower	12	Normal	2

Table 19. Event Text for Fault MMCX Data Base Element ACS type of Drive, EVENT Treat element 336 - 337 (Continued)

IND1_12	Short circuit INT1	I1_TR	Short circuit INT1	13	Fault	1
IND1_12	No Short circuit INT1	I1_TR	Short Circuit Int1	13	Normal	2
IND1_13	Short circuit INT2	I1_TR	Short Circuit Int2	14	Fault	1
IND1_13	No Short circuit INT2	I1_TR	Short Circuit Int2	14	Normal	2
IND1_14	Short circuit INT3	I1_TR	Short Circuit Int3	15	Fault	1
IND1_14	No Short circuit INT3	I1_TR	Short Circuit Int3	15	Normal	2
IND1_15	Short circuit INT4	I1_TR	Short Circuit Int4	16	Fault	1
IND1_15	No Short circuit INT4	I1_TR	Short Circuit Int4	16	Normal	2
IND2_00	Supply section fault	I2_TR	Supply Section	1	Fault	1
IND2_00	No supply section fault	I2_TR	Supply Section	1	Normal	2
IND2_02	DC under voltage	I2_TR	DC Under Voltage	3	Fault	1
IND2_02	No DC under voltage	I2_TR	DC Under Voltage	3	Normal	2
IND2_04	Run disabled	I2_TR	Run Disabled	5	Fault	1
IND2_04	Run enabled	I2_TR	Run Disabled	5	Normal	2
IND2_05	Encoder fault	I2_TR	Speed Encoder	6	Fault	1
IND2_05	No encoder fault	I2_TR	Speed Encoder	6	Normal	2

Table 19. Event Text for Fault MMCX Data Base Element ACS type of Drive, EVENT Treat element 336 - 337 (Continued)

IND2_06	I/O link fault channel 1	I2_TR	I/O Link channel 1	7	Fault	1
IND2_06	No I/O link fault channel 1	I2_TR	I/O Link channel 1	7	Normal	2
IND2_07	Cabinett overtemperature	I2_TR	Cabinett overtemp	8	Fault	1
IND2_07	No Cabinett overtemperature	I2_TR	Cabinett overtemp	8	Normal	2
IND2_09	Over svitching frequency	I2_TR	Over Switching	10	Fault	1
IND2_09	No over switching frequency	I2_TR	Over Switching	10	Normal	2
IND2_11	PPCC link fault	I2_TR	PPCC Link	12	Fault	1
IND2_11	No PPCC link fault	I2_TR	PPCC Link	12	Normal	2
IND2_12	Communication fault Channel 0	I2_TR	Communication Ch 0	13	Fault	1
IND2_12	No Communication fault Channel 0	I2_TR	Communication Ch 0	13	Normal	2
IND2_13	Pannel loss	I2_TR	Pannel loss	14	Fault	1
IND2_13	No pannel loss	I2_TR	Pannel loss	14	Normal	2
IND2_14	Motor stalled	I2_TR	Motor Stalled	15	Fault	1
IND2_14	No motor stalled	I2_TR	Motor Stalled	15	Normal	2
IND2_15	Motor phase missing	I2_TR	Motor Phase Missing	16	Fault	1
IND2_15	No motor phase missing	I2_TR	Motor Phase Missing	16	Normal	2

Table 20. Event Text for Alarm MMCX Data Base Element ACS type of Drive, EVENT Treat element 338 - 339

Signal	Events	Event Treat pointer	Property text		Event Text	
			Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
IND1_00	Prevent unexpected start	I1_TR	Prevent Start	1	Alarm	1
IND1_00	No preventation of unexpected start	I1_TR	Prevent Start	1	Normal	2
IND1_01	Emergency stop	I1_TR	Emergency Stop	2	Alarm	1
IND1_01	Emergency stop	I1_TR	Emergency Stop	2	Normal	2
IND1_02	Measured motor temp alarm	I1_TR	Meas. Motor Temp	3	Alarm	1
IND1_02	Measured motor temp alarm	I1_TR	Meas. Motor Temp	3	Normal	2
IND1_03	Over temperature alarm of the thermal model	I1_TR	Motor Overload	4	Alarm	1
IND1_03	No over temperature alarm of the thermal model	I1_TR	Motor Overload	4	Normal	2
IND1_04	Power plate over temperature alarm	I1_TR	Temp Power Plate	5	Alarm	1
IND1_04	No power plate over temperature alarm	I1_TR	Temp Power Plate	5	Normal	2

Table 20. Event Text for Alarm MMCX Data Base Element ACS type of Drive, EVENT Treat element 338 - 339 (Continued)

IND1_05	Pulse encoder alarm	I1_TR	Pulse Encoder	6	Alarm	1
IND1_05	No Pulse encoder alarm	I1_TR	Pulse Encoder	6	Normal	2
IND1_07	Standard digital I/O alarm	I1_TR	Standard Digital I/O	8	Alarm	1
IND1_07	No Standard digital I/O alarm	I1_TR	Standard Digital I/O	8	Normal	2
IND1_08	Standard analogue I/O alarm	I1_TR	Standard Analog I/O	9	Alarm	1
IND1_08	No Standard analogue I/O alarm	I1_TR	Standard Analog I/O	9	Normal	2
IND1_09	External digital I/O alarm	I1_TR	Ext. Digital I/O	10	Alarm	1
IND1_09	No External digital I/O alarm	I1_TR	Ext. Digital I/O	10	Normal	2
IND1_10	External analogue I/O alarm	I1_TR	Ext. Analog I/O	11	Alarm	1
IND1_10	No External analogue I/O alarm	I1_TR	Ext. Analog I/O	11	Normal	2
IND1_11	Master - Follower alarm	I1_TR	Master - Follow alarm	12	Alarm	1
IND1_11	Master - Follower alarm	I1_TR	Master - Follow alarm	12	Normal	2
IND1_12	Communication alarm Channel 0	I1_TR	Comm. Channel 0	13	Alarm	1
IND1_12	Communication alarm Channel 0	I1_TR	Comm. Channel 0	13	Normal	2
IND1_14	Earth fault alarm	I1_TR	Earth fault	15	Alarm	1

Table 20. Event Text for Alarm MMCX Data Base Element ACS type of Drive, EVENT Treat element 338 - 339 (Continued)

IND1_14	No Earth fault alarm	I1_TR	Earth fault	15	Normal	2
IND2_00	Power failure	I2_TR	Power Failure	1	Alarm	1
IND2_00	No power failure	I2_TR	Power Failure	1	Normal	2
IND2_01	Under load alarm	I2_TR	Under Load	2	Alarm	1
IND2_01	No under load alarm	I2_TR	Under Load	2	Normal	2
IND2_02	Identity run	I2_TR	Identity run	3	Alarm	1
IND2_02	No Identity run	I2_TR	Identity run	3	Normal	2
IND2_03	DC under voltage alarm	I2_TR	DC Under Voltage	5	Alarm	1
IND2_03	No DC under voltage alarm	I2_TR	DC Under Voltage	5	Normal	2
IND2_04	DC over voltage alarm	I2_TR	DC Over Voltage	5	Alarm	1
IND2_04	No DC over voltage alarm	I2_TR	DC Over Voltage	5	Normal	2
IND2_05	Over current alarm	I2_TR	Over Current	6	Alarm	1
IND2_05	No over current alarm	I2_TR	Over Current	6	Normal	2
IND2_06	Over frequency alarm	I2_TR	Over Frequency	7	Alarm	1
IND2_06	No over frequency alarm	I2_TR	Over Frequency	7	Normal	2
IND2_07	Power failure file	I2_TR	Power Failure file	8	Alarm	1
IND2_07	No power failure file	I2_TR	Power Failure file	8	Normal	2

Table 20. Event Text for Alarm MMCX Data Base Element ACS type of Drive, EVENT Treat element 338 - 339 (Continued)

IND2_08	Power down file	I2_TR	Power Down file	8	Alarm	1
IND2_07	No power down file	I2_TR	Power Down file	8	Normal	2
IND2_08	Motor stalled alarm	I2_TR	Stall	9	Alarm	1
IND2_08	No motor stalled alarm	I2_TR	Stall	9	Normal	2
IND2_09	Supply phase	I2_TR	Supply Phase	10	Alarm	1
IND2_09	No supply phase missing	I2_TR	Supply Phase	10	Normal	2
IND2_10	Timeout channel 0	I2_TR	DDCS Timeout Ch0	11	Alarm	1
IND2_10	No timeout channel 0	I2_TR	DDCS Timeout Ch0	11	Normal	2
IND2_13	Pannel loss	I2_TR	Pannel loss	14	Alarm	1
IND2_13	No pannel loss	I2_TR	Pannel loss	14	Normal	2

Table 21. Event Text for Main MMCX Data Base Element DCS type of Drive, EVENT Treat element 340 - 341

Signal	Events	Event Treat pointer	Property text		Event Text	
			Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
IND1_07	Alarm from drive	I1_TR	Alarm from drive	8	Alarm	1
IND1_07	No alarm	I1_TR	Alarm from drive	8	Normal	2

Table 21. Event Text for Main MMCX Data Base Element DCS type of Drive, EVENT Treat element 340 - 341 (Continued)

IND1_11	Fault from drive	I1_TR	Fault in Drive	12	Alarm	1
IND1_11	No fault	I1_TR	Fault in Drive	12	Normal	2
IND1_12	Communication fault	I1_TR	Communication Error	13	Alarm	1
IND1_12	Communication fault	I1_TR	Communication Error	13	Normal	2
IND2_00	In service ready to switch on	I2_TR	In service	1	On	1
IND2_00	Not in service	I2_TR	In service	1	Off	2
IND2_01	Drive ready to start	I2_TR	Ready to Start	2	On	1
IND2_01	Drive is not ready to start	I2_TR	Ready to Start	2	Off	2
IND2_02	Running with selected reference	I2_TR	Run	3	On	1
IND2_02	Stopped	I2_TR	Run	3	Off	2
IND2_08	C-Interlock	I2_TR	C-Interlock	9	On	1
IND2_08	C-Interlock	I2_TR	C-Interlock	9	Off	2
IND2_10	B-Interlock 1	I2_TR	B-Interlock 1	11	On	1
IND2_10	B-Interlock 1	I2_TR	B-Interlock 1	11	Off	2
IND2_11	B-Interlock 2	I2_TR	B-Interlock 2	12	On	1
IND2_11	B-Interlock 2	I2_TR	B-Interlock 2	12	Off	2
IND2_12	B-Interlock 3	I2_TR	B-Interlock 3	13	On	1
IND2_12	B-Interlock 3	I2_TR	B-Interlock 3	13	Off	2
IND2_13	B-Interlock 4	I2_TR	B-Interlock 4	14	On	1

Table 21. Event Text for Main MMCX Data Base Element DCS type of Drive, EVENT Treat element 340 - 341 (Continued)

IND2_13	B-Interlock 4	I2_TR	B-Interlock 4	14	Off	2
IND2_14	A-Interlock	I2_TR	A-Interlock	15	On	1
IND2_14	A-Interlock	I2_TR	A-Interlock	15	Off	2
IND2_15	Override interlock	I2_TR	Override Interlock	16	On	1
IND2_15	Don't override interlock	I2_TR	Override Interlock	16	Off	2

Table 22. Event Text for Fault MMCX Data Base Element DCS type of Drive, EVENT Treat element 342 - 343

Signal	Events	Event Treat pointer	Property text		Event Text	
			Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
IND1_00	Auxiliary under voltage	I1_TR	Auxil. Undervoltage	1	Fault	1
IND1_00	Auxiliary under voltage	I1_TR	Auxil. Undervoltage	1	Normal	2
IND1_01	Over current	I1_TR	Over Current	2	Fault	1
IND1_01	No Over current	I1_TR	Over Current	2	Normal	2
IND1_02	Armature over voltage fault	I1_TR	Armature Overvolt.	3	Fault	1
IND1_02	No Armature over voltage fault	I1_TR	Armature Overvolt.	3	Normal	2

Table 22. Event Text for Fault MMCX Data Base Element DCS type of Drive, EVENT Treat element 342 - 343 (Continued)

IND1_03	Converter over temperature fault	I1_TR	Overtemp Conv.	4	Fault	1
IND1_03	No converter over temperature fault	I1_TR	Overtemp Conv.	4	Normal	2
IND1_04	Earth fault	I1_TR	Earth fault	5	Fault	1
IND1_04	No earth fault	I1_TR	Earth fault	5	Normal	2
IND1_05	Motor 1 over temperature fault	I1_TR	Motor 1 overtemp	6	Fault	1
IND1_05	No over temperature fault motor 1	I1_TR	Motor 1 overtemp	6	Normal	2
IND1_06	Motor 1 over load fault	I1_TR	Motor 1 overload	7	Fault	1
IND1_06	No over load fault motor 1	I1_TR	Motor 1 overload	7	Normal	2
IND1_07	I/O-board not found	I1_TR	I/O-board not found	8	Fault	1
IND1_07	I/O-board found	I1_TR	I/O-board not foundt	8	Normal	2
IND1_08	Motor 2 over temperature fault	I1_TR	Motor 2 overtemp	9	Fault	1
IND1_08	No over temperature fault motor 2	I1_TR	Motor 2 overtemp	9	Normal	2
IND1_09	Motor 2 over load fault	I1_TR	Motor 2 overload	10	Fault	1
IND1_09	No over load fault motor 2	I1_TR	Motor 2 overload	10	Normal	2

Table 22. Event Text for Fault MMCX Data Base Element DCS type of Drive, EVENT Treat element 342 - 343 (Continued)

IND1_11	Main supply under voltage	I1_TR	Mains undervoltage	12	Alarm	1
IND1_11	No main supply under voltage	I1_TR	Mains undervoltage	12	Normal	2
IND1_12	Main supply over voltage	I1_TR	Mains overvoltage	13	Fault	1
IND1_12	No main supply over voltage	I1_TR	Mains overvoltage	13	Normal	2
IND1_13	Not synchronized	I1_TR	Synchronism	14	Fault	1
IND1_13	Synchronized	I1_TR	Synchronism	14	Normal	2
IND1_14	Field exciter 1 over current	I1_TR	Fld ex 1 overcurrent	15	Fault	1
IND1_14	No over current field exciter 1	I1_TR	Fld ex 1 overcurrent	15	Normal	2
IND1_15	Field exciter 1 communication fault	I1_TR	Fld ex 1 commerror	16	Fault	1
IND1_15	No communication fault field exciter 1	I1_TR	Fld ex 1 commerror	16	Normal	2
IND2_00	Armature current ripple	I2_TR	Arm current ripple	1	Fault	1
IND2_00	No armature current ripple	I2_TR	Arm current ripple	1	Normal	2
IND2_01	Field exciter 2 over current	I1_TR	Fld ex 2 overcurrent	2	Fault	1

Table 22. Event Text for Fault MMCX Data Base Element DCS type of Drive, EVENT Treat element 342 - 343 (Continued)

IND2_01	No over current field exciter 2	I1_TR	Fld ex 2 overcurrent	2	Normal	2
IND2_02	Field exciter 2 communication fault	I1_TR	Fld ex 2 commerror	3	Fault	1
IND2_02	No communication fault field exciter 2	I1_TR	Fld ex 2 commerror	3	Normal	2
IND2_03	Phase sequence fault	I2_TR	Phase sequence	4	Fault	1
IND2_03	No phase sequence fault	I2_TR	Phase sequence	4	Normal	2
IND2_04	No field acknowledge	I2_TR	Field acknowledge	5	Fault	1
IND2_04	Field acknowledge	I2_TR	Field acknowledge	5	Normal	2
IND2_05	Measured speed fault	I2_TR	Speed measured	6	Fault	1
IND2_05	No measured speed fault	I2_TR	Speed measured	6	Normal	2
IND2_06	No external fan acknowledge	I2_TR	Ext. Fan acknowledge	7	Fault	1
IND2_06	External fan acknowledge	I2_TR	Ext. Fan acknowledge	7	Normal	2
IND2_07	No main contactor acknowledge	I2_TR	Main cont ack	8	Fault	1

Table 22. Event Text for Fault MMCX Data Base Element DCS type of Drive, EVENT Treat element 342 - 343 (Continued)

IND2_07	Main contactor acknowledge	I2_TR	Main cont ack	8	Normal	2
IND2_08	Type coding fault	I2_TR	Type coding	9	Fault	1
IND2_08	No type coding fault	I2_TR	Type coding	9	Normal	2
IND2_09	Parameter backup fault	I2_TR	Par backup	10	Fault	1
IND2_09	No parameter backup fault	I2_TR	Par backup	10	Normal	2
IND2_10	No central fan acknowledge	I2_TR	C Fan ack	11	Fault	1
IND2_10	Central fan acknowledge	I2_TR	C Fan ack	11	Normal	2
IND2_11	DDCS ch. 0 communication fault	I2_TR	DDCS Comm. Ch 0.	12	Fault	1
IND2_11	No DDCS ch. 0 communication faultr	I2_TR	DDCS Comm. Ch 0.	12	Normal	2
IND2_12	Field exciter 1 fault	I1_TR	Fld ex 1	13	Fault	1
IND2_12	Field exciter 1 ok	I1_TR	Fld ex 1	13	Normal	2
IND2_13	Field exciter 2 fault	I1_TR	Fld ex 2	14	Fault	1
IND2_13	Field exciter 2 ok	I1_TR	Fld ex 2	14	Normal	2
IND2_14	Motor stalled	I2_TR	Motor stalled	15	Fault	1
IND2_14	No motor stall	I2_TR	Motor stalled	15	Normal	2

Table 22. Event Text for Fault MMCX Data Base Element DCS type of Drive, EVENT Treat element 342 - 343 (Continued)

IND2_15	Motor over speed	I2_TR	Motor overspeed	16	Fault	1
IND2_15	No motor over speed	I2_TR	Motor overspeed	16	Normal	2

Table 23. Event Text for Alarm MMCX Data Base Element DCS type of Drive, EVENT Treat element 344 - 345

Signal	Events	Event Treat pointer	Property text		Event Text	
			Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
IND1_00	Prevent unexpected start	I1_TR	Prevent Start	1	Alarm	1
IND1_00	No preventation of unexpected start	I1_TR	Prevent Start	1	Normal	2
IND1_01	Emergency stop	I1_TR	Emergency Stop	2	Alarm	1
IND1_01	Emergency stop	I1_TR	Emergency Stop	2	Normal	2
IND1_02	Motor 1 over temperature alarm	I1_TR	Motor 1 Temp	3	Alarm	1
IND1_02	No over temperature alarm motor 1	I1_TR	Motor 1 Temp	3	Normal	2
IND1_03	Motor 1 over load alarm	I1_TR	Motor 1 Overload	4	Alarm	1
IND1_03	No over load alarmmotor 1	I1_TR	Motor 1 Overload	4	Normal	2

Table 23. Event Text for Alarm MMCX Data Base Element DCS type of Drive, EVENT Treat element 344 - 345 (Continued)

IND1_04	Converter over temperature alarm	I1_TR	Conv. Overtemp	5	Alarm	1
IND1_04	No converter over temperature alarm	I1_TR	Conv. Overtemp	5	Normal	2
IND1_05	Register Blocked alarm	I1_TR	Register Blocked	6	Alarm	1
IND1_05	No Register Blocked alarm	I1_TR	Register Blocked	6	Normal	2
IND1_07	RAM backup alarm	I1_TR	RAM backup	8	Alarm	1
IND1_07	No RAM backup alarm	I1_TR	RAM backup	8	Normal	2
IND1_08	Motor 2 over temperature alarm	I1_TR	Motor 2 Temp	9	Alarm	1
IND1_08	No over temperature alarm motor 2	I1_TR	Motor 2 Temp	9	Normal	2
IND1_09	Motor 2 over load alarm	I1_TR	Motor 2 Overload	10	Alarm	1
IND1_09	No over load alarmmotor 2	I1_TR	Motor 2 Overload	10	Normal	2
IND1_10	Main supply under voltage	I1_TR	Mains undervoltage	11	Alarm	1
IND1_10	No main supply under voltage	I1_TR	Mains undervoltage	11	Normal	2
IND1_12	Converter Fan alarm	I1_TR	Converter Fan	13	Alarm	1
IND1_12	No Converter Fan alarm	I1_TR	Converter Fan	13	Normal	2
IND1_13	Armature current deviation alarm	I1_TR	Armature current dev.	14	Alarm	1

Table 23. Event Text for Alarm MMCX Data Base Element DCS type of Drive, EVENT Treat element 344 - 345 (Continued)

IND1_13	No armature current deviation alarm	I1_TR	Armature current dev.	14	Normal	2
IND1_15	External fan acknowledge alarm	I2_TR	External Fan	16	Alarm	1
IND1_15	No external fan acknowledge alarm	I2_TR	External Fan	16	Normal	2
IND2_00	Pannel loss alarm	I2_TR	Pannel loss	1	Alarm	1
IND2_00	No pannel loss alarm	I2_TR	Pannel loss	1	Normal	2
IND2_01	Type code changed alarm	I2_TR	Type code changed	2	Alarm	1
IND2_01	No type code changed alarm	I2_TR	Type code changed	2	Normal	2
IND2_02	Init values read, S2 alarm	I2_TR	Init values read, S2	3	Alarm	1
IND2_02	No init values read, S2 alarm	I2_TR	Init values read, S2	3	Normal	2
IND2_03	Parameter set 2 missing alarm	I2_TR	Param set 2 missing	4	Alarm	1
IND2_03	Parameter set 2 not missing	I2_TR	Param set 2 missing	4	Normal	2
IND2_04	Backup not allowed alarm	I2_TR	Backup not allowed	5	Alarm	1
IND2_04	Backup allowed	I2_TR	Backup not allowed	5	Normal	2
IND2_05	Write backup alarm	I2_TR	Write backup	6	Alarm	1
IND2_05	No write backup alarm	I2_TR	Write backup	6	Normal	2

The Event Treat data base element in the Advant Station 500 Series of Operator Station has a predefined configuration shown in Table 24.

Table 24. Predefined Event Treat for DRICONE

Property	EVENT											
	334	335	336	337	338	339	340	341	342	343	344	345
AUDIBLE	1	0	1	1	1	1	1	0	1	1	1	1
AL_PRIO	2	4	2	2	2	2	2	4	2	2	2	2
AL_TOBLK	0	1	0	0	0	0	0	1	0	0	0	0
AL_FRBLK	1	1	1	1	1	1	1	1	1	1	1	1
PERSISTB	0	1	0	0	0	0	0	1	0	0	0	0
TEXT_TOB	0	0	0	0	0	0	0	0	0	0	0	0
TEXT_FRB	0	0	0	0	0	0	0	0	0	0	0	0
TEXTCOMB	17	17	17	17	17	17	17	17	17	17	17	17

Standard Drive, DRICONS

This section describes the principles for events and alarms, the standard event texts for DRICONS main data base element and DRICONS help data base element. The section contains also a layout of the Event Treat configuration for DRICONS.

Figure 90 shows diagrammatically the relation between indications and the parameters which can be used to block event handling for certain indications and points out a required handling in the operator's station.

The possibilities of the operator to block event printouts and alarms are shown under the heading ‘Event and alarm blocking’ below.

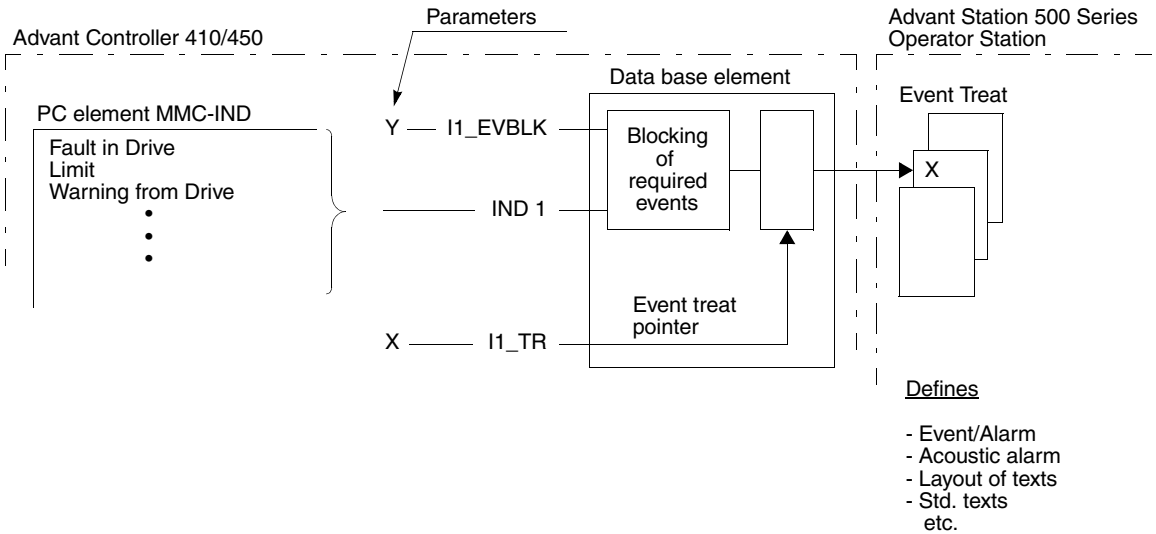


Figure 90. Event Handling Principles

The events are divided into groups. One pointer is allocated to each group which means that the individual events of the group are, in several respects, handled in the same way in Advant Station 500 Series Operator Station. For each group different handling can be selected for status changes 0-->1 and 1-->0 respectively. The texts which are written out in different lists are individual for each event. The group division - which events are associated with the pointer - is given in ‘Event

description with standard texts' below. This also specifies which event handling is obtained as default and which alternative standard function.

Table 25. Event text for main MMCX data base element, EVENT Treat element 330 - 331

Signal	Events	Event Treat pointer	Property text		Event Text	
			Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
IND1_09	Limit exceeded	I1_TR	Limit	10	Alarm	1
IND1_09	No Limit	I1_TR	Limit	10	Normal	2
IND1_10	Warning from drive	I1_TR	Warning from drive	11	Alarm	1
IND1_10	No warning	I1_TR	Warning from drive	11	Normal	2
IND1_11	Fault from drive	I1_TR	Fault in drive	12	Alarm	1
IND1_11	No fault	I1_TR	Fault in drive	12	Normal	2
IND1_12	Communication error	I1_TR	Communication error	13	Alarm	1
IND1_12	Communication error	I1_TR	Communication error	13	Normal	2
IND2_00	In service ready to switch on	I2_TR	In service	1	On	1
IND2_00	Not in service	I2_TR	In service	1	Off	2
IND2_01	Drive enabled to start	I2_TR	Enabled	2	On	1
IND2_01	Drive is not enabled to start	I2_TR	Enabled	2	Off	2
IND2_02	Running with selected reference	I2_TR	Run	3	On	1

Table 25. Event text for main MMCX data base element, EVENT Treat element 330 - 331

IND2_02	Stopped	I2_TR	Run	3	Off	2
IND2_04 (1)	No OFF 2	I2_TR	Emergency	5	Off	5
IND2_04 (1)	OFF 2	I2_TR	Emergency	5	On	6
IND2_05 (1)	No OFF 3	I2_TR	Emergency stop	6	Off	5
IND2_05 (1)	OFF 3	I2_TR	Emergency stop	6	On	6
IND2_06 (1)	Switch on is inhibited	I2_TR	Inhibit switch on	7	Yes	3
IND2_06 (1)	Switch on is not inhibited	I2_TR	Inhibit switch on	7	No	4
IND2_07	Drive runs at setpoint	I2_TR	At setpoint	8	On	1
IND2_07	Drive is not at setpoint	I2_TR	At setpoint	8	Off	2
IND2_08	C-Interlock	I2_TR	C-Interlock	9	On	1
IND2_08	C-Interlock	I2_TR	C-Interlock	9	Off	2
IND2_09	Drive runs remotely controlled	I2_TR	Remote	10	On	1
IND2_09	Drive runs locally controlled	I2_TR	Remote	10	Off	2
IND2_10	B-Interlock 1	I2_TR	B-Interlock 1	11	On	1
IND2_10	B-Interlock 1	I2_TR	B-Interlock 1	11	Off	2
IND2_11	B-Interlock 2	I2_TR	B-Interlock 2	12	On	1

Table 25. Event text for main MMCX data base element, EVENT Treat element 330 - 331

IND2_11	B-Interlock 2	I2_TR	B-Interlock 2	12	Off	2
IND2_12	B-Interlock 3	I2_TR	B-Interlock 3	13	On	1
IND2_12	B-Interlock 3	I2_TR	B-Interlock 3	13	Off	2
IND2_13	B-Interlock 4	I2_TR	B-Interlock 4	14	On	1
IND2_13	B-Interlock 4	I2_TR	B-Interlock 4	14	Off	2
IND2_14	A-Interlock	I2_TR	A-Interlock	15	On	1
IND2_14	A-Interlock	I2_TR	A-Interlock	15	Off	2
IND2_15	Override interlock	I2_TR	Override interlock	16	On	1
IND2_15	Don't override interlock	I2_TR	Override interlock	16	Off	2

(1) IND2_04 - IND2_06 are only valid if a Drive with software version 5.0 is used.

Table 26. Event text for help MMCX data base element, EVENT Treat element 332 - 333

Signal	Events	Event Treat pointer	Property text		Event Text	
			Standard text	Line No in Event Treat	Standard text	Line No in Event Treat
IND1_00	Short circuit	I1_TR	Short circuit	1	Fault	1
IND1_00	Short circuit	I1_TR	Short circuit	1	Normal	2
IND1_01	Over current	I1_TR	Over current	2	Fault	1
IND1_01	Over current	I1_TR	Over current	2	Normal	2
IND1_02	DC over voltage fault	I1_TR	DC over voltage	3	Fault	1

Table 26. Event text for help MMCX data base element, EVENT Treat element 332 - 333

IND1_02	No DC over voltage fault	I1_TR	DC over voltage	3	Normal	2
IND1_03	Power plate over temperature fault	I1_TR	Over temp pow plate	4	Fault	1
IND1_03	No power plate over temperature fault	I1_TR	Over temp pow plate	4	Normal	2
IND1_04	Earth fault	I1_TR	Earth fault	5	Fault	1
IND1_04	No earth fault	I1_TR	Earth fault	5	Normal	2
IND1_05	Motor over temperature fault	I1_TR	Over temp motor	6	Fault	1
IND1_05	No motor over temperature fault	I1_TR	Over temp motor	6	Normal	2
IND1_06	Motor over load fault	I1_TR	Over load motor	7	Fault	1
IND1_06	No motor over load fault	I1_TR	Over load motor	7	Normal	2
IND1_07	System fault	I1_TR	System fault	8	Fault	1
IND1_07	No system fault	I1_TR	System fault	8	Normal	2
IND1_08	Under load fault	I1_TR	Under load	9	Fault	1
IND1_08	No under load fault	I1_TR	Under load	9	Normal	2
IND1_09	Over speed fault	I1_TR	Over speed	10	Fault	1
IND1_09	No over speed fault	I1_TR	Over speed	10	Normal	2
IND1_10	Supply Section fault	I1_TR	Supply Section	11	Fault	1
IND1_10	No Supply Section fault	I1_TR	Supply Section	11	Normal	2

Table 26. Event text for help MMCX data base element, EVENT Treat element 332 - 333

IND1_11	Master/Follower fault	I1_TR	Master/Follower	12	Fault	1
IND1_11	No Master/Follower fault	I1_TR	Master/Follower	12	Normal	2
IND1_12	Short circuit INT1	I1_TR	Short circuit INT1	13	Fault	1
IND1_12	No Short circuit INT1	I1_TR	Short circuit INT1	13	Normal	2
IND1_13	Short circuit INT2	I1_TR	Short circuit INT2	14	Fault	1
IND1_13	No Short circuit INT2	I1_TR	Short circuit INT2	14	Normal	2
IND1_14	Short circuit INT3	I1_TR	Short circuit INT3	15	Fault	1
IND1_14	No Short circuit INT3	I1_TR	Short circuit INT3	15	Normal	2
IND1_15	Short circuit INT4	I1_TR	Short circuit INT4	16	Fault	1
IND1_15	No Short circuit INT4	I1_TR	Short circuit INT4	16	Normal	2
IND2_00	Prevent unexpected start	I2_TR	Prevent start	1	Alarm	1
IND2_00	No preventation of unexpected start	I2_TR	Prevent start	1	Normal	2
IND2_01	Emergency stop	I2_TR	Emergency stop	2	Alarm	1
IND2_01	Emergency stop	I2_TR	Emergency stop	2	Normal	2
IND2_02	Measured motor temp alarm	I2_TR	Meas. motor temp	3	Alarm	1

Table 26. Event text for help MMCX data base element, EVENT Treat element 332 - 333

IND2_02	Measured motor temp alarm	I2_TR	Meas. motor temp	3	Normal	2
IND2_03	Over temperature alarm of the thermal model	I2_TR	Motor overload	4	Alarm	1
IND2_03	No over temperature alarm of the thermal model	I2_TR	Motor overload	4	Normal	2
IND2_04	Power plate over temperature alarm	I2_TR	Temp power plate	5	Alarm	1
IND2_04	No power plate over temperature alarm	I2_TR	Temp power plate	5	Normal	2
IND2_05	Pulse encoder alarm	I2_TR	Pulse encoder	6	Alarm	1
IND2_05	No Pulse encoder alarm	I2_TR	Pulse encoder	6	Normal	2
IND2_07	Standard digital I/O alarm	I2_TR	Standard Digital I/O	8	Alarm	1
IND2_07	No Standard digital I/O alarm	I2_TR	Standard Digital I/O	8	Normal	2
IND2_08	Standard analogue I/O alarm	I2_TR	Standard Analog I/O	9	Alarm	1
IND2_08	No Standard analogue I/O alarm	I2_TR	Standard Analog I/O	9	Normal	2
IND2_09	External digital I/O alarm	I2_TR	Ext. Digital I/O	10	Alarm	1
IND2_09	No External digital I/O alarm	I2_TR	Ext. Digital I/O	10	Normal	2

Table 26. Event text for help MMCX data base element, EVENT Treat element 332 - 333

IND2_10	External analogue I/O alarm	I2_TR	Ext. Analog I/O	11	Alarm	1
IND2_10	No External analogue I/O alarm	I2_TR	Ext. Analog I/O	11	Normal	2
IND2_11	Master - Follower alarm	I2_TR	Master - Follow alarm	12	Alarm	1
IND2_11	Master - Follower alarm	I2_TR	Master - Follow alarm	12	Normal	2
IND2_14	Earth fault alarm	I2_TR	Earth fault	15	Alarm	1
IND2_14	No Earth fault alarm	I2_TR	Earth fault	15	Normal	2

The Event Treat data base element in the Advant Station 500 Series of Operator Station has a predefined configuration shown in Table 24.

Table 27. Predefined Event Treat for DRICONS

Property	EVENT			
	330	331	332	333
AUDIBLE	1	0	1	1
AL_PRIO	2	4	2	2
AL_TOBLK	0	1	0	0
AL_FRBLK	1	1	1	1
PERSISTB	0	1	0	0
TEXT_TOB	0	0	0	0
TEXT_FRB	0	0	0	0
TEXTCOMB	17	17	17	17

Appendix D Control Aspect

RTA Board Control Aspect

The RTA - Real-Time Accelerator Board - is an intelligent communication board for connection to the MasterBus 300 network on which the AC 400 Series Controllers communicate. The RTA Board Control aspect makes it possible to perform some maintenance and fault tracing work on the board.

The RTA Board aspect is located on the RTA Board object under the network object in the Control Structure. See Figure 91 for a fast location of the RTA Board control object and its aspects.

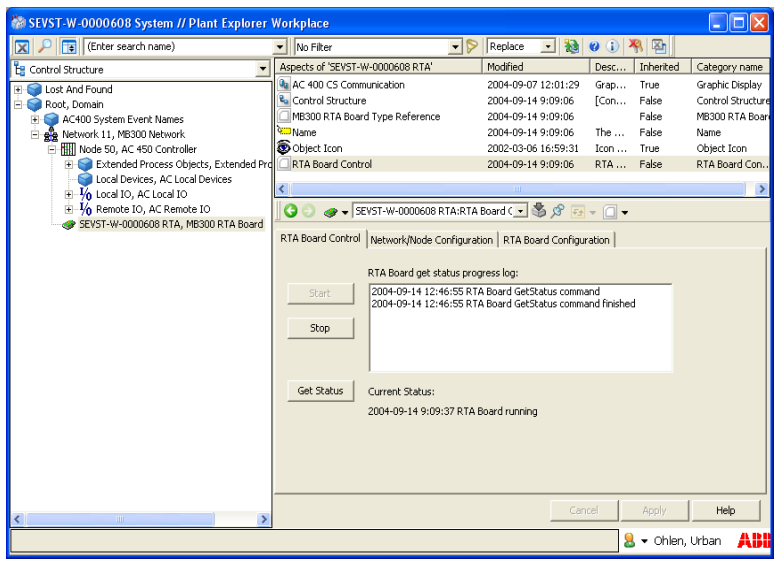


Figure 91. Locating the RTA Board Control Aspect

Select the RTA Board Control aspect using any of the possible techniques for aspect selection.

The aspect has three tabs:

- RTA Board Control, see RTA Board Control Tab on page 232.
- Network / Node Configuration, see RTA Board - Network and Node Configuration Tab on page 235.
- RTA Board Configuration - see RTA Board Configuration Tab on page 236 for details on how to use it for configuration.

RTA Board Control Tab

This tab is useful for the following purposes:

- You can request the current status of the board
- You can Start and Stop the RTA Board (for example after changing the network address or if it has halted).

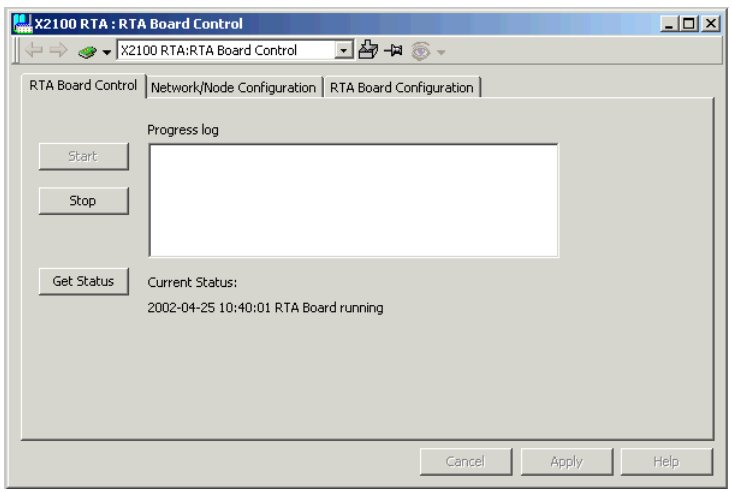


Figure 92. RTA Board Control

The aspect view contains a Progress log, the Current Status and the following buttons:

Start

Loads the RTA Board load image and the previously saved configuration and starts the RTA Board in Operational mode.

Stop

Stops the RTA Board. A warning message, see Figure 93, will pop-up and must be acknowledged before the stop command is executed.

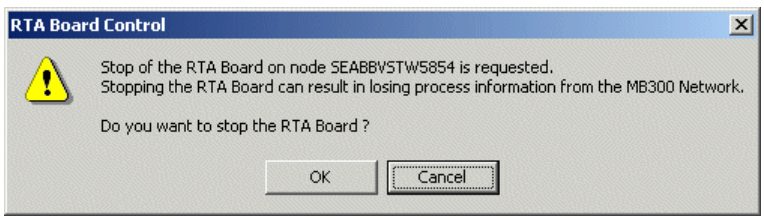


Figure 93. Confirming the Stop of the RTA Board

If you stop the RTA Board, the Connectivity Server does no longer collect any data from the AC 400 Series Controllers, and can not be used to monitor or control the process during the time the RTA Board is stopped.

Get Status

Requests the current status from the RTA Board, and presents the result in the window. (Status is also updated cyclically).

Table 28. RTA Board Status

Status text: RTA Board....	Comment
Running	Normal state. Everything OK
Stopped	Manually stopped. No run-time data is available.

Table 28. RTA Board Status (Continued)

Status text: RTA Board....	Comment
Start-up in progress	The start-up is progressing, please wait
Stop in progress	The stopping is progressing, please wait
Start failed, network address not set	Network and Node must be set before the RTA can be started
Start failed	There are a number of potential possible causes: One of the files required for boot does not exist or The path to the file is incorrect. If start fails, the cause will be displayed in the “Progress log” list in RTA Board Control. If you have tried to find the cause repeatedly, a reinstallation is advised. If reinstallation fails, the RTA Board could be out of order.
Stop failed	Bootstrapper file not found, see Start failed

RTA Board - Network and Node Configuration Tab

You use this tab to change the network and node addresses on the MB 300/AC 400 network.

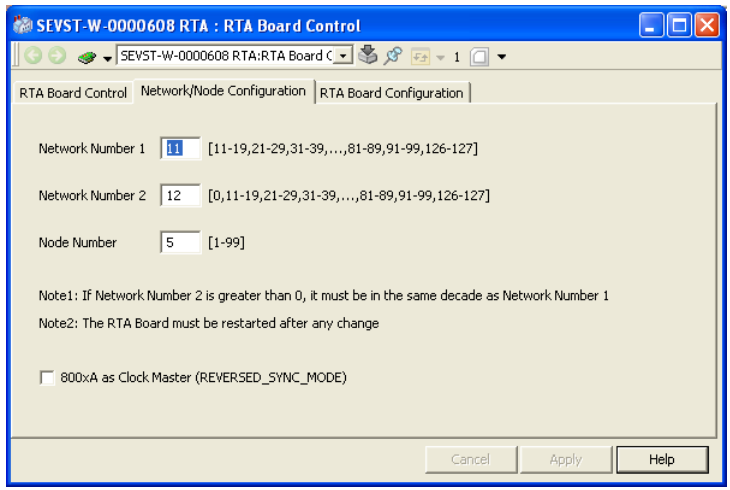


Figure 94. RTA Set Network and Node

The tab contains the following:

Network Number 1

Network address (network number) for the normal MB 300 network.
Network number range: 11-19, 21-29, 31-39,...81-89, 91-99, 126-127.

Network Number 2

Network address (network number) for the redundant MB 300 network.
Redundant network number range: 0, 11-19, 21-29, 31-39,...81-89, 91-99, 126-127.

Redundant network number 0 indicates no redundant network.

Node Number

Node address (node number) for this node. Node number range: 1-99.

You must be very careful when setting up network and node numbers. If you get two nodes with identical addresses, the entire network communication could be disturbed with unpredictable consequences. Changing network and node numbers should normally only be necessary when the control network is rearranged.

Use the buttons in the figure as follows:

- Click **Cancel** to interrupt the input.
- Click **Apply** to set the new network address (network/node number). You must restart the RTA Board to get the new network/node address to take effect (see RTA Board Control Tab on page 232).
- Click **Help** to get information about this window.

800xA as Clock Master

Select this check box if you want to have this connectivity server as Clock Master. You must restart the connectivity server to make the changes take effect.

RTA Board Configuration Tab

Use the **RTA Board Configuration** tab to start the On-line Builder program and setup the configuration for the RTA Board.The window contains the following:

RTA Board Config

Starts the On-line Builder tool that is used for configuration of the database on the RTA Board.

RTA Board Backup

Starts an application backup of the RTA Board. See Backup on page 122 for more information.

Character Conversion

In the drop-down menu you select which language the RTA Board shall use. For more information, refer to *System 800xA, NLS Localization (2PAA101940*).*

Control Connection Aspect (CCA)

There is a CCA for all process and system objects. The CCA:s can be useful to trace problems specific for a certain object or for the presentation of an object. Configuration of this aspect is preferable done on the object type, but can also be done locally for each instance in the Control Structure. The CCA has a number of tabs, here is a short description of some of the tabs and how you can use them in your trouble shooting. For more detailed information about the tabs see *System 800xA, Configuration (3BDS011222*).*

CCA Property View Tab

The CCA Property View tab, see Figure 95, gives you a list of all the properties, their data format and how they can be accessed. If you mark the **Subscribe for live data** check box, you will also get the current value of all properties updated cyclically.

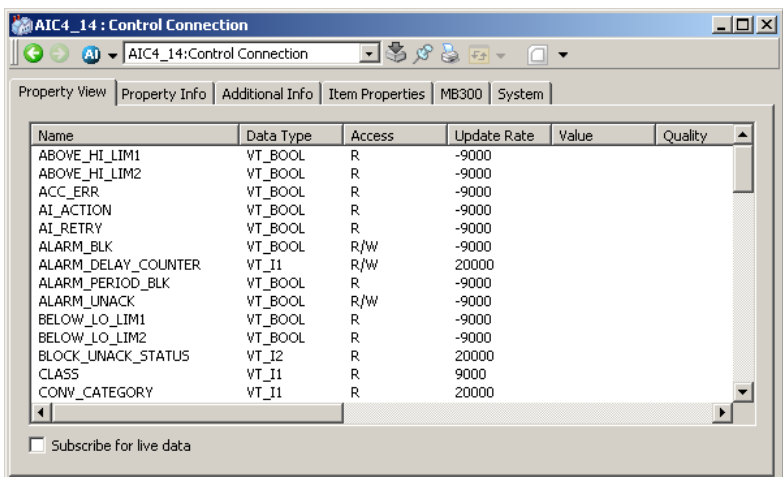


Figure 95. CCA Property View Tab

Use this tab to verify the property values of an object, how they are configured and the current property values to compare with other presentations of it. If you suspect an error in a node, an I/O board or a transducer, you can use the CCA:s to narrow in on the problem by verifying which objects that are available over the network and which are not - and what their status properties tell you.

The CCA Property View tab gives you a complete list of properties, their names, data type, access.

A "**-**" before the update rate value enables "on event" updating, of the values in between the cyclic updates according to AC 400 Event handling. If you locally change the update rate in a graphic display, " **-** " must be included if you want the "on event" updating to be activated. Acceptable numbers are: -1000, -3000 and -9000.

CCA Property Info Tab

The Property Info tab presents all details about the selected property. An example of a Property Info tab is found in Figure 96.

The Property Info tab includes the possibility to set the definitions for the Property values. To define property value handling and to set these values could severely affect the control of the industrial process, including loosing control of the process. This functionality is only included for testing and fault tracing, and should not be used during normal operation. Do not change any Property values unless you are fully aware of all the consequences.

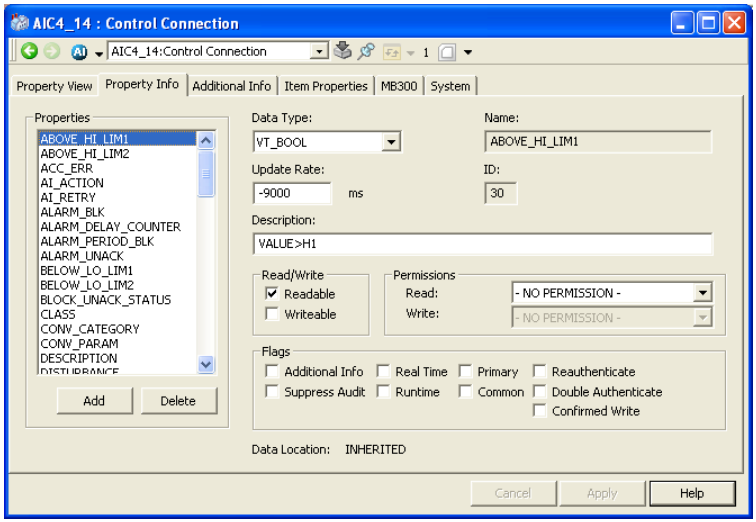


Figure 96. CCA Property Info Tab

The following Property values are presented in the dialog box:

- **Type** - Type of value (boolean, integer, float, etc.).
- **Name** - The name of the property.
- **Update Rate** - The default update rate for this property (in ms).
- **Description** - A user friendly name of the property. This name can be presented in Audit Trail lists.
- **R/W** - Flag that indicates if the property is Readable and/or Writable.
- **Permissions** - Defines the Read and Write permissions for the property.
- **Flags** - Indicates how the property is updated.

CCA - Additional Info Tab

The Additional Info tab presents details on property specific information. The example in Figure 96 presents information about an analog value.

The Additional Info tab includes the possibility to set definitions for the values. To define value handling and to set these values could severely affect the control of the industrial process, including loosing control of the process.
This functionality is only included for testing fault tracing, and should not be used during normal operation.
Do not change any Additional Info values unless you are fully aware of all the consequences.

The following Additional info values are presented and could be changed:

- **Normal Maximum** - Maximum range value (for an analog value).
- **Normal Minimum** - Minimum range value (for an analog value).
- **Engineering Unit** - Engineering unit for this property.
- **No of Decimals** - This value should be presented with this number of decimals.
- **Presentation Mode** - Defines if this value should be presented between the discrete points in trend curves: Stepped (value is constant between points) or Interpolated (value is linear between points).

The Additional Info values are changed when you click on the **Set** button.

The Presentation Mode makes it possible for you to select the default presentation mode in Trace presentations: Stepped or Interpolated, see *System 800xA, Operations, Operator Workplace Configuration (3BSE030322*)* for more information.

CCA MasterBus 300 Tab

The **MB 300** tab presents detailed address information about the MB 300 object.

The following information is available for each object:

- The MasterBus 300 address: Network number, Node number, Logical file number, Logical record number, Reference type.
- The ID property number of each attribute.

Values in this tab are updated when upload is performed, and should not be altered manually.

Appendix E Cyclic OPC Properties

OPC Object Type Properties

The following table lists process objects and the properties that supports cyclic subscriptions. The cyclic subscription times are 1, 3 and 9 seconds.

See Section 7, Data Transfer via OPC DA for more information.

Table 29. Properties Supporting Cyclical Subscriptions via OPC

Object Type	Property	Subproperties
AI	VALUE	-
	SIGNAL_STATUS	IMPLEMENTED, ERROR, UPDATED, UPD_BLK, MAN_ENTRY, SELECTED, NORMAL_OBJ_TREAT, H2_R_FCL, ABOVE_HI_LIM2, ABOVE_HI_LIM1, BELOW_LO_LIM1, BELOW_LO_LIM2, ALARM_UNACK, H1_R_FCL, DISTURBANCE, OVERFLOW, ALARM_BLK, ALARM_PERIOD_BLK, PRINT_BLK, L1_R_FCL, LINKED, RELINK, NOERR_AT_OVF, TESTED, ACC_ERR, OLD_LOCK, L2_R_FCL, ER_R_FCL, RP_F_BLK, AI_ACTION, AI_RETRY, ERR_CTRL
	NO_OF_DEC	-

Table 29. Properties Supporting Cyclical Subscriptions via OPC (Continued)

Object Type	Property	Subproperties
AO	VALUE	-
	SIGNAL_STATUS	IMPLEMENTED, ERROR, AO_SPARE_2, OUTP_BLK, AO_SPARE_4, SELECTED, NORMAL_OBJ_TREAT, MAN_MODE, ON_MAX_LIM, AO_SPARE_9, AO_SPARE_10, ON_MIN_LIM, ALARM_UNACK, AO_SPARE_13, AO_SPARE_14, AO_SPARE_15, ALARM_BLK, ALARM_PERIOD_BLK, PRINT_BLK, OUTP_RESTART, OLD_LOCK, USE_MAX_LIM, USE_MIN_LIM, TESTED, AO_SPARE_24, USER_DEF_1, LINKED, USER_DEF_2, USER_DEF_3, USER_DEF_4, USER_DEF_5, OSP_CTRL
	NO_OF_DEC	-
DI	SIGNAL_STATUS	IMPLEMENTED, ERROR, UPDATED, UPD_BLK, MAN_ENTRY, SELECTED, NORMAL_OBJ_TREAT, ERR_CTRL, VALUE, SEC_VALUE, NORM_POSN, SEC_NORM_POSN, ALARM_UNACK, REPEAT_FAIL_BLK, DISTURBANCE, CALC_VALUE, ALARM_BLK, ALARM_PERIOD_BLK, PRINT_BLK, REPEAT_FAIL_CTRL, OLD_LOCK, INVERTED, DUAL_IND, TESTED, DI_SPARE_24, V9_CONTROL, LINKED, DI_RETRY, DIC_IND_RED, DI_ACTION, DIC_IND_YELLOW, DIC_IND_GREEN

Table 29. Properties Supporting Cyclical Subscriptions via OPC (Continued)

Object Type	Property	Subproperties
DO	SIGNAL_STATUS	IMPLEMENTED, ERROR, DO_SPARE_2, OUTP_BLK, DO_SPARE_4, SELECTED, NORMAL_OBJ_TREAT, MAN_MODE, VALUE, DO_SPARE_9, START_VALUE, DO_SPARE_11, ALARM_UNACK, DO_SPARE_13, DO_SPARE_14, USER_DEF_1, ALARM_BLK, ALARM_PERIOD_BLK, PRINT_BLK, OUTP_RESTART, OLD_LOCK, INVERTED, OSP_VAL, TESTED, ORDER_TO, ORDER_FROM, LINKED, USER_DEF_2, USER_DEF_3, USER_DEF_4, USER_DEF_5, OSP_CTRL
DAT	VALID	-
	VAL_TYPE	-
	VALUE	B0_VAL, B1_VAL, B2_VAL, B3_VAL, B4_VAL, B5_VAL, B6_VAL, B7_VAL, B8_VAL, B9_VAL, B10_VAL, B11_VAL, B12_VAL, B13_VAL, B14_VAL, B15_VAL, B16_VAL, B17_VAL, B18_VAL, B19_VAL, B20_VAL, B21_VAL, B22_VAL, B23_VAL, B24_VAL, B25_VAL, B26_VAL, B27_VAL, B28_VAL, B29_VAL, B30_VAL, B31_VAL
	I_VAL	-
	IL_VAL	-
	R_VAL	-
DRICONE, DRICONS, GROUP, MOTCON, MOTCONI, VALVECON	STATUS	IMPLEMENTED, ALARM_BLK, ALARM_PERIOD_BLK, PRINT_BLK, IND1_DIST, IND2_DIST, SELECTED, AU_IND, BOOL_A, BOOL_B, BOOL_C, BOOL_D, BOOL_E, BOOL_F, BOOL_G, BOOL_H

Table 29. Properties Supporting Cyclical Subscriptions via OPC (Continued)

Object Type	Property	Subproperties
DRICONE, DRICONS, GROUP, MOTCON, MOTCONI, VALVECON (cont.)	ALARM_UNACK	AU_IND1_00, AU_IND1_01, AU_IND1_02, AU_IND1_03, AU_IND1_04, AU_IND1_05, AU_IND1_06, AU_IND1_07, AU_IND1_08, AU_IND1_09, AU_IND1_10, AU_IND1_11, AU_IND1_12, AU_IND1_13, AU_IND1_14, AU_IND1_15, AU_IND2_00, AU_IND2_01, AU_IND2_02, AU_IND2_03, AU_IND2_04, AU_IND2_05, AU_IND2_06, AU_IND2_07, AU_IND2_08, AU_IND2_09, AU_IND2_10, AU_IND2_11, AU_IND2_12, AU_IND2_13, AU_IND2_14, AU_IND2_15
	IND1	IND1_00, IND1_01, IND1_02, IND1_03, IND1_04, IND1_05, IND1_06, IND1_07, IND1_08, IND1_09, IND1_10, IND1_11, IND1_12, IND1_13, IND1_14, IND1_15
	IND2	IND2_00, IND2_01, IND2_02, IND2_03, IND2_04, IND2_05, IND2_06, IND2_07, IND2_08, IND2_09, IND2_10, IND2_11, IND2_12, IND2_13, IND2_14, IND2_15
	REAL_RES	-
	INTL_RES	-
	ACT_PRES_TEXT	-
	INTW_A	INTWA_00, INTWA_01, INTWA_02, INTWA_03, INTWA_04, INTWA_05, INTWA_06, INTWA_07, INTWA_08, INTWA_09, INTWA_10, INTWA_11, INTWA_12, INTWA_13, INTWA_14, INTWA_15
	INTW_B	INTWB_00, INTWB_01, INTWB_02, INTWB_03, INTWB_04, INTWB_05, INTWB_06, INTWB_07, INTWB_08, INTWB_09, INTWB_10, INTWB_11, INTWB_12, INTWB_13, INTWB_14, INTWB_15
	REAL_A	

Table 29. Properties Supporting Cyclical Subscriptions via OPC (Continued)

Object Type	Property	Subproperties
DRICONE, DRICONS, GROUP, MOTCON, MOTCONI, VALVECON (cont.)	REAL_B	
	REAL_C	
	REAL_D	
	REAL_E	
GENBIN, GENCON, GENUSD	STATUS_1	IMPLEMENTED, ALARM_BLK, ALARM_PERIOD_BLK, PRINT_BLK, REPEAT_FAIL_BLK, EVENT_BLK, SELECTED, MORD_EV_BLK, ALARM_UNACK, AU_ST_00, AU_ST_01, AU_ST_02, AU_ST_03, AU_ST_04, AU_ST_05, AU_ST_06, AU_ST_07, PC_STATUS, PC_ST_00, PC_ST_01, PC_ST_02, PC_ST_03, PC_ST_04, PC_ST_05, PC_ST_06, PC_ST_07, PC_ST_08, PC_ST_09, PC_ST_10, PC_ST_11, PC_ST_12, PC_ST_13, PC_ST_14, PC_ST_15
	STATUS_2	PC_ST_16, PC_ST_17, PC_ST_18, PC_ST_19, PC_ST_20, PC_ST_21, PC_ST_22, PC_ST_23, PC_ST_24, PC_ST_25, PC_ST_26, PC_ST_27, PC_ST_28, PC_ST_29, PC_ST_30, PC_ST_31
	MV	-
	MV_INTL	-
	SP	-
	OP	-
	H2	-
	L2	-

Table 29. Properties Supporting Cyclical Subscriptions via OPC (Continued)

Object Type	Property	Subproperties
MANSTN	STATUS	-
	STATUS_1	IMPLEMENTED, F1_ACTION, RUNNING, ACTUATOR, MVDIR, SELECTED, F2_ACTION, TESTED, PANEL_CTRL, SPARE_STATUS_09, SPARE_STATUS_10, OUT_EQ_LL (OUT=LL), OUT_EQ_HL (OUT=HL), ALARM_UNACK, DIST, ALARM_F1_BLK, ALARM_F1_PERIOD_BLK, ALARM_F2_BLK, ALARM_F2_PERIOD_BLK
	STATUS_2	MMI_MAN, MMI_E1, TS_MV_BELOW_L1 (TS_MV<L1), TS_MV_ABOVE_H1 (TS_MV>H1), TS_MV_BELOW_L2 (TS_MV<L2), TS_MV_ABOVE_H2 (TS_MV>H2), TS_AI_ERROR, TS_AO_ERROR
	STATUS_3	REMOTE, CENTRAL, LOCAL, SERVICE_UNIT, MAN, E1
	STATUS_4	AU_MV_BELOW_L1 (AU_MV<L1), AU_MV_ABOVE_H1 (AU_MV>H1), PRINT_F1_BLK, AU_MV_BELOW_L2 (AU_MV<L2), AU_MV_ABOVE_H2 (AU_MV>H2), PRINT_F2_BLK
	MV	-
	OUTREF	-
	OUT	-
	TS	-
	POUT	-
	MNO_OF_DEC	-
	PONO_OF_DEC	-
	PRES1	-

Table 29. Properties Supporting Cyclical Subscriptions via OPC (Continued)

Object Type	Property	Subproperties
PIDCON	STATUS_1	IMPLEMENTED, F1_ACTION, RUNNING, ACTUATOR, MVDIR, SELECTED, F2_ACTION, TESTED, PANEL_CTRL, ON_OFF_CTRL, F4_ACTION, OUT_EQ_LL (OUT=LL), OUT_EQ_HL (OUT=HL), SP_EQ_LL (SP=LL), SP_EQ_HL (SP=HL), ALARM_UNACK, REPEAT_FAIL_BLK, DIST, ALARM_F1_BLK, ALARM_F1_PERIOD_BLK, ALARM_F2_BLK, ALARM_F2_PERIOD_BLK
	STATUS_2	MMI_MANF, MMI_MAN, MMI_AUTO, MMI_INT_BLK, MMI_DER_BLK, MMI_E1, MMI_E2, MMI_E3, TS_MV_BELOW_L1 (TS_MV<L1), TS_MV_ABOVE_H1 (TS_MV>H1), TS_DEV_BELOW_L (TS_DEV<L), TS_DEV_ABOVE_H (TS_DEV>H), TS_MV_BELOW_L2 (TS_MV<L2), TS_MV_ABOVE_H2 (TS_MV>H2), TS_LOCAL_FL, TS_MAN_FL, TS_AUTO_FL, TS_E1_FL, TS_E2_FL, TS_E3_FL, TS_AI_ERROR, TS_AO_ERROR, TS_DCM_ERROR, TS_PC_BLK, TS_SERVUC, TS_HW_ERR
	STATUS_3	REMOTE, CENTRAL, LOCAL, SERVICE_UNIT, BAL, MAN, AUTO, E1, E2, E3, LOCAL_OUT, CLAMP_OUT, EXT_OUT_LIMIT, EXT_GAIN_ENBL, EXT_TI_ENBL, EXT_TD_ENBL, EXT_TF_ENBL, EXT_INT_BLK, EXT_DER_BLK, SPARE_PARAM5_15
	STATUS_4	AU_MV_BELOW_L1 (AU_MV<L1), AU_MV_ABOVE_H1 (AU_MV>H1), AU_DEV_BELOW_L (AU_DEV<L), AU_DEV_ABOVE_H (AU_DEV>H), PRINT_F1_BLK, REPEAT_F1_FAIL_CTRL, AU_MV_BELOW_L2 (AU_MV<L2), AU_MV_ABOVE_H2 (AU_MV>H2), PRINT_F2_BLK, REPEAT_F2_FAIL_CTRL

Table 29. Properties Supporting Cyclical Subscriptions via OPC (Continued)

Object Type	Property	Subproperties
PIDCON (cont.)	MV	-
	AUTOSP	-
	WSP	-
	DEVIATION	-
	OUT	-
	TS	-
	MAN_OUT	-
	SPARE_PARAM6_01	-
	POUT	-
	MNO_OF_DEC	-
	EXT_OUT_LL	-
	EXT_OUT_HL	-
	EXT_GAIN	-
	EXT_TI	-
	EXT_TD	-
	EXT _TF	-
	PRES1	-
	PRES2	-
PIDCONA	STATUS_1	IMPLEMENTED, ACTUATOR, MVDIR, SELECTED, MANPREF, SHOW_ACT, SHOW_SCHED, AUTOSP_TRACKING, STATUS_SPARE01, OUT_EQ_LL, OUT_EQ_HL, SP_EQ_LL, SP_EQ_HL, ALARM_UNACK, DIST

Table 29. Properties Supporting Cyclical Subscriptions via OPC (Continued)

Object Type	Property	Subproperties
PIDCONA (cont.)	STATUS_2	TS_MV_BELOW_L1, TS_MV_ABOVE_H1, TS_DEV_BELOW_L, TS_DEV_ABOVE_H, TS_MV_BELOW_L2, TS_MV_ABOVE_H2, TS_LOCAL_FL, TS_MAN_FL, TS_AUTO_FL, TS_E1_FL, TS_E2_FL, TS_E3_FL, TS_AI_ERROR, TS_AT_PC_ABORTED, TS_AT_FAILED, TS_AT_ALERT, TS_INV_TSAMP, TS_ADAP_FAIL, TS_AT_SPARE01, TS_AT_SPARE02, TS_AT_SPARE03
	STATUS_3	MMI_MANF, MMI_MAN, MMI_AUTO, PARAM11_SPARE01, PARAM11_SPARE02, MMI_E1, MMI_E2, MMI_E3, BAL, MAN, AUTO, E1, E2, E3, LOCAL_OUT, CLAMP_OUT, EXT_OUT_LIMIT, BADSN, OUTSIDE_START_ZONE, AT_TIMEOUT_FLAG, VERIFY_UNCERTAIN, AT_INDICATOR, AD_INDICATOR, ATENABLE, GSENABLE
	STATUS_4	AU_MV_BELOW_L1, AU_MV_ABOVE_H1, AU_DEV_BELOW_L, AU_DEV_ABOVE_H, ALARM_F1_BLK, ALARM_F1_PERIOD_BLK, PRINT_F1_BLK, AU_MV_BELOW_L2, AU_MV_ABOVE_H2, ALARM_F2_BLK, ALARM_F2_PERIOD_BLK, PRINT_F2_BLK
	STATUS_5	START_TUNE, CONT_TUNE, SAVE_REQUEST, RESTORE_REQUEST, ADAPENBL, FREQ_UNLIMITED, VERIFY_REQUEST, RETRIEVE, RETRIEVEALLOWED, PARAM17_SPARE01, PARAM17_SPARE02, PARAM17_SPARE03, AU_AT_PC_ABORTED, AU_AT_FAILED, AU_AT_ALERT, AU_INV_TSAMP, AU_ADAP_FAIL, AU_AT_SPARE01, AU_AT_SPARE02, AU_AT_SPARE03
	PARAM13	-

Table 29. Properties Supporting Cyclical Subscriptions via OPC (Continued)

Object Type	Property	Subproperties
PIDCONA (cont.)	MV_NONFILTERED	-
	AUTOSP	-
	WSP	-
	DEVIATION	-
	OUT	-
	MV_AAFILTERED	-
	EXTREF1	-
	EXTREF2	-
	EXTREF3	-
	EXT_OUT_LL	-
	EXT_OUT_HL	-
	ACTPOS	-
	AT_PHASE	-
	AT_COUNTER	-
	AT_START_ZONE	-
	CURRENT_ZONE	-
	SAVED_PAR_MAN_AD	-
	CONTR_PAR_MAN_AD	-
	Z1_MAN_AD_OBT_PAR	-
	Z2_MAN_AD_OBT_PAR	-
	Z3_MAN_AD_OBT_PAR	-
	Z4_MAN_AD_OBT_PAR	-
	Z5_MAN_AD_OBT_PAR	-

Table 29. Properties Supporting Cyclical Subscriptions via OPC (Continued)

Object Type	Property	Subproperties
PIDCONA (cont.)	SAVED_GAIN	-
	SAVED_TI	-
	SAVED_TD	-
	SAVED_BETA	-
	SAVED_TS	-
	SAVED_DOM_FREQ	-
	SCHEDIN	-
	ZLIM12	-
	ZLIM23	-
	ZLIM34	-
	ZLIM45	-
	TRIM_REP	-
	PARAM16_SPARE01	-
	MNO_OF_DEC,	-
	ONO_OF_DEC	-
RATIOSTN	STATUS	-
	STATUS_1	IMPLEMENTED, F1_ACTION, RUNNING, SPARE_STATUS_03, SPARE_STATUS_04, SELECTED, F2_ACTION, TESTED, PANEL_CTRL, SPARE_STATUS_09, SPARE_STATUS_10, OUT_EQ_LL (OUT=LL), OUT_EQ_HL (OUT=HL), RATIO_EQ_LL (RATIO=LL), RATIO_EQ_HL (RATIO=HL), ALARM_UNACK, REPEAT_FAIL_BLK, DIST, ALARM_F1_BLK, ALARM_F1_PERIOD_BLK, ALARM_F2_BLK, ALARM_F2_PERIOD_BLK

Table 29. Properties Supporting Cyclical Subscriptions via OPC (Continued)

Object Type	Property	Subproperties
RATIOSTN (cont.)	STATUS_2	MMI_MAN, MMI_AUTO, MMI_E1, TS_MV_BELOW_L1 (TS_MV<L1), TS_MV_ABOVE_H1 (TS_MV>H1), TS_MV_BELOW_L2 (TS_MV<L2), TS_MV_ABOVE_H2 (TS_MV>H2), TS_AI_ERROR, SPARE_TS_05, TS_DCM_ERROR, TS_PC_BLK, TS_SERVUC, TS_HW_ERR
	STATUS_3	TRACK_EQ_1 (TRACK=1), MAN, AUTO, E1, EXT_BIAS_ENBL, EXT_OUT_LIMIT, SPARE_PARAM5_06
	STATUS_4	AU_MV_BELOW_L1 (AU_MV<L1), AU_MV_ABOVE_H1 (AU_MV>H1), PRINT_F1_BLK, AU_MV_BELOW_L2 (AU_MV<L2), AU_MV_ABOVE_H2 (AU_MV>H2), PRINT_F2_BLK
	MMI_RATIOREF	-
	MMI_MAN_OUT	-
	MMI_BIAS	-
	MMI_OUT_LL	-
	MMI_OUT_HL	-
	SPARE_PARAM2_01	-
	MV	-
	RATIOREF	-
	WRATIO	-
	OUT	-
	MAN_OUT	-
	TS	-
	POUT	-

Table 29. Properties Supporting Cyclical Subscriptions via OPC (Continued)

Object Type	Property	Subproperties
RATIOSTN (cont.)	MNO_OF_DEC	-
	RNO_OF_DEC	-
	PONO_OF_DEC	-
	EXT_OUT_LL	-
	EXT_OUT_HL	-

Table 29. Properties Supporting Cyclical Subscriptions via OPC (Continued)

Object Type	Property	Subproperties
SEQ	STATUS	-
	STATUS_1	IMPLEMENTED, SEQ_RFS, SEQ_COMPLETE, BLOCKED, SPARE_04, SELECTED, SPARE_06, TESTED, AUTOM, MANM, HOLDM, UNCONDM, RUN, END, NEXT, IND_SPARE_07, DI1_ALARM_BLK, DI1_ALARM_PERIOD_BLK, DI1_PRINT_BLK, DI1_REPEAT_FAIL_CTRL, DI1_SECOND_FAIL_BLK, DI1_SPARE_05, DI1_SPARE_06, DI1_SPARE_07, DI2_SPARE_00, DI2_SPARE_01, DI2_SPARE_02, DI2_SPARE_03, DI2_ALARM_UNACK, DI2_REPEAT_FAIL_BLK, DI2_DIST, DI2_SPARE_07
	STATUS_2	REMOTE, CENTRAL, LOCAL, SERVICE_UNIT, TS_SPARE_00, TS_POSN_F, TS_SERVUC, TS_SEQAL, TS_STEPAL, AU_SPARE_00, AU_POSN_F, AU_SERVUC, AU_SEQAL, AU_STEPAL, TF_ALARM_BLK, TF_ALARM_PERIOD_BLK, TF_PRINT_BLK, TF_REPEAT_FAIL_CTRL, TF_ALARM_PERIOD_BL2, TF_SPARE_05, TF_SPARE_06, TF_SPARE_07
	STATUS_3	PT_ALARM_BLK, PT_SPARE_01, PT_PRINT_BLK, SI_ALARM_BLK, SI_SPARE_01, SI_PRINT_BLK,
	POSN	-
	SEQTD	-
	SEQTE	-
	STEPTD	-
	STEPTE	-
	JPOSN_OUTP	-
	ACT_TURN	-

Table 29. Properties Supporting Cyclical Subscriptions via OPC (Continued)

Object Type	Property	Subproperties
SEQ (cont.)	INTERV_TIME_EL	-
	ALARM_UNACK	-
	STEPSTAT	CONDSTA1, CONDSTA2, CONDSTA3, CONDSTA4, CONDSTA5, CONDSTA6, CONDSTA7, CONDSTA8, ACTSTA1, ACTSTA2, ACTSTA3, ACTSTA4, ALLACT, BLANKINF, LOAD_DB, NEXTCOND, JCSTA1, JCSTA2, JCSTA3, JCSTA4, INFVAL, NEXTSTEP
TEXT	INT_LONG	-
	REAL	-
	DISPMAX	-
	DISPMIN	-
	NO_OF_DEC	-
	STATUS	VALID, SELECTED, MAN, BOOLEAN, COLOUR1, COLOUR2, BLANKT, BLANKB, BLANKR, BLANKIL, B1_VAL, B2_VAL, B3_VAL, B4_VAL, SPARE_B1, SPARE_B2
	TEXT	-

Appendix F Object Type Names

The Object Types have different names in the 800xA Object Type Structure and in the Advant Master controller. The table below contains the Object Type names as displayed in the Object Type Structure and the corresponding Object Type names (Functional Unit names) as named in the 800xA for Advant Master controller.

800xA Object Type Structure Name	Advant Master Controller Name (Functional Units)
MB300 Adaptive PID Ctrl	PIDCONA
MB300 Binary Control	GENBIN
MB300 DatB	DATB
MB300 DatI	DATI
MB300 DatIL	DATIL
MB300 DatR	DATR
MB300 Engineered drive	DRICONE
MB300 Genusd	GENUSD
MB300 Group Alarm	GroupAlarm
MB300 Group	GROUP
MB300 Manual Station	MANSTN
MB300 MotConI	MOTCON_I
MB300 Motor	MOTCON
MB300 PI Controller	GENCON
MB300 PID Controller	PIDCON
MB300 Ratio Station	RATIOSTN
MB300 Sequence Control	SEQUENCE

800xA Object Type Structure Name	Advant Master Controller Name (Functional Units)
MB300 Standard drive	DRICONS
MB300 Text data	TEXT_DATA
MB300 Valve	VALVECON
MB300 AI	AI
MB300 AO	AO
MB300 DI	DI
MB300 DO	DO

Appendix G Hardware Installation

RTA Unit PU410

The RTA Unit PU410 is a standalone hardware unit which supports both single and dual point to point connection with the connectivity server node through Ethernet. Single or redundant connection with MasterBus 300 is enabled through RJ-45 connectors.

Once the installation has been completed, the term RTA Board is used in Plant Explorer, dialogs and system status for the PU410 RTA hardware unit.

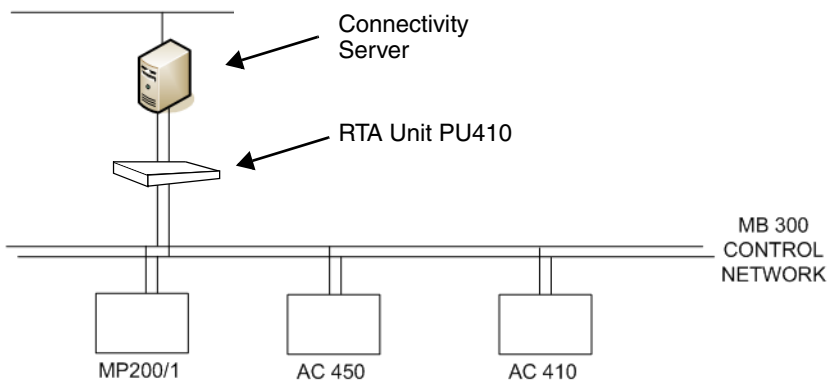


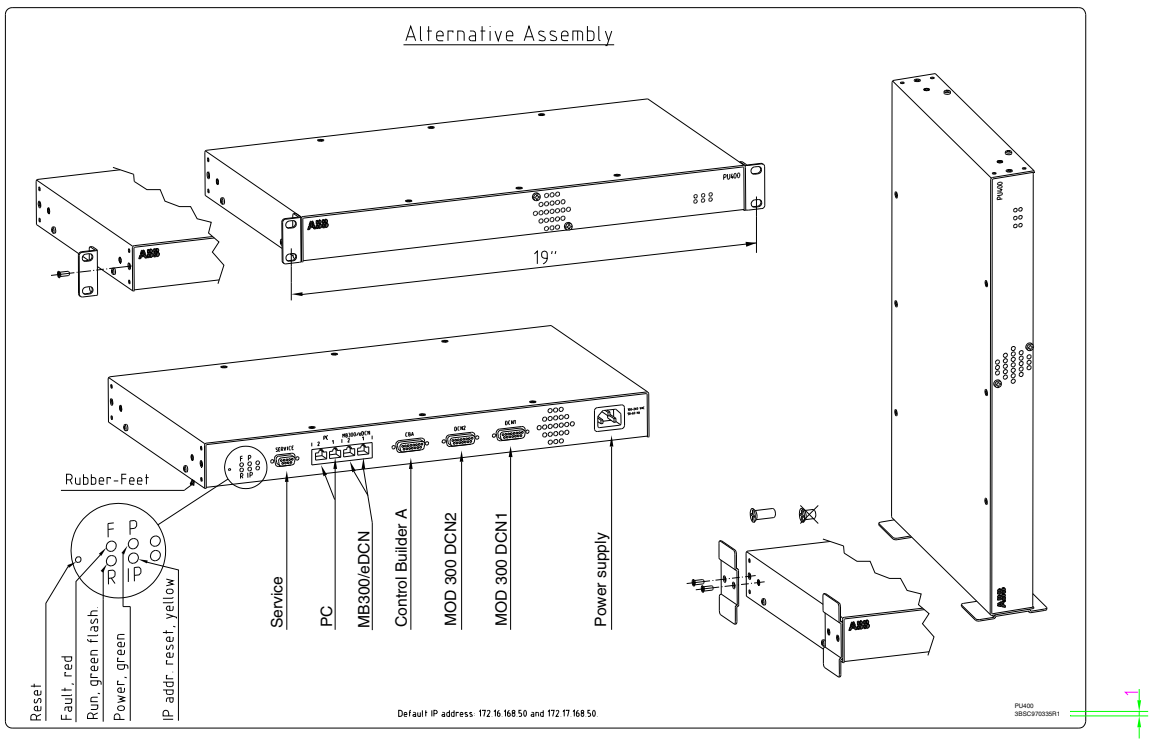
Figure 97. Overview - Typical Installation

PU410 Specification

– Connectors:
Two shielded RJ-45 connectors for 100Mbit/s full duplex communication with the PC (Connectivity Server), PC 1 and 2 ports.

- Two shielded RJ-45 connectors for 10Mbit/s half duplex communication with MasterBus 300/300E or eDCN control network, MB300/eDCN 1 and 2 ports.
- Two 15-pos Dsub socket connectors for DCN communication, DCN1 and DCN2 ports. (Not used for PU410)
- One 15-pos Dsub socket connector for RS232 and RS422 communication with Control Builder A and CBA port. (Not used in 800xA for Advant Master connectivity server)
- One 9-pos Dsub socket connector for RS232 communication for debug and upgrading of firmware, SERVICE port.
- Power:
  100-230Vac 50-60Hz
- Power Consumption: Typical 12W without I/O connected, maximum 25W
- Size:
  HxWxD (mm): 45 x 443 x 243
- Net weight: 3.2 kg

The informative label is attached underneath the unit.



In case of rack assembly, place the PU410 Unit so that the connectors and the reset button are within reach.

Before returning the PU410 Unit for replacement or repair, always remove accessories such as brackets and cables.

Figure 98 shows the rear and front view of PU410.

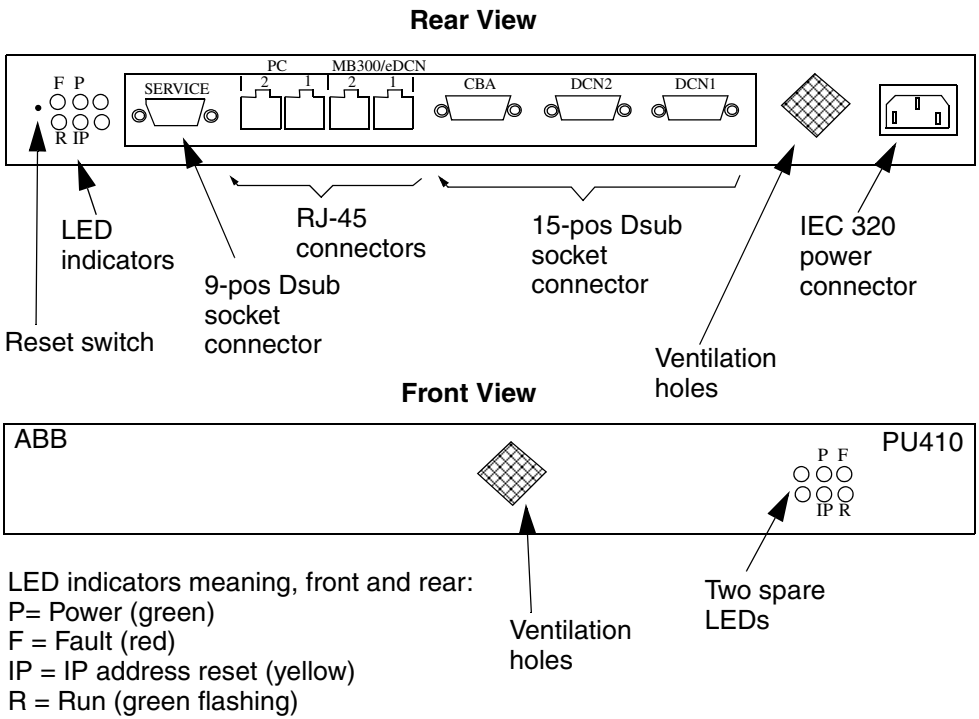


Figure 98. The PU410 External RTA Unit, Front and Rear View

Since the PU410 unit includes a supervised fan, it is recommended to check the fan and vacuum clean the unit from time to time.

If a fan problem is detected, the red fault LED will be illuminated and the unit will shut down. The 800xA for Advant Master connectivity server will try to restart the unit. The red fault LED will still be illuminated until the unit is reset, even if the fan problem disappears.

The red fault LED can illuminate for other problems as well.

The red fault LED will also illuminate for a few seconds at PU410 start-up.

Mechanical Installation

The PU410K01 kit includes brackets for 19” rack mount and for floor or desktop mounting.

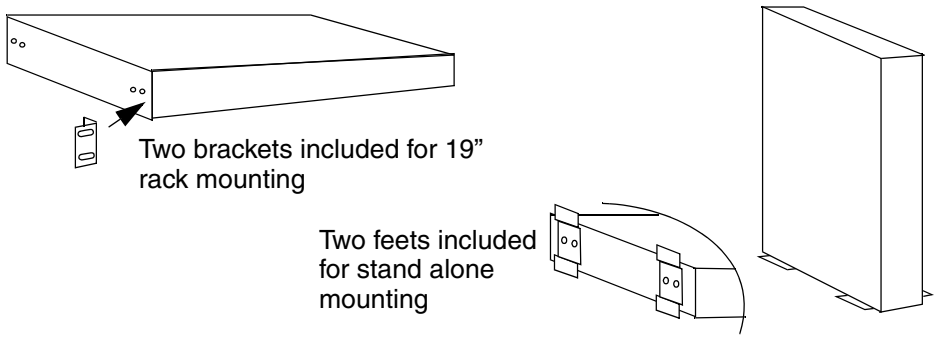


Figure 99. Brackets for PU410 Installation in 19” Rack or Stand Alone

Connection To Connectivity Server

The PU410 unit has two ports called PC 1 and 2 for connection to the connectivity server. Only one port is necessary for the communication but if redundancy is wanted both ports can be used.

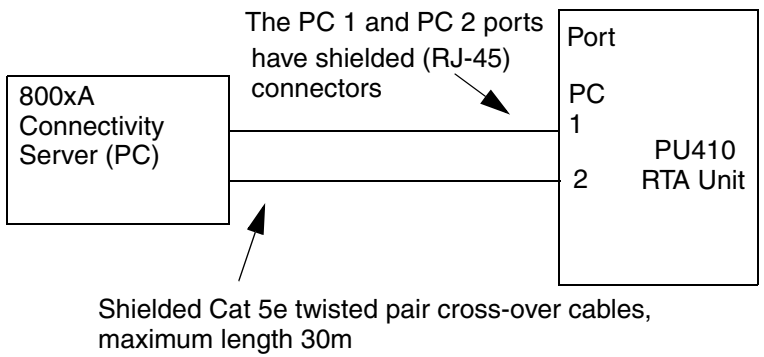


Figure 100. Connecting PU410 to Connectivity Server

- Always use shielded Cat 5e (or better) twisted pair cross-over cables to reduce electric noise disturbances. The maximum cable length is 30m but keep cable length as short as possible.
- The default IP addresses of PU410 are for the PC 1 port 172.16.168.50 and for the PC 2 port 172.17.168.50 and can be used in multiple connectivity servers as they are ‘local’ IP addresses.
- The IP address for the Network Interface Card (NIC) in the connectivity server shall be 172.16.168.y with subnet mask 255.255.252.0, where *y* is any number except the number used by PU410. When using dual cables, the IP address of the additional NIC shall be 172.17.168.y with subnet mask 255.255.252.0. Redundant combined aspect and connectivity servers must have different value of *y*.

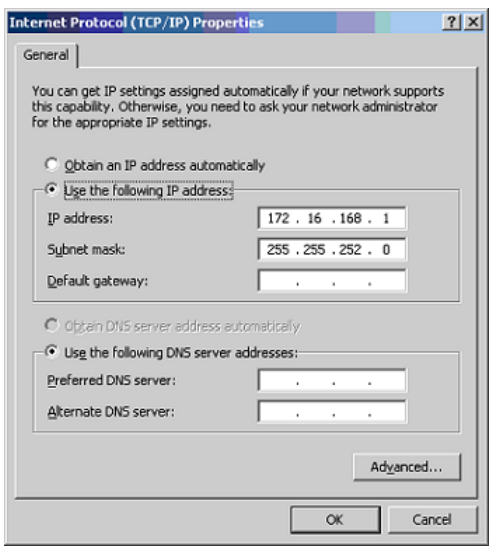


Figure 101. Internet Protocol (TCP/IP) Properties

- When the connection between connectivity server and PU410 is lost. For example, due to communication problem or power loss in PU410, it will be reported in the connectivity server after four seconds and appropriate actions

will be taken. The connectivity server will periodically try to reconnect and restart PU410.

- The PU410 unit communicates with 100Mbit/s on the PC 1 and 2 ports. It is recommended to configure the network interface card in the connectivity server to 100Mbit/s full duplex.

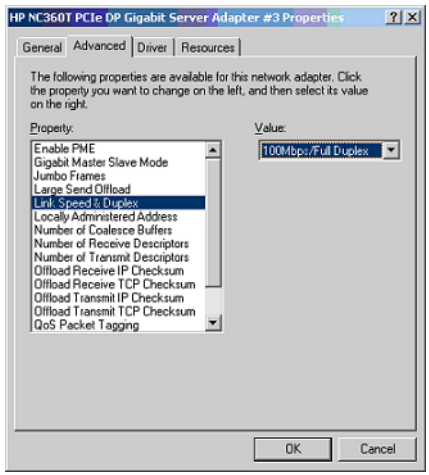


Figure 102. Example of NIC Configuration

- RNRP is used to supervise the communication between the connectivity server and PU410. The detection time for lost communication will increase from 4s to at least 30s if RNRP is not configured correctly. No additional RNRP configuration is required when the default implicit configuration of RNRP is used.
- The communication status is reported by MasterBus 300 RTA Management service provider object in Service structure. Use the Service Event List and System Status Viewer to see actual status.
- If the default implicit configuration of RNRP is modified, an explicit configuration of RNRP must be added for the PU410 connection. Configure “Network area” to 10 and “Network area local" to 1 if the default IP addresses for PU410 are used.

- The IP address and subnet mask of PU410 can be modified by a tool, but this is required only if Network area 10 (NetId 172.16.40.0) is already used in the connectivity server (refer to Modify IP address of PU410 on page 268).

  RNRP requires the IP address of the first port of PU410 to always start with 172.16 (in PU410 FW version 1.0.1.0 and earlier). The default IP addresses of PU410 are restored by pressing and holding down the reset switch of PU410 for about 20 seconds until the yellow IP LED is lit.

Modify IP address of PU410

The IP address of PU410 shall be modified only if NetId 172.16.40.0 is already used in connectivity server.

Execute the following steps to modify the IP address of PU410.

1. Start the *ManagementTool.exe* located in *C:\Program Files(x86)\ABB 800xA\AC 400 Connect\RTADriver\ManagementTool*.
2. Select the IP Config view and click **Get**.
3. Modify the IP settings and click **Set**.

The modified IP address will be effective when the PU410 is reset, in the Reset view or by pressing the reset switch of PU410 unit.

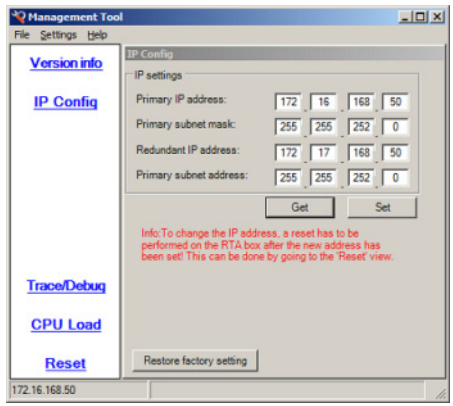


Figure 103. IP Config view of Management Tool

After modifying the IP address of PU410, the Target IP address of Management Tool must be modified in the **Settings** menu to get contact again. The Target IP address is shown at the bottom left in the Management Tool.

A registry setting must be modified in the connectivity server to match the modified IP address of PU410 (see Figure 104). Right click the IP address, select **Modify** from the context menu, and enter the Primary IP address of PU410.

An update of the 800xA for Advant Master SW in the connectivity server might reset the registry setting in the connectivity server to the default IP address.

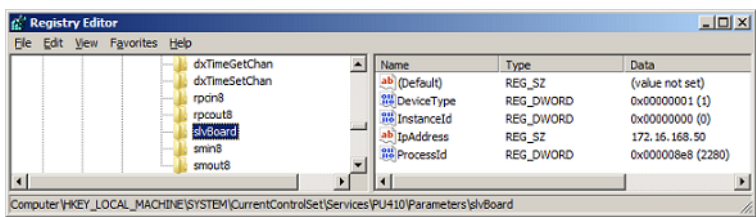


Figure 104. IP address setting in Registry

Hardware Use Cases

The different methods of connecting the connectivity server to the MB300 control network are described below.

Connecting PU410 to a 10BaseT MB300 Control Network

If the MB300 control network is made up with HUBs with RJ-45 ports then use shielded Cat 5e or better straight-through cables to connect the PU410 unit.

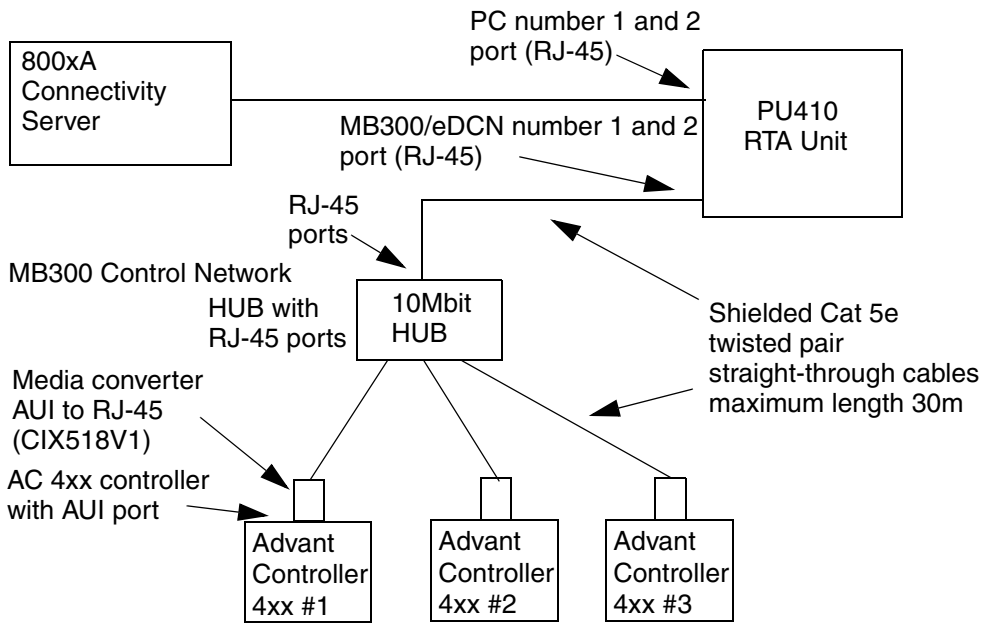


Figure 105. Connecting the PU410 to the MB300 10BaseT Control Network

Connection to MB300 Control Network (10Base5 “Thick Ethernet” cable)

Connection to a MasterBus 300, MB300, control network made with 10Base5 “Thick Ethernet” cable is shown below:

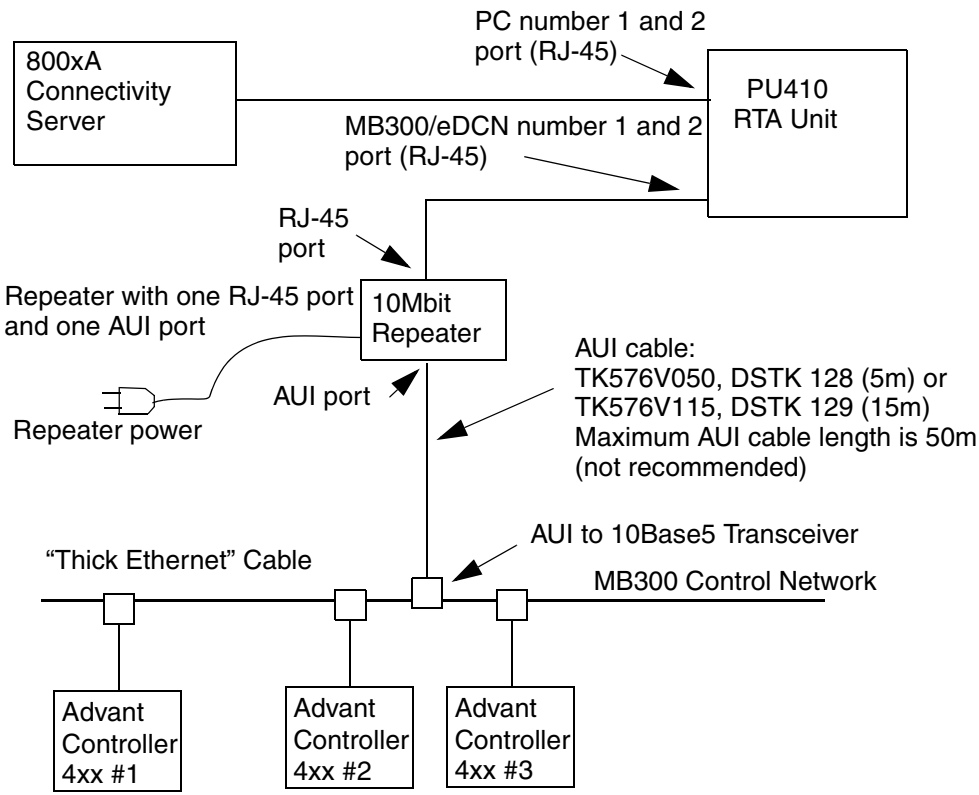


Figure 106. Connecting the PU410 to the MB300 10Base5 Control Network

- The MB300/eDCN RJ-45 connectors must be connected to a converter, repeater, that converts RJ45 interface to AUI interface.
- Note that the AUI to 10Base5 transceiver must be powered with 12Vdc 0.5A from the repeater to work. Therefore the repeater must have its own power supply.

- Note that the SQE/Heartbeat shall be disabled on the AUI to 10Base5 transceiver.
- Always use shielded Cat 5e twisted pair straight-through cables for connection between PU410 and the repeater to reduce electric noise disturbances. The maximum cable length is 30m but keep cable length as short as possible.
- The AUI cable between the repeater and the 10Base5 transceiver must have locking posts at the repeater end and a sliding latch at the transceiver end (standard AUI fixing).

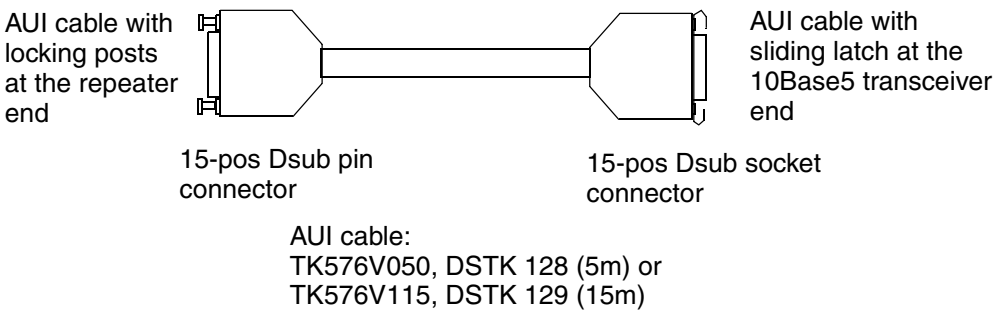


Figure 107. AUI Cables and Connectors for Connection to 10Base5 Transceiver

Replacing The AdvaCommad Unix Workstation With 800xA

The AdvaCommand Unix workstation has an internal RTA board for connection to the MB300 Control Network. None of these ISA/PCI boards (PU513, PU513V1, PU513V2, PU515, PU515A, PU519) can be reused in the 800xA connectivity server. Instead the PU410 unit must be used.

Appendix H Special Configuration

Some special behavior of Alarm & Event OPC server and Data Access (OPC DA) can be obtained by editing the configuration files located in the “800xA for Advant Master” installation folder in the connectivity servers. This is only recommended in very special occasions and should only be done by system engineers with thorough knowledge of the system. Notepad or Excel application can be used to edit the file. The numbers in bold below should be altered according to description for each parameter.

The configuration files are overwritten when system software is updated. User has to manually update these files in the Connectivity Server after installing a new Revision or Temporary Correction.

Alarm and Event

Examples of special configuration of Alarm and Event are:

- Mapping of Priorities to OPC Severity.
- Handling of events to 3’rd party clients.

The concerned configuration parameters reside in the following file in each connectivity server:

C:\Program Files(x86)\ABB 800xA\AC 400 Connect\bin\
AdvMbAeOPCServer.csv

Search the relevant line and alter the numbers to a suitable value according to comments in the file. After modifying the file, restarting the event collector is needed.

Mapping of Advant Master Priorities to OPC Severity

PriorityDef,1,**875**,

PriorityDef,2,**750**,

PriorityDef,3,**625**,

PriorityDef,4,**500**,

PriorityDef,5,**375**,

PriorityDef,6,**250**,

PriorityDef,7,**125**,

The values in bold (the OPC severity) above can be altered and the range can be from 1..999. Note that the value must be lower than previous line.

For more informtaion on how to map OPC Severity into 800xA Priority Level, see the “Advant Master OPC Event Server” Alarm Collector definition, aspect Alarm Priority mapping.

Definition of severity for Advant Operator Logging

By default, the severity of events from OperatorLogging (MB300) category is defined by the EventTreat DB element. This can be overridden by altering the parameter as given below:

SeverityOperatorLogDef,**0**,

Severity 1..999 is allowed

Use description text from controller in Alarm/Event lists

By default, the description from Name aspect is presented in the ObjectDescription column. The Name aspect is updated at Upload from the controller.

UseRemoteDescrDef,**0**,

0=Use Description text from Name Aspect (Default).

1=Use Description text from controller.

Keep System Alarms in Alarm List after acknowledge

By default system alarms are removed from alarm list when acknowledged.

RemoveSystemAlarmWhenAckedDef,**1**,

0=System Alarms remain in list after acknowledge.

1=System Alarms are removed from list when acknowledge (Default).

Sending of Events to third party clients

By default, no process events (only alarms) are sent to third party clients. The user can customize the behavior using the following three parameters.

1. Switch for sending of events as OPC type Simple.

   ```
   SimpleSendModeDef,0,
   ```

   This is used to send events as OPC type **Simple** to clients using the category SimpleProcess(MB300).

   0 = No sending (default)

   1 = Send Process Events as OPC type Simple

2. Switch for suppress events of OPC type **Condition**. It avoids dual events if **SimpleSendMode** switch is enabled.

   ```
   NormalSendModeDef,0,
   ```

   This is used to suppress the events of OPC type Condition using the category Process(MB300).

   0 = No (default)

   1 = Yes

3. Selections of text in Message attribute.

   ```
   SimpleMessageModeDef,0,
   ```

   This is the content of Message attribute from category SimpeProcess(MB300).

   0 = Only Message (default)

   1 = Description + Message

   2 = Condition + Message

   3 = Description + Condition + Message

Advant Master Alarm Refresh

Enable the Alarm Refresh function for the controllers:

```
RefreshActDef,0,
```

0 = OFF (default)

1= ON

Configuration of the maximum flow of Alarm Refresh status check requests per second. This parameter can be tuned to achieve a reasonable load influence on the controller load:

```
RefreshFlowDef,20,
```

Allowed interval: 4 to 200

Default: 20 request/second

The time lagging in seconds before Alarm Refresh is started. This can be set for the different Connectivity Servers to equalize the controller load.

```
RefreshDelayDef,10,
```

Allowed interval: 0 to 200 seconds

Default: 10 seconds

Data Access

One example of special configuration of Data Access is the configuration of Data Subscription flow control. The configuration parameters concerned reside in the following file in each connectivity server:

C:\Program Files (x86)\ABB 800xA\AC 400 Connect\bin\AdvDsMasterAdapter.csv

Find the relevant line, and alter the numbers to a suitable value according to comments in the file. Restart of the OPC DA Service provider is needed after some modifications.

Enable function Advant Operator Logging

UseEventTrackingDef,**0**,

0 = Off (default).

1 = On.

No restart is needed for this parameter.

Controller nodes not supporting fast write to DAT object

Older controllers (of type MP2x0) do not support the protocol for fast write to DAT objects. With this parameter it is possible to define those nodes and thus use the suitable protocol. Enter the node numbers for these (max 20) nodes. -1 = All nodes:

SlowDatNodeDef,**0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0**,

Restart is needed for this parameter.

Data subscription flow control

This data subscription flow control function divides the items in one OPC DA subscription request call (AddItem and RemoveItem) into several chunks with delay in-between. This is useful particularly when having large subscriptions from 3'rd party clients. This function prevents jam in data communication on MB300 when clients are started, which could lead to high controller load and lost subscriptions. A too slow setting will affect the display call up time negative.

How to configure:

By default, each AddItem and RemoveItem request is divided into chunks of 2000 items with a delay of 1000ms between each chunk. The parameters below can be altered to achieve a more evident flow control. No restart is needed after the parameters have been altered. Just save the configuration file:

AddItemsChunkSizeDef,**2000**,

AddItems requests with more items than this will be split up in several chunks with delay in between. Allowed value: 100.. 2000.

AddItemsChunkDelayDef,**1000**,

Delay time in ms between each chunk of added items. Allowed value: 0.. 4000.

Lock policy

The default lock policy for MB300 Process objects is "Locking optional for operation"which means that locking is not mandatory. The operator can operate without a preceding lock request.

With "Operation requires locking" policy, it is mandatory to lock the process object before operation is possible.

OperationRequiresLockingDef,**0**,

0 = Object locking is optional (Default).

1 = Object Locking is required for operation of objects supporting lock.

No restart is needed for this parameter. But faceplates must be reopened if this parameter is changed.

Object Lock timeout time

This parameter enables the user to define the time duration for an object to remain locked at inactivity.

LockPeriodDef,**270**,

Allowed interval is 30 to 270 s.

This parameter requires restart of connectivity servers after modification.

Revision History

This section provides information on the revision history of this User Manual.

The following table lists the revision history of this User Manual.

Revision Index	Description	Date
-	First version published for 800xA 6.0	December 2014
A	Updated information for the following topics: Configuring the Alarm Refresh and Controller behavior to consider. Additionally, removed references to VB Extensions as VBPG is not supported in System 800xA version 6.0 onwards.	October 2015

Index

Numerics

800xA for Advant Master Product 15
- Functionality 15
- Hardware and Software requirements 16
- Product overview 15

A

Advant Master Central Backup 133
Alarm Refresh 65
Alarm Refresh configuration 276
Audit List 69

B

Backup and Restore 122
Backup Info aspect 136
Building Control structure
- Create FCB source file 43
- Demo structure 47
- Off-line from FCB source file 43
- Test structure 47

C

Clock Master
- 800xA for Advant Master 25
- AC 400 Series Controller 23

Configuration 17, 97
- Before you start 17
- Getting started 21

Configuration of Advant Master Central Backup 134
Configuring a scheduled backup 139
Control Connection Aspect 237
- Additional information 239
- MasterBus 300 240
- Property information 238
- Property view 237

E

Engineering with Control Builder A (CBA) in an 800xA System 49
EVENT elements 151
Event text
- Analog Input signal AI 158
- Analog Output signal AO 159
- Binary object GENBIN 173
- Digital Input signal DI 160
- Digital Output signal DO 161
- Group alarm GRPALARM 187
- Group start GROUP 185
- Manual station MANSTN 168
- Motor Control MOTCON 179
- Process Controller PIDCON 164
- Process Controller PIDCONA 166
- Ratio station RATIOSTN 169
- Sequence SEQ 161
- Standard event text 56
- User defined controller GENCON 170
- User defined object GENUSD 176
- User-defined alarm and event text 57
- Valve control VALVECON 182

EVENT TREAT 145
Event treat 157
EVENT TREAT properties
- AL_FRBLK 146
- AL_PRIO 146
- AL_TOBLK 146
- PERSISTB 147
- TEXT_FRBLK 147
- TEXT_TOBLK 147
- TEXTCOMB 147

Event treatment 53
Overview 143, 156
Standard event text 55 to 56
User defined alarm text 57
User defined event text 57

F

Fault Tracing 103

H

Hot Keys in 800xA for Advant Master 31

L

Log index 72

M

Maintenance of Advant Master Central Backup 140
MB300 Upload
Upload from AC400 controller 42 to 43

O

OPC Logs 93

R

Restoring a backup 140
RTA Board - Get status 232
RTA Board configuration 22
Network and Node numbers 22
Save configuration 236
Setup Network and Node 235
Start On-Line Builder 236
Start/Stop of RTA 232
RTA Board PU515A 261
RTA Unit PU410 261

S

Selecting RTA Unit Character Conversion 23
Setting PU410 redundant cable supervision 22
Simulated process data
Export control structure 48
Property signal generator 47
Starting a manual backup 135
System Alarms 120
Alarm list for Controllers 120
Alarm list for network 120
System Error Messages 103
System Status
Controller node 111
Network 105
System Status - Service Structure 104

T

TTD Logs
Create log template 85
Create TTD log in a controller 91
Log template
Log definition 75
Node configuration, controller details 80
Node configuration, Download to controller 82
Node configuration, Main view 77
Node configuration, Upload from controller 81
Setup in 800xA System 83
View TTD log in a controller 84

Contact us

www.abb.com/800xA
www.abb.com/controlsystems

3BSE030340-600 A

Power and productivity
for a better world™