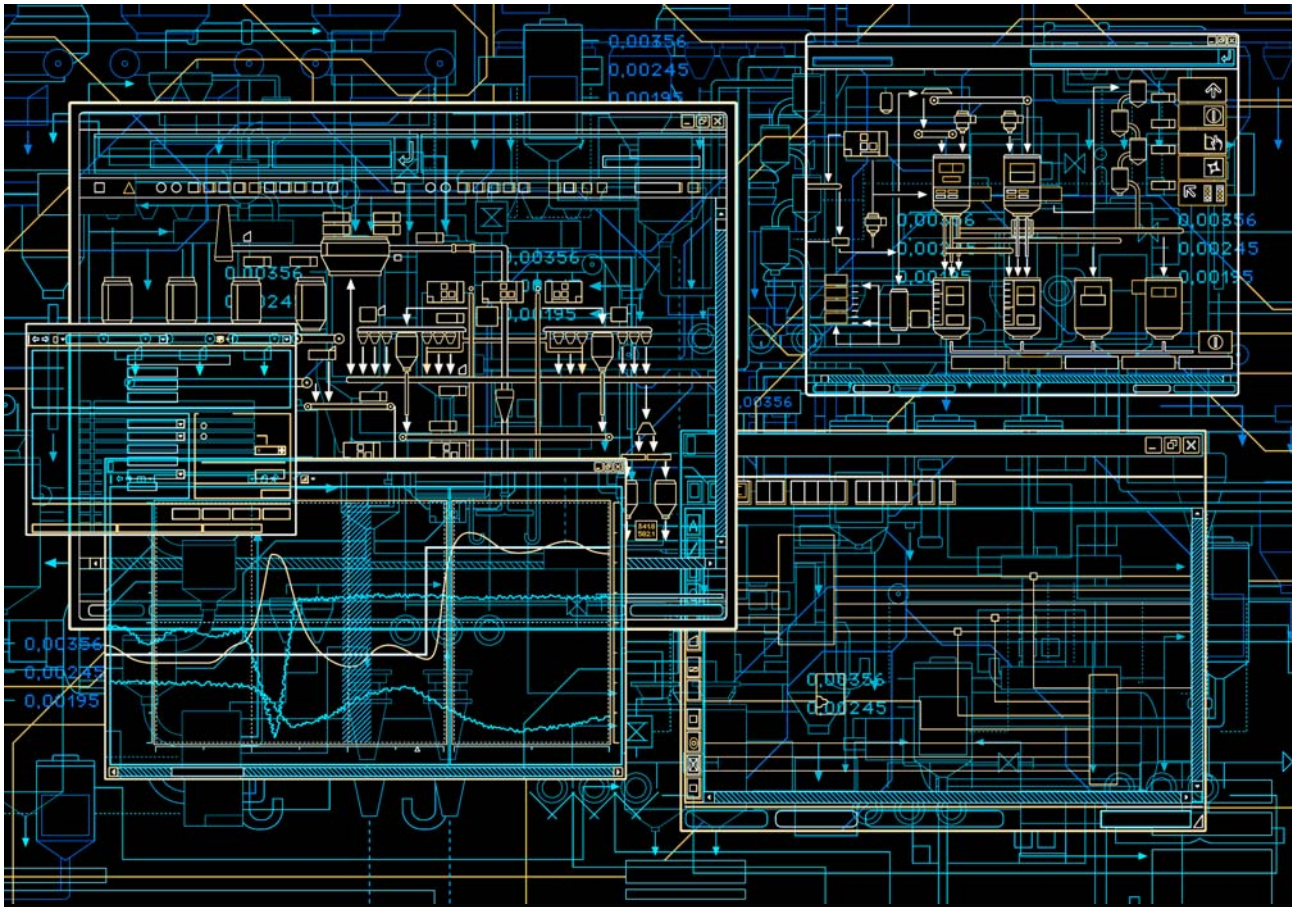


System 800xA Engineering

Engineering Studio Function Designer
Getting Started

System Version 5.1

Power and productivity
for a better world™

System 800xA Engineering

Engineering Studio Function Designer
Getting Started

System Version 5.1

NOTICE

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

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

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

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

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

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: December 2013
Document number: 3BDS100968-510 D

Table of Contents

About this User Manual

General 9
User Manual Conventions 10
Feature Pack 10
Warning, Caution, Information, and Tip Icons 11
Terminology 11
Released User Manuals and Release Notes 12

Section 1 - Introduction

Product Scope 16
Prerequisites and Requirements 18

Section 2 - Basic Operation

Reporting Problems 20
Starting Function Designer 20
Exploring User Interface 20
Exiting Function Designer 23
Working with Function Diagrams 25
Creating a new Function Diagram 25
Inserting Function Components 26
Configuring Favorite Object Types and Default Visible Ports 27
Showing / Hiding Ports 28
Reserving Function Diagram and Application 28
Adding Pages to a Function Diagram 29
Connecting Function Components 30
Navigating Through Function Diagrams 33

Allocating a Function Diagram 35
Configuring Data Flow Order 36
Allocating I/O 37
Generating Configuration Data 39
Downloading Configuration Data 39
Testing Configuration 40
Printing Documentation 41
Opening and Attaching BDM_for_Function_Diagrams 43
Verifying Diagram States 44
Types of Sequences 46
IEC 61131-3 Sequence 47
Sequence2D 47
Overview Diagram 49
Inserting a Sequence\Sequence2D Component 53
Editing Overview Diagram 54
Editing Step Detail Diagram 56
Editing Action Diagram 59
Editing Transition Detail Diagram 60
Deleting a Sequence 61
Navigating Between the Sequence Diagrams 61
Allocating, Generating Configuration Data, and Testing 61
Printing Documentation 62

Section 3 - Configuration

Process Model 63
Configuring Process Functions 65
Preparing Control Structure 66
Configuring Functional Structure 70
Configuring Function Diagrams 72
Working with Typicals 75
Configuring User-defined Object Types 82
Completing the Function 100
Reusing the Function 104

Allocating the Function Diagrams 106
Allocating the Signals 107
Configuring Interlock Connections 107
Generating Configuration Data 109
Downloading and Testing the Application 110
Printing Documentation 110
Process Sequence Model 113
Configuring Process Sequences 113
Configuring the Function Diagram 113
Configuring the Sequence 115
Reusing a Complete Sequence 118
Downloading and Testing the Application 119
Printing Documentation 120
Process Sequence2D Model 121
Configuring Sequence2D Processes 121
Configuring the Function Diagram 121
IO Allocation 129
Downloading and Testing the Application 129

Section 4 - Test and Commissioning

Displaying Live Values 132
Using Watch Window 134
CBM Views 137
Process Sequences On-Line 137
Displaying Live Values 137
Single-Stepping a Sequence 139
Stopping a Sequence 140

Index

Revision History

Updates in Revision Index A 145
Updates in Revision Index B 146

Updates in Revision Index C 146
Updates in Revision Index D 147

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 user manual describes how to use Function Designer to engineer and commission applications for AC 800M controllers in System 800xA. It outlines the basic workflows and also provides tutorials for practical exercise.

Section 1, Introduction provides an elementary description about Function Designer.

Section 2, Basic Operation describes how to startup Function Designer and the workflow steps required to work on Function Diagrams and Sequences.

Section 3, Configuration provides a tutorial on engineering and configuring Function Designer.

Section 4, Test and Commissioning provides a tutorial on commissioning Function Designer.

Users need to perform the described steps on an Engineering Workplace with Function Designer.

For more details on Function Designer, refer to *System 800xA Engineering, Engineering Studio Function Designer (3BDS011224*).*

For more details on Bulk Data Manager and IO Allocation, refer to *System 800xA Engineering, Engineering Studio (3BDS011223*)*.

Conceptual information on engineering with Function Designer is described in *System 800xA, System Planning (3BSE041389*)*.

For details on the configuration of Function Designer, refer to *System 800xA,Configuration (3BDS011222*)*.

For more details on the latest product updates, refer to the Release Notes delivered with the product.

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 that could result in *electrical shock.*

Warning icon indicates the presence of a hazard that 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 that 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.

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

Function Designer is the engineering portal for plant oriented functional planning with Aspect Objects and Symbol Objects. It provides a graphical user interface to easily engineer, document, and maintain the complex AC 800M control strategies in System 800xA.

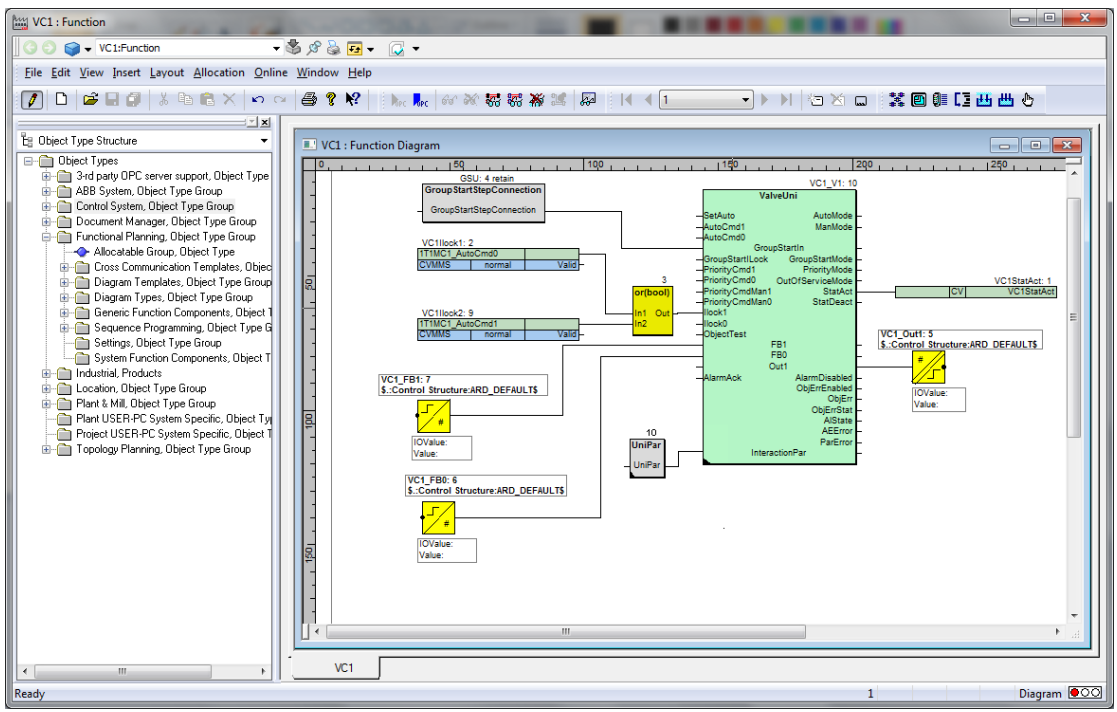


Figure 1. Sample Function Diagram

Function Designer is used to create Function Diagrams (refer Figure above). Function Diagrams are interlinked with Function aspect of the objects in Functional Structure. These objects represent Process Functions or Process Sequences of the plant.

Function Diagrams can be created as instances or templates in the Functional Structure. Additionally, they can be created as types in the libraries in the Object Type Structure.

A Function Diagram can be accessed in Diagram view from the Function aspect. User can insert and graphically connect the block symbols in a Function Diagram. The block symbols represent the Function Component Aspect Object types.

The standard library object types of AC 800M Connect and HART Device Integration are provided as ready-made Function Component Aspect Object types.

A Function Diagram can have a complete process control loop with Function Blocks / Control Modules, Field Devices, CBM_Signal objects (engineering signals), and their signal connections (refer to Figure above. Additionally, user can include textual comments, schematic drawings, and ActiveX components also.

User can include sequences into Function Diagrams and their steps, transitions, and actions can be configured in additional Function Diagrams.

User can generate configuration data for a Function Diagram after allocating it to an application in Control Builder M.

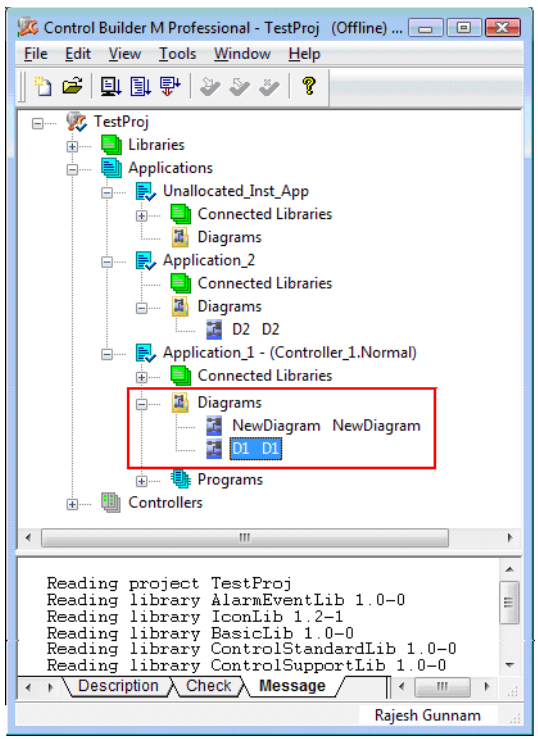


Figure 2. Function Diagram Represented as Diagram

During configuration data generation, Communication Variables automatically configure Cross Communication between connected diagrams in different applications.

User can download the diagram to the controller using Control Builder Professional. Function Designer has *Online Display* and *Watch Window* to test the diagrams. The Control Builder has the online functions to test the diagrams.

User can print Function Diagrams as a single diagram or in bulk for plant documentation on paper or on electronic media. Additionally the table of contents for the diagrams and documentation of hidden ports can also be included.

The Bulk Data Manager is used for copying, allocating, and generating configuration data of Function Diagrams in bulk.

The component view of the Function aspect is used to design graphical (block) symbols representing Function Component Aspect Object types.

Product Scope

The Function Designer is a component of the Engineering Workplace in the System 800xA.

The Function Designer, by default supports the functional planning approach for engineering of the AC 800M applications. The standard libraries of AC 800M Connect, HART Device Integration, and PROFIBUS Device Integration are used to create the logics for the following field devices:

- local and remote S800/S900 I/O
- HART devices
- PROFIBUS devices.

The user-defined Function Components are used to customize the Function Designer.

Enable the Environment support in the Configuration Wizard to use Function Designer in the Engineering and Production environment.

In the Configure and Deploy method, the Function Designer is used to configure applications. These configured applications can be downloaded through Load Evaluate Go method.

Configure and Deploy

The **Configure and Deploy** feature allows modifications in the Engineering Environment. These modifications will not affect the running process, until the modifications are deployed to the Production Environment for immediate implementation.

Load Evaluate Go

Load Evaluate Go allows the user to modify, download, and evaluate a different version of the current application to a controller without interfering with the operations of the current application.

A separate license is required to enable the **Configure and Deploy** and the **Load Evaluate Go** functions.

For more information on the **Configure and Deploy**, and on the **Load Evaluate Go**, refer to *System 800xA Engineering, Engineering and Production Environments (3BSE045030*)*.

The Function Designer is not certified for building the SIL-certified applications. It can be used to configure non-SIL-certified applications for the AC 800M High Integrity controller. SIL stands for Safety Integrity Level, as specified in the standard IEC-61508.

The Control Builder Professional is used to configure SIL certified applications. The SIL certified AC 800M High Integrity controller is required to run the SIL certified AC 800M applications.

For information on guidelines and safety considerations related to all safety life-cycle phases of an AC 800M High Integrity controller, refer to *System 800xA Safety, AC 800M High Integrity Safety Manual (3BNP004865*)*. This manual also lists the available SIL certified library object types.

For more information on SIL-application related configuration procedures to perform with Control Builder M Professional, refer to *System 800xA Control, AC 800M Configuration (3BSE035980*)*.

Prerequisites and Requirements

Prerequisites, requirements, and the installation procedures are explained in the *System 800xA, Installation* manual. Post installation procedures (including system creation and system extension loading) are explained in the *System 800xA, Post Installation* manual.

The following system extensions related to Function Designer have to be loaded:

- AC800M Connect.
- ABB Engineering Base.
- ABB DM & PM Application.
- ABB Signal Extension for AC800M Connect.
- ABB Function Designer for Fieldbus Builder PROFIBUS/HART.
- ABB CI Extension for AC 800M Connect.
- PROFIBUS & HART Device Integration Library - Basics.
- ABB Function Designer.
- ABB Function Designer for AC800M Connect.

Section 2 Basic Operation

User can verify the product version through **Control Panel** > **Programs** > **Programs and Features** in windows classic start menu. If the **Version** tab is not visible, then follow the steps to view the **Version** tab:

1. Right-click any available tab (such as **Name**, **Publisher**, etc).
2. Select **More...,** see Figure 3.

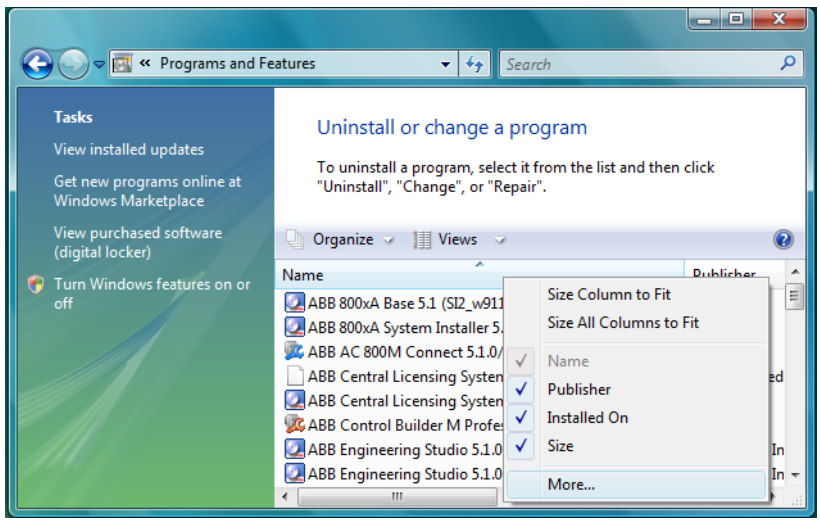


Figure 3. Context Menu to Access More Information

3. Select the **Version** check box.
4. Click **OK**.

The version details are displayed below the **Version** tab.

Follow the above procedure to view other tabs such as **Support Link**, **Support Telephone**, **Help Link**, etc.

User can also verify the product version from the Plant Explorer through **Help** > **About Industrial**$^{\mathbf{IT}}$.

Further, version of the specific product can be verified through its **Help** > **About** menu command. For example, version details of Function Designer can be verified through its **Help** > **About** menu command from the Diagram or Component view.

Reporting Problems

The support information contains the URL http://www.abb.com/service.

Using this URL you can report problems by E-mail to the responsible ABB service organization and you can retrieve the corresponding phone and fax number.

Make sure to report problems with the relevant information, such as the product version designation and build number displayed in the support information, the description of the problem scenario, and the detailed description of the error messages you got.

Starting Function Designer

Double-click Engineering Workplace icon on the PC's desktop to open the Workplace.

Engineering Workplace is the recommended workplace, as it provides useful engineering commands and functions in the **Advanced** context menu of aspect objects, for example Bulk Data Manager and I/O Allocation.

In the Plant Explorer or Engineering Workplace, select Functional Structure. Create or open a Function Aspect of any Aspect Object, select Component view or Diagram view to start the Function Designer.

Exploring User Interface

If the user clicks on the Function aspect, Function Designer displays the Function Diagram (**Diagram** view) in the preview area of the Plant Explorer as shown in

Figure below. The preview offers full functionality for working with function diagrams.

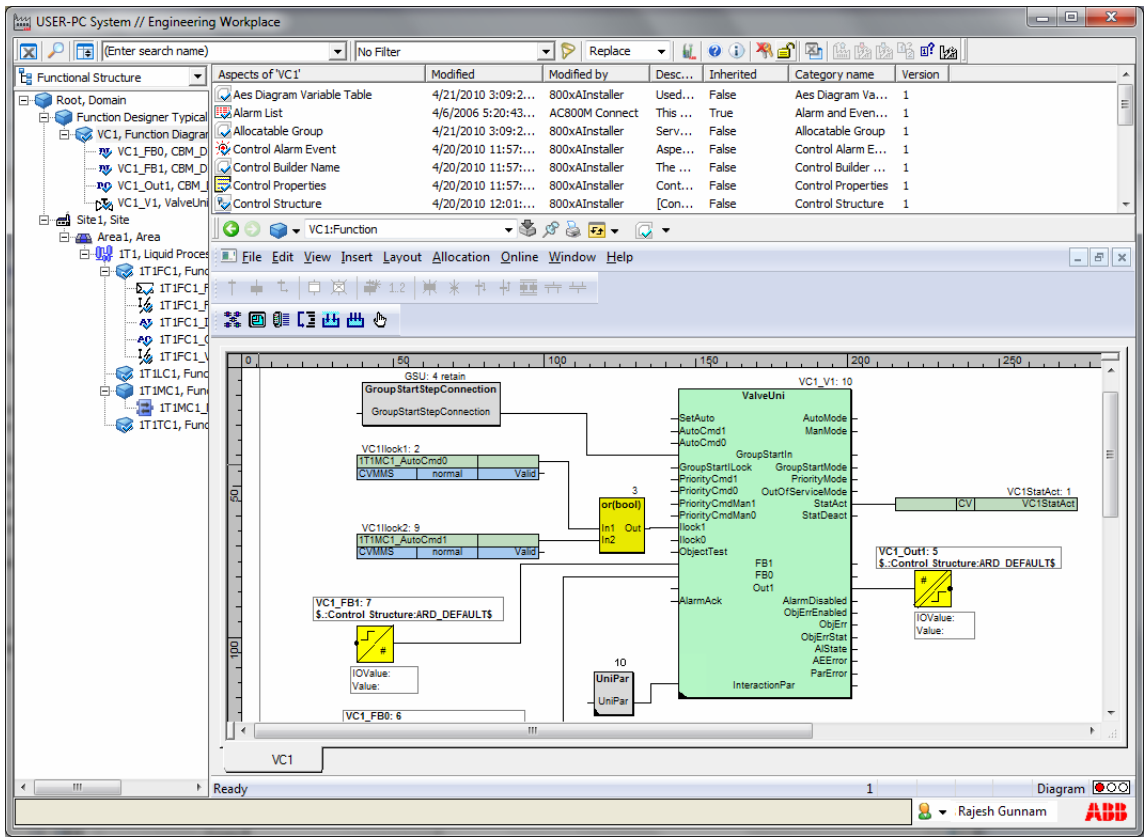


Figure 4. Function Designer in Plant Explorer Preview Window

To open Function Designer in a separate (popup / overlap) window as shown in Figure below, double-click the Function Aspect. Alternatively, right-click the Function Aspect and choose **Diagram** from the context menu. Choosing

Component from the context menu, can directly open the Function Aspect's Component View.

It is recommended to switch off the preview area in Plant Explorer using the **Toggle Preview** button and to work with Function Designer in a separate (popup / overlap) window.

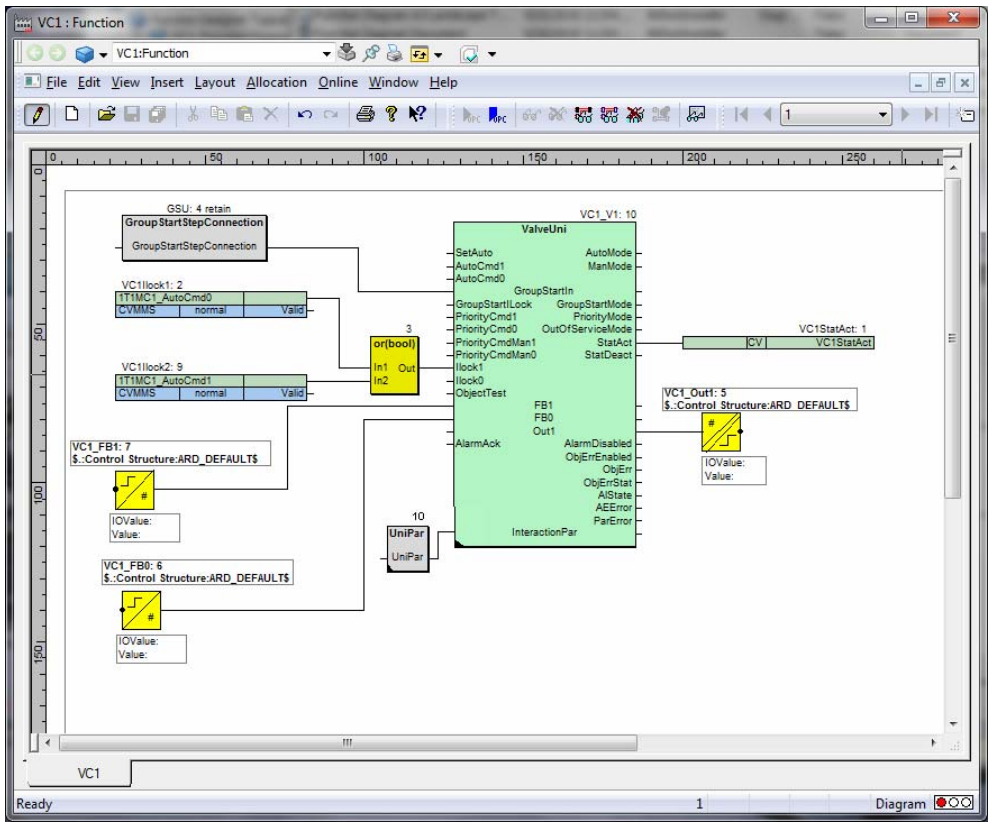


Figure 5. Function Designer in Plant Explorer Popup Window

The corresponding window title available with the **:Function** suffix in the drop down list at the top of the window area indicates the currently displayed Function aspect.

Typically the Function Designer user interface preferences is set such that within the preview window only a minimal set of menus, toolbars, and windows are displayed. But, within the Engineering Workplace popup window, a more comprehensive set is displayed.

Right-click on the menu bar of Function Designer to access the context menu and select the required user interface items which need to be included. If the user intends to open the Function Designer in the preview window and the Engineering Workplace overlap window, customize the menu bar in both views.

Exiting Function Designer

If a Function Diagram is edited, save the data using **File > Save** or **File > Save All** or using **File > Generate Configuration Data (Full Build)** or **File > Generate Configuration Data** before exiting Function Designer.

Any attempt to exit Function Designer before saving the data leads to a message as shown in the figure below.

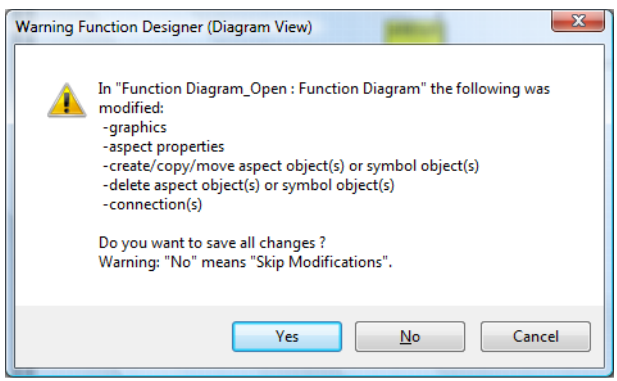


Figure 6. Exiting Function Designer

If user clicks **No**, then a warning message appears as shown in Figure 7.

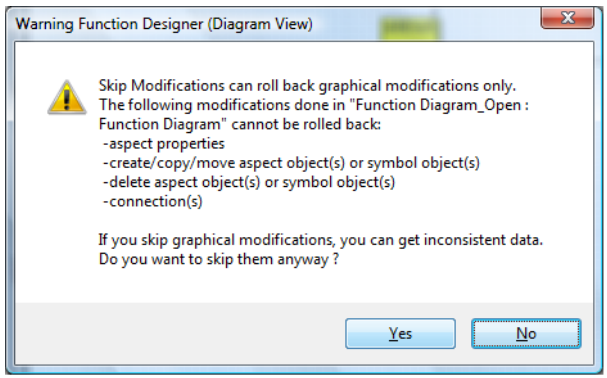


Figure 7. Warning Message if User Exits Function Designer Without Saving Changes

If the user clicks **No** again, another message appears as shown in Figure 8 to reconfirm if Function Designer can be quit without saving the data changes.

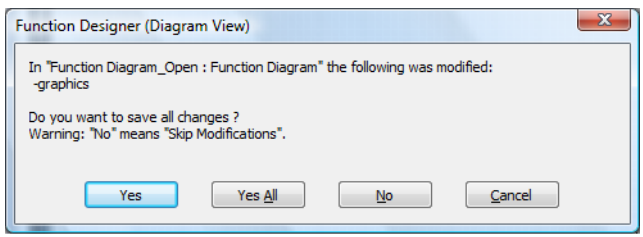


Figure 8. Attempt to Exit Function Designer Without Saving the Changes

Function Designer can rollback only the graphical changes.

To avoid possible inconsistencies always save the necessary data changes.

Always save the modified Function Diagram to ensure that all user-defined function blocks / control modules are visible to users having Read Only permission.

Working with Function Diagrams

Few elementary workflows using Function Diagrams is described in this subsection.

The workflows presented can be used in Engineering Environment enabled systems.

The workflow described in this subsection does not include the preparation of the Control Structure. For information on the Control Structure, refer to Preparing Control Structure. To work with Function Designer, a project must be assigned to an application and a controller in Control Builder M.

Creating a new Function Diagram

Follow the steps to create an object using a ready-made diagram template object type that shall hold a Function Diagram for a process function or sequence:

1. In the Plant Explorer, navigate to a parent object in the Functional Structure.
2. Right-click the parent object and click **New Object...** from the context menu to open the **New Object** dialog.
3. In the structure browser of this dialog, navigate to and click an appropriate pre-configured diagram object type under Object Types\Functional Planning\Diagram Types\Based on Diagram Templates, such as Function Diagram A3 Landscape.
4. Enter a name in the **Name** field.
5. Enter a description in the **Object Description** field.
6. Click **Create.**

An instance of the object type is created with the entered object name. This object contains a Function aspect with an empty Function Diagram, derived from the A3 Landscape template. It also contains a Function Diagram Document aspect that can be used to enter administrative data (meta data) to be displayed in the Function Diagram footer and to represent the Function Diagram in Document Management.

Alternatively, a generic object can be created and a Function aspect can be added. Use **File > New** in Function Designer to select the required Function Diagram template.

It is recommended to always:

- Allocate the Function Diagram to an application, since any modifications to a diagram which is assigned to the unallocated applications, may result in inconsistent data.
- Save and generate configuration data for Function Diagrams before performing copy/paste operation.

Inserting Function Components

Ensure that Control Builder M is started with an appropriate project. For information on starting and opening a project in Control Builder M, refer to *System 800xA Control, AC 800M Getting Started (3BSE041880*)*.

Follow the steps to insert an instance of a Function Component object type into a Function Diagram:

1. Navigate to the required object holding the Function aspect.
2. Double-click the Function aspect to open the Function Designer.
3. Right-click the tool bar of Function Designer and select **Structure Browser 1** from the context menu, if the structure browser is not visible.
4. Select **Object Type Structure** in the combo box of the browser window.
5. Navigate to a Functional Component Object Type in Object Type Structure, for example to Object Types\Control System\AC 800M/C Connect\Libraries\ProcessObjExtLib x.x-x\Control Module Types\MotorUniM.
6. Drag and drop **MotorUniM** into the Function Diagram drawing area.
7. Enter a valid name and description in the **New Component Name** dialog. Retain the default value in the **Data Flow Order Insert Before** drop-down.
8. Click **OK**.

The graphical block symbol for the selected Function Component appears on the Function Diagram. **Name** and **Description** ports are connected to the entered strings. Other ports are still hidden.

Other Function Components such as Off - Diagram References, Variables, CBM_Signals, HART Devices, Control Modules, Function Blocks, and System Functions can be inserted accordingly.

Function Components can be inserted into a Function Diagram by any of the following methods:

- **Insert** > **Object...** menu item in Function Designer.
- Drag and drop the required Function Component from the **Object Type Structure** in the Plant Explorer to the Function Diagram.
- Using the **New Object...** context menu of the parent object in the Plant Explorer.

For information regarding **Insert** > **Object...** menu item, refer to *System 800xA Engineering, Engineering Studio Function Designer (3BDS011224*).*

Configuring Favorite Object Types and Default Visible Ports

The **Insert Object** dialog can be customized to organize user favorite Object Types in pages. These pages can consist of the more frequently used Object Types. There is also an option to display the required visible ports for each of these frequently used object types.

Perform the following steps to configure the pages and ports of frequently used function blocks:

1. In a reserved Function Diagram, click **Insert** > **Object**.
2. Click **Configure...**, to open the **Configure Favorite Object Types** dialog box.
3. Click **Add Page**.
4. Provide a name in the **Favorites Page Name** window.
5. Click **OK**. Now a new tab appears in the **Configure Favorite Object Types** dialog.
6. Drag and drop the required object (ex. MotorUniM) from the structure browser window to the right pane of the **Configure Favorite Object Types** dialog.
7. Click **Edit Port(s)**.
8. Select the required check boxes for the desired ports.

9. Click **OK**.
10. Repeat Step 6 to Step 9 for other required objects.
11. Click **OK** in the **Configure Favorite Object Types** dialog.
12. Drag and drop the configured objects from the object list of the newly available tab in the **Insert Objects** dialog to the Function Diagram.

Showing / Hiding Ports

By default, newly instantiated components do not display all ports. Symbols with hidden ports are indicated by a small black triangle displayed in the lower left corner. The ports of the instantiated components can be made visible or hidden by using the Show/Hide functionality.

1. In diagram view, select the required component with hidden ports.
2. Right-click and select **Show Hidden Ports(s)**.
3. The **Show Hidden Ports** dialog appears.
4. Select **All Hidden Ports** check box or the required individual hidden ports.
5. Click **OK**.

Follow the steps to hide individual ports:

1. Select the port to be hidden in the Diagram view of the required symbol.
2. Right-click on the port and invoke the command **Hide Port(s)** from the context menu.

Reserving Function Diagram and Application

Function Designer requires reservation of an existing, non-blank Function Diagram for further modification.

Function Diagrams that are not reserved are read only and display the READ message in the status bar.

Functions related to an application such as **Allocate** or **Generate Configuration Data** require reservation of the application.

1. Click **File > Reserve Diagram** to acquire exclusive modify access to a Function Diagram. Click **File > Reserve Diagram** again to release the diagram.
   Click the corresponding **Reserve** / **Release**  toggle button of the Standard toolbar to achieve the same results.

By default, a newly created diagram is reserved. Closing the diagram releases the diagram automatically.

An application can be reserved in the Control Builder M or the **Manage** dialog of Plant Explorer.

2. Click **File > Release Reservation** to cancel reservation of a diagram or application.

In the diagram view of a reserved Function Diagram, all the modifications are saved automatically, while releasing the reservation through Engineering Workplace using the **Release…** command in the context menu.

Graphical changes such as, moving the existing Function Blocks / Control Modules, Field Devices, CBM_Signal objects and their signal connections are not saved by this operation.
It is always recommended to release a Function Diagram from the Function Designer.

Adding Pages to a Function Diagram

A Function Diagram can have several pages. Follow the steps to add a page:

1. Reserve the Function Diagram and select **Insert > Page...**.
2. The **New Page Function Designer (Diagram View)** appears with the message “Do you want to insert the new page before the 1st page?”. Click **Yes** or **No** accordingly.
3. Provide a valid comment in the **Enter Page Comment** dialog.
4. Click **OK** to insert the entered comment at the right, lower corner of the drawing area of the new page or click **Cancel** to insert the new page without comment.

The new page gets inserted. Inspect the page number shown in the lower right corner of the diagram document footer and in the combo box of the Page tool bar.

Connecting Function Components

Function Components are represented by symbols with ports. Several cases are available to connect the ports of the different symbols:

Port to String Connection

Follow the steps to connect a port to a string:

1. Click the port to be connected.
2. Enter the required connection string in the edit message box of the Connection tool bar.
3. Click **Enter.**

The connection string now appears at the port and is available in the combo box for reuse.

Ports on the Same Page

Follow the steps to connect different ports of the same page:

1. Click the source port and drag the mouse to the sink port.
2. Release the mouse button.

The graphical connection link gets automatically routed.

Ports on Different Pages

Follow the steps to connect ports of different pages:

1. Click and select the source port.
2. Select the required page from the combo box of the Page tool bar. (Or use **Page Up**/**Page Down** keys to navigate).
3. Press **ALT** and click the mouse simultaneously on the sink port.

Off-page references connected to the graphical connection links are created automatically.

To achieve maximum performance in Function Designer and to avoid auto routing issues, it is recommended to have:

1. Less than five pages per Function Diagram.
2. Less than 10 blocks per page.
3. Less than 4MB blob size for a Function Diagram.

Ports on Different Function Diagrams

Follow the steps to connect between ports available on different Function Diagrams:

In Source Diagram:

Follow the steps on the required port in the source diagram:

1. Open the Function Diagram containing the source port.
2. Right-click the required source port and select **New Diagram Output Reference...** from the context menu.
3. A Reduced Output Reference is inserted and a **Variable Properties** dialog appears.
4. Accept the defaults or change properties.
5. Click **Connect...** and the **Connect To Diagram Variable** dialog appears.
6. Click **OK** to accept the default name for the Communication Variable and to return to the **Variable Properties** dialog.
7. Click **OK**.

The Communication Variable connected to the Reduced Output Reference is now created and can be used in the sink diagram.

In Sink Diagram:

Follow the steps on the required port in the sink diagram:

1. Open the Function Diagram containing the sink port.

2. Right-click the required sink port and select **New Diagram Input Reference...** from the context menu.
3. An Input Reference is inserted and a **Variable Properties** dialog appears.
4. Accept the defaults or change properties.
5. Click **Connect...** and the **Connect To Diagram Variable** dialog appears.
6. Select the Communication Variable created with the Reduced Output Reference from the list available.
7. Click **OK** and return to the **Variable Properties** dialog.
8. Click **OK**.

The Communication Variable is now connected to the Reduced Input Reference and both Reduced Output Reference and Reduced Input Reference are updated with cross reference information.

Port to Diagram Parameter

Diagram parameters can be used only in Single Control Module based Function Diagrams or in Control Module Types created using Function Diagrams in the Object Type Structure.

Follow the steps to connect the port to the diagram parameters:

1. In Object Type Structure navigate to Object Types\Functional Planning\Generic Function Components\Connectors\Diagram Parameters.
2. Depending on the direction of the port to be connected to the Diagram Parameter select Input Parameter or Output Parameter or InOutParameter.
3. Drag & drop the selected object type onto the Function Diagram.
4. Edit or accept the defaults of the **Variable Properties** dialog.
5. Connect the port of the Diagram Parameter with the port of the Function, Function Block or Control Module to be exposed as Diagram Parameter.

Creation and connection procedures for diagram references, diagram parameters and variables can be increased by using the port context menu: Right-click the port to see this menu. Use item **New Diagram Input** (Output) **Reference** or **New Diagram Input** (Output) (InOut) **Parameter** or **New Variable**.

Navigating Through Function Diagrams

Page References

Follow the steps to navigate through the different pages of a Function Diagram based on the signal flow:

1. Click and select an output page reference symbol.
2. Press the **Right Arrow** key.
3. The corresponding diagram page is shown with the corresponding input page reference symbol selected.
4. Press the **Left Arrow** key to go back.

User can navigate through the pages of a Function Diagram by one of the following methods:

Method 1:

Use the combo box of the **Page** tool bar to switch between pages of a Function Diagram.

Method 2:

Use the menu item **View > Pages > Goto Page**.

Diagram References

Follow the steps to navigate through different Function Diagrams based on the signal flow between them:

1. Select and right-click an output diagram reference symbol to access the context menu.
2. Click **Goto Reference...**.

3. If the output diagram reference refers to more than one input diagram reference, a **Goto Reference** dialog appears, displaying the list of references. Select the required reference and click **OK.**

The Function Diagram with the corresponding input diagram reference appears with the reference selected.

Follow the steps to navigate to the initially opened Function Diagram:

1. Right-click the input diagram reference to access the context menu.
2. Click **Goto Reference...**.

The Function Diagram with the corresponding output diagram reference appears with the reference selected.

From 800xA 5.1 onwards, the newly created output diagram reference symbol indicates if the connected variable is a communication variable (CV) or a global variable (GV). For more details, refer to *System 800xA Engineering, Engineering Studio Function Designer (3BDS011224*).*

If the menu **Window >Workbook Mode** is selected (as it is by default) then both diagrams are opened simultaneously in different tabs and can be switched by clicking the corresponding workbook tab.

Parent - Child Diagrams

User can navigate between Function Diagrams with a parent-child relation (nested diagrams) based on hierarchy:

1. Right-click the symbol representing the child diagram (colored in light blue) to open the context menu.
2. Click **Goto Child Diagram**. The child Function Diagram appears.
3. To navigate to the parent diagram, right-click a blank space in the Function Diagram and access the context menu.
4. Click **Goto Parent Diagram.** The parent Function Diagram appears.

Communication between the parent and the child diagrams of newly created Function Diagrams can be achieved through communication variables.

Arbitrary Diagrams

User can navigate between the required Function Diagrams in Function Designer:

1. Click **File > Open...**. The **Open Aspect** dialog appears.
2. Double-click the required object to access the Function aspect.
3. Double-click the Function aspect to open the Function Diagram.

If the menu **Window >Workbook Mode** is selected (as it is by default), then click the respective workbook tab to navigate through all the opened diagrams in a Function Designer session.

The user can select the following options from the drop down list of the Engineering Workplace toolbar:

1. **Replace** - An opened Function Diagram closes on opening another Function Diagram.
2. **Preserve** - Each Function Diagram opens in a separate window without closing the previously opened Function Diagrams.

Function Diagrams can also be opened using the aspect list of the Plant Explorer, but it does not participate in the Function Designer **Workbook Mode** and closes the previously opened diagram.

Allocating a Function Diagram

Follow the steps to allocate a Function Diagram to the required application using the Allocatable Group aspect:

1. Click **Allocate Or UnAllocate Diagram** icon from the **Quick Access** toolbar.
2. In the **Allocatable Group** dialog, click **Allocate...**.
3. The **Allocate Diagram to Control Builder Application** dialog appears. Select the required application to which the Function Diagram must be allocated.
4. Click **OK**.
5. To close the **Allocatable Group** dialog click **Close** on window level.

Alternatively right-click the blank space of a Function Diagram and select the **Allocatable Group** aspect. User can also open the **Config View** of the **Allocatable Group** aspect from the context menu of the aspect in the Plant Explorer.

Allocating Function Diagrams can also be done in bulk, see Allocating the Function Diagrams.

Configuring Data Flow Order

Data Flow Order can be used to influence the configuration data generation of Function Designer for Control Builder M.

When a new Function Component is instantiated, a default Data Flow Order is assigned. If the **Extended Name** dialog or the **Variable Properties** dialog pops up during instantiation, the default Data Flow Order number is changeable.

Click **Auto Sort Order**  icon from the **Quick Access** toolbar to sort function components automatically according to the Data Flow Order.

Auto Sort Order should not be performed inside the sequence.

The **Generate** button can be used to sort the function components available in the Function Diagram according to the default Data Flow Order. The default order is calculated according the rule “left up to right down”. After generating the default Data Flow Order, click **Apply** to assign the Data Flow Order accordingly.

After the Function Diagram is edited, (re-)generate the Data Flow Order before generating configuration data:

1. Click **Allocation > Define Data Flow Order...** to open the dialog Data Flow Order in Allocatable Group.
2. Modify the Data Flow Order by selecting a function component from the **Data Flow Order** list and move it to the required location using the **Up** or **Down** arrow buttons.
3. Click **Apply** to assign the new Data Flow Order to the respective function components.
4. Click **Close** to exit.

Allocating I/O

The IO Allocation tool is used to allocate engineering signals (CBM_Signal objects) of the Function Diagram to the I/O boards of a Controller:

1. Click **IO Allocation**  icon from the **Quick Access** toolbar.
2. In the grid, on the right side of the **IO - Allocation** dialog all engineering signal objects inserted in the Function Diagram are shown.
3. In the tree, on the left side of the **IO-Allocation** dialog right-click the Boards object and select **Insert Board** from the available context menu.
4. In the **Insert Board** dialog, navigate to the required location and select **ModuleBus**, click **Insert**.
5. Click **Close** to exit the **Insert Board** dialog.
6. Drag and Drop one or several signal objects from the grid on the right side of the IO-Allocation dialog to the Boards object, or to an I/O board object, or to a channel stub of an I/O board object on the left side. After Drop the allocation is shown on both sides.
7. Click **File > Exit** to close the I/O Allocation dialog**.**
8. To verify allocation repeat steps 1 to 3, where the used boards are shown.

Alternatively right-click the blank space of a Function Diagram and select **IO Allocation** to open the **IO - Allocation** dialog with the context of the current Function Diagram or use the short cut keys CTRL+SHIFT+I to access the IO Allocation dialog.

In the Plant Explorer object context menu, execute **Advanced > IO Allocation** to open the **IO - Allocation** dialog for other contexts such as a Controller.

Disable the function **Automatic Write Allocation into CBM** in the IO Allocation tool before performing import operation.

IO Allocation supports allocating signal objects to channels of I/O boards of HART devices. From 800xA 5.0 SP2 onwards IO Allocation also supports allocating signal objects to channels of modules of PROFIBUS devices.

HART and PROFIBUS specific device libraries can be installed using the Device Library Wizard. User has to prepare these libraries using HWDProcessor to use them in Engineering Studio for allocating or de-allocating signals to the channels of HART or PROFIBUS modules.

For more information about preparation of HART and PROFIBUS devices, refer to *System 800xA Engineering, Engineering Studio (3BDS011223*).*

IO Allocation tool supports allocating signal objects to the channels of the below mentioned communication interfaces:

- CI801 S800
- CI840 S800
- CI853 Serial Communication
- CI854 Profibus DP
- CI855 MasterBus 300
- CI856 S100
- CI857 Insum
- CI858 Drive Bus
- CI871 Profinet
- CI862 Trio
- CI865 Sattbus
- CI867 Modbus TCP
- CI868 IEC 61850
- CI920 S900

It does not support any interfaces connected to the communication interfaces.

Generating Configuration Data

Execute the following to generate configuration data for a Function Diagram for Control Builder M and transfer the result to the required application it is allocated to:

Click **Generate Configuration Data (Full Build)**  icon from the **Quick Access** toolbar.

The message window shows the progress during the generation.

If **DisplayDetailsOnGenerateConfigData** is set to **True** in the **Function Settings** aspect of: Object Type Structure\Object Types\Functional Planning\Settings, and if errors occur, the message window stays open to allow the user to inspect the error messages. Click **Close** to continue.

Alternatively use **File > Generate Configuration Data (Full Build)** or open the Config View of the **Allocatable Group** aspect from the context menu of the aspect in Engineering Workplace and click **Generate Configuration Data (Full Build)**.

Generating Configuration Data for Function Diagrams can also be done in bulk, refer to Click File > Exit to close the I/O Allocation dialog.

If any Function Diagram is modified, it is advisable to generate Configuration Data for only that Function Diagram.

Generating Configuration Data in bulk for the Function Diagrams using Bulk Data Manager, requires more time.

The generated Function Diagram appears as an FuD_Codeblock tab, in the Diagram Viewer of Control Builder. If Sequence objects or Sequence2D objects are present within the same Function Diagram, these appear as separate tabs in Diagram Viewer.

Downloading Configuration Data

1. Ensure the following in the Project Explorer:
   - The required project is open and free of errors.
   - A task is assigned to the required application.
   - The project is configured with the required controllers to run the applications.
2. Click **Tools** > **Download Project and Go Online** in Control Builder M to load the generated configuration data to the controller.

For more information, refer to or in the corresponding subsections of *System 800xA Configuration (3BDS011222*) and System 800xA Control AC 800M Getting Started (3BSE041880*)* manuals.

Testing Configuration

After downloading the corresponding application, user can display online values and force the values using an integrated Watch Window. This allows the user to monitor and influence the signal flow.

Displaying Online Values in Function Diagram

Click the **Subscribe for Live Data For All Output Ports**  icon from the **Quick Access** toolbar.

Alternatively, follow the steps to display the value of a port:

1. Click and select the required port on Function Diagram.
2. Click **Online > Subscribe for Live Data All Output Ports.**
   Now the value gets displayed at the output port and also gets updated.

In case of port connection for boolean values, the connection lines indicate red color or green color based on the value transferred between the ports.

Displaying and Forcing Online Values in Watch Window

Follow the steps to display and force a value of a port:

1. Click **Online > Watch Window** to insert the Watch Window into the Function Designers main window.
2. In the Watch Window tool bar, click **Add variables to the Watch Window.**
3. In the structure selection (left) pane of the **Add Variable to Watch Window** dialog, navigate to the object for which an online value will be displayed.
4. Click on the variable in the grid of the right pane of the dialog to select it.
5. Click **Apply** and Click **Close**.
6. Enter the force value in the Prepared Value field for the variable in the Watch Window grid.

7. Click **Activate** to force the variable value.

Alternatively, user can test the configuration from within Function Diagrams using:

- Control Builder M Professional online editors (and interaction windows, if available) on Control Modules, Function Blocks, or Functions which are invoked using the context menu on the representing symbol in Function Diagram.
- Faceplates of Control Modules or Function Blocks which are invoked using the context menu on the representing symbol in Function Diagram.

Printing Documentation

User can print documentation for a single Function Diagram or for all Function Diagrams of a subtree of objects with or without a generated contents diagram. Additional pages can be selected to be included with port documentation information and information on instances of diagram types.

The following printing workflows assume that Windows standard printer and Page Layout settings are already performed.

Single Diagrams

Follow the steps to preview and print an already opened single Function Diagram without additional information:

1. Click **File > Print Preview** to open the preview of the Function Diagram.
2. In the preview window click **Print,** to open the Print dialog for the printer or click **Close** to discard printing.
3. Click **OK** to print or click **Cancel** to discard printing.

Preview and print out are based on the settings of the **Print Components on Layers** panel present in the **Print** tab of the **Options** dialog. User can access this dialog through **Edit > Options**.

Follow the steps to print an already opened single Function Diagram without preview and without additional information:

1. Click **File > Print...** to open the **Print** dialog.
2. Accept the defaults of all check boxes and radio options of the **Print** dialog.

3. Click **OK** to open the Print dialog for the printer.
4. Click **OK** to print or click **Cancel** to discard printing.

Follow the steps to print an already opened single Function Diagram without preview but with the maximum additional information on additional pages:

1. Click **File > Print...**.
2. In the **Print** dialog, select the **Diagrams** and **Print as one Job** check boxes.
3. Select the **Instances of Diagram Types** check box, if the Function Diagrams of the used types with instance specific information is additionally required to be printed out.
4. Select the Port Documentation check boxes (**Hidden Ports with Modified Initial Value**, **Connected Hidden Ports**, **Visible Ports with Modified Initial Value**) as required. This will generate and print out additional pages with information for these kinds of ports.
5. Select the **Hide Online Values** check box if user wants to mask out online values from the printed Function Diagram.
6. Click **OK** to open the **Print** dialog for the printer.
7. Click **OK** to print or click **Cancel** to discard printing.

Multiple Diagrams Including Contents Diagram

Follow the steps to prepare printing of all Function Diagrams of a sub-tree including a leading contents diagram:

1. Create a Function aspect in the top object of the sub-tree.
2. Double-click the Function aspect in Engineering Workplace to open Function Designer.
3. Click **File > Create Contents...** to open the **Create Contents** dialog.
4. Either select the radio option **Complete Contents** or **Overview Contents**. Complete contents includes the page comments of the Function Diagrams.
5. Click one of the listed templates to select it.
6. Click **OK** to generate the contents of the contents diagram.

To print out the diagrams perform Step 1 to Step 7 as described in Single Diagrams, but in Step 2 additionally select the **Contents** check box in the **Print** dialog.

Opening and Attaching BDM_for_Function_Diagrams

Engineering Workplace is the recommended workplace, as it provides useful engineering commands and functions in the **Advanced** context menu of aspect objects, for example Bulk Data Manager and I/O Allocation.

A pre-configured Bulk Data Manager workbook BDM_for_Function_Diagrams is provided to support copying, allocating, and generating configuration data of Function Diagrams.

To open and attach this workbook to a start object:

1. In Engineering Workplace, right-click the required start object in Functional Structure and click **Advanced > Engineering Templates** to open the Engineering templates folder.
2. Select BDM_for_Function_Diagrams.xlsx.
3. In the Bulk Data Manager toolbar:
   a. Select **Activate** from the Bulk Data Manager option present in the **Add-Ins** menu.
   b. Click the **Attach / Detach** toggle button if the corresponding text indicates *No System*.
4. In the **Attach System** dialog, navigate to the required object in Functional Structure and click **OK** to attach this start object.

Copy/paste or move Function Diagrams between different projects is not supported through Bulk Data Manger.

Refer to Configuration, for detailed information with examples on the usage of the four worksheets (Copy FuD, Allocate FuD, Generate Code, Typical) of the workbook.

Verifying Diagram States

Function Diagrams reflect different diagram states as displayed in a traffic light symbol in the status bar of Function Designer window.

Table 1. Function Diagram States

State	Description
	**Unallocated** The function has not been assigned to any application in the Control Builder. Any Control Modules or Function Block that are part of this function, are placed in the 'Unallocated_Inst_App' folder of the Control Builder. The Single Control Modules that will contain the code of this function are either empty or do not yet exist. Modification of the diagram will not cause a state change. This is the initial state if a new function is created.
	**Modified** Diagram has been altered (compared to what is in the Control Builder). There are two ways the user can get a 'Modified' diagram: 1) User can change the data in the Function Designer, but she/he does not generate new code for the Control Builder. 2) A change in Plant Explorer or in Control Builder has updated the Function Diagram, but requires that the code needs to be re-generated. Example: if a Function Block is renamed, then the corresponding invocation in generated code block needs to be updated as well
	**Outdated MMS Cross Communication Data *** Diagrams created by automatic MMS cross communication can be outdated due to modified allocation or modified diagram references/connections over application boundaries. In that case, user must rerun MMS cross communication analysis and generation.

Table 1. Function Diagram States (Continued)

State	Description
	**Generated** Configuration data for Control Builder was successfully generated. The data in Control Builder corresponds to the data in Function Designer. Function Designer also invokes the IO Allocation tool for all IO signals on the diagram that are placed in Control Structure under a hardware unit. By that, the IO signals are allocated and the IO signal data in Control Builder corresponds to the IO signal properties. The diagram is now ready to be loaded into the controller.
	**Generated but IO not allocated** The signals are not allocated or assigned to hardware.
	**Generated but Outdated MMS Cross Communication Data *** Diagrams created by automatic MMS cross communication can be outdated due to modified allocation or modified diagram references/connections over application boundaries. In that case, user must rerun MMS cross communication analysis and generation, see Outdated MMS Cross Communication Data.
	**Generated but IO not allocated and Outdated MMS Cross Communication Data *** The configuration data generation is completed but the signals are not allocated or assigned to hardware and the MMS cross communication is outdated.
	**Loaded** The application that contains the Diagram is loaded into the Controller. If a diagram is in this state, user can perform online operations like display, force, and tuning.
	**Loaded but IO not allocated** The application that contains the Diagram is loaded into the Controller but at least one IO signal is not allocated.

Table 1. Function Diagram States (Continued)

State	Description
	**Loaded but Outdated MMS Cross Communication Data *** The application that contains the Diagram is loaded into the controller but at least one MMS cross communication connection is outdated.
	**Loaded but IO not allocated and Outdated MMS Cross Communication Data *** The application that contains the Diagram is loaded into the controller but at least one IO signal is not allocated and at least one MMS cross communication connection is outdated.
	**Unknown due to bad OPC quality** The state cannot be determined, because no data is received from OPC server.

* - Applicable only for SCM related Function Diagrams

Types of Sequences

Function Designer supports the design of sequences with steps, transitions, and actions. Multiple sequences are possible within a single Function Diagram. The data for the sequences are stored in the Function aspect. A sequence can be inserted in a Function Diagram or in a Function Diagram type. Use any of the following sequence components:

- IEC 61131-3 Sequence.
- Two direction sequence (Sequence2D).

To start the SFCViewer from the faceplate (SFC2DHeader) for a sequence2D, the user must:

1. Allot the fixed name **SFC**, to the Sequence2D object.

2. Assign the diagram name to the **ExtSFCObjectName** port of SFC2DHeader.

In a Function Diagram, only one Sequence2D can exist with the fixed name **SFC**, refer Figure 11.

IEC 61131-3 Sequence

IEC 61131-3 **Sequence** consists of a default step which follows a single-line chain as shown in Figure 9. When a sequence is inserted in a Function Diagram, the sequence maps to a SFC codeblock in Control Builder which is created by Configuration Data Generation. A new overview diagram for the sequence is added to the Function aspect. The ports available on the sequence component correspond to implicit variables generated by the Control Builder compiler. These ports can be used in the Function Diagram to connect logic and to control execution of the sequence.

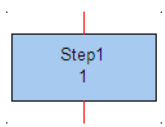


Figure 9. IEC 61131-3 Sequence Process Flow

Sequence2D

Sequence2D consists of steps and transitions which follows a two-line chain as shown in Figure 10.

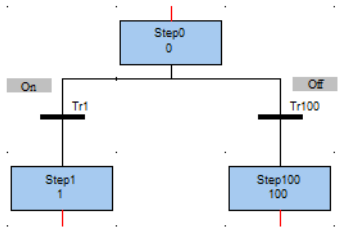


Figure 10. Sequence2D Process Flow

The user must connect a control module of type SFC2DHeader to the sequence as shown in Figure 11. This control module provides access to the related faceplate.

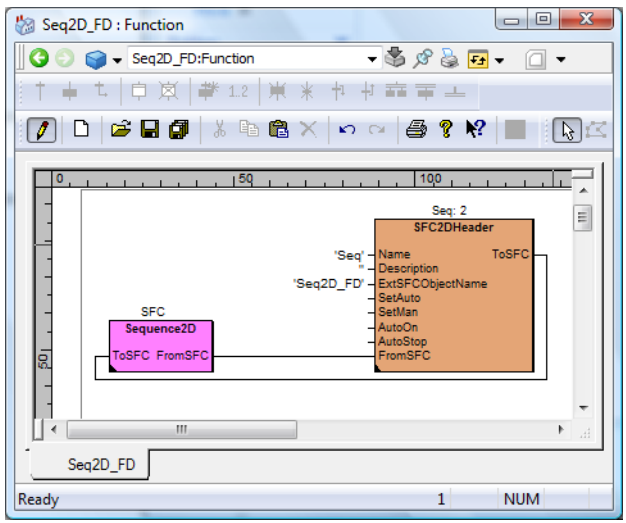


Figure 11. Sequence2D to SFC2DHeader Connection

Sequence2D is based on SeqStartLib library. For more information, refer to the 800xA Control Builder online help.

Sequence2DHeader is available in the following path: Object Types\Control System\AC 800M/C Connect\Libraries\SeqStartLib\Control Module Type\SFC2DHeader

User defined or customized SFC2DHeader is not supported. If the SFC2DHeader is used then it would display an error message *Syntax Error in XML* while performing Configuration Data Generation. SFC2DHeader is supported only for Function Diagram created in Single Control Module (SCM). For more information, refer to *System 800xA Engineering, Engineering Studio Function Designer (3BDS011224*)*

For information regarding SFC Overview, refer to *System 800xA Engineering, Engineering Studio Function Designer (3BDS011224*).*

Overview Diagram

For a sequence a grid-structured overview diagram is supported. Each cell or a consecutive pair of cells in a column of the grid of a sequence overview diagram can contain one of the following elements:

Table 2. Elements of an Overview Diagram

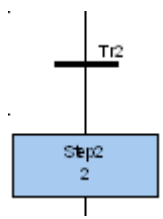
Cell Contents	Description	Symbol
Empty	No meaning.	-
Normal transition with step.	Two consecutive cells are used.The transition cell is related to a transition detail diagram containing a transition symbol and transition logic. The step cell is related to a step detail diagram containing a step detail symbol and optional action diagrams or diagram references.	

Table 2. Elements of an Overview Diagram (Continued)

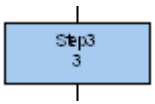
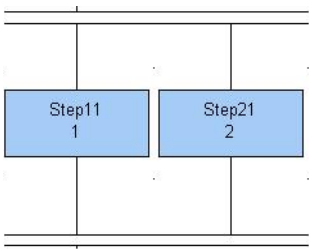
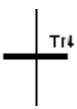
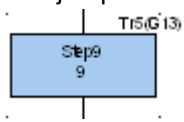
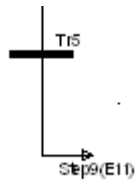
Cell Contents	Description	Symbol
Step only	Required as starting step for simultaneous branches. This cell is related to a detail diagram containing a step symbol and optional action diagrams or a diagram reference, but no transition logic and no transition symbol.	
Simultaneous Sequence	The divergence and convergence of simultaneous sequences is indicated by a double horizontal line. Only one common transition symbol is possible, above and below the double horizontal line of synchronization. **Note:** 1. It is possible to use only up to eight simultaneous sequence branches.	
Transition only	Required as last transition for alternative branches. This cell is related to a detail diagram containing transition logic and a transition symbol, but no step symbol and optional action diagrams.	

Table 2. Elements of an Overview Diagram (Continued)

Cell Contents	Description	Symbol
Transition with jump	Two consecutive cells are used. The first cell is related to a detail diagram containing transition logic and a transition symbol, but no step symbol and optional action diagrams. The jump symbol in the second cell defines the step name to jump to. A step, which is a jump target, displays an additional reference to the jump.  **Note:** 1. User must connect a jump to a step only through a transition. Also a jump can be connected to only a single step. 2. Jump can be used after a parallel branch, only if connected through a step.	

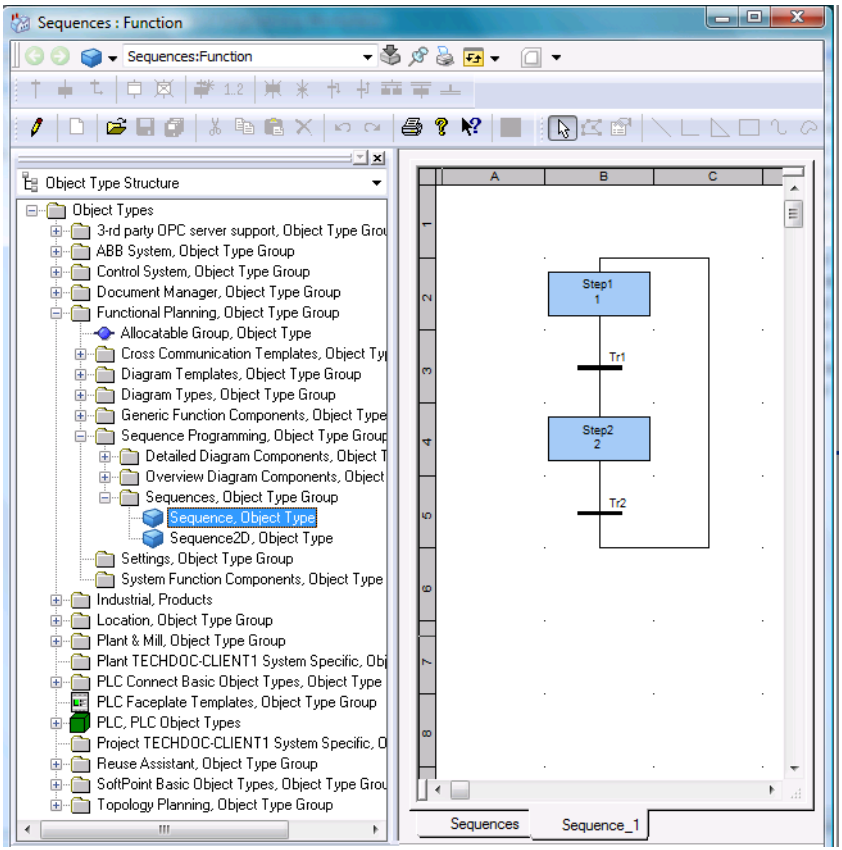


Figure 12. Sequence Overview Diagram

In Function Designer, a simultaneous sequence can be connected to a previous step only through a transition and jump.

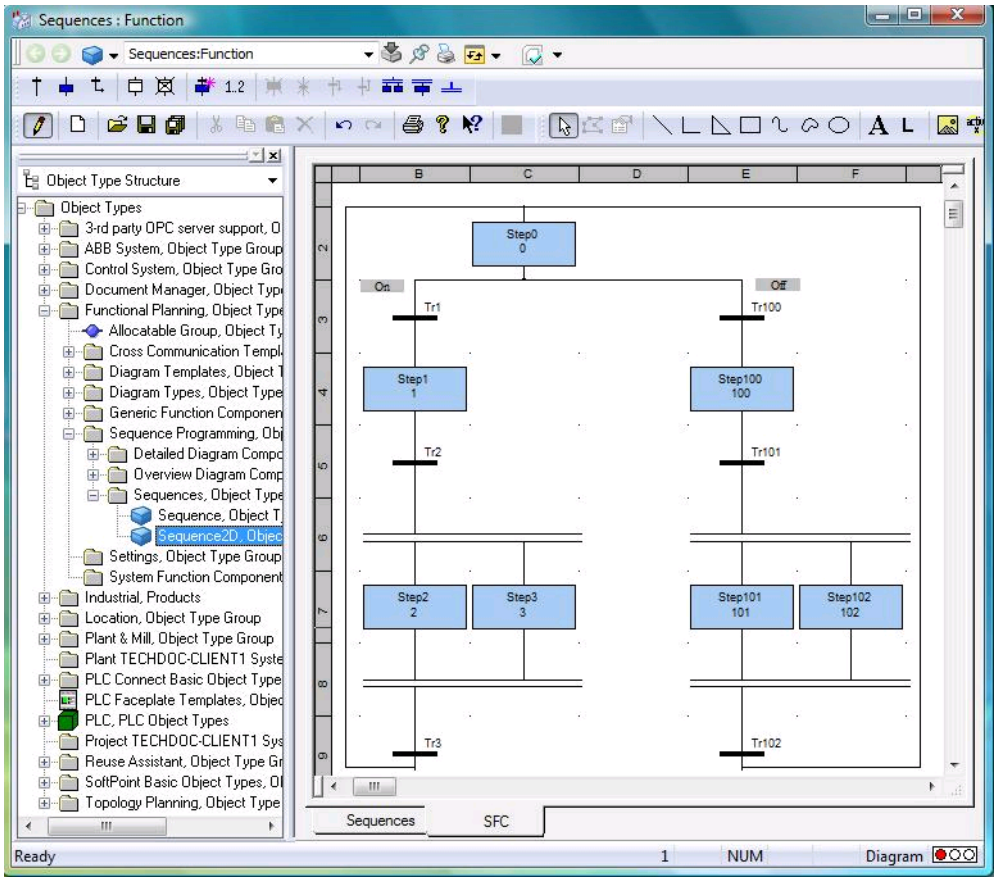


Figure 13. Sequence2D Overview Diagram

Inserting a Sequence\Sequence2D Component

Follow the steps to insert a IEC 61131-3 sequence\sequence2D in a Function Diagram:

1. Open the Function Diagram, example 1TT2SeqCtrl. Enlarge the popup window to full size.
2. Click **Insert** > **Object...** menu item.

3. In the **SPL** tab, select the required Sequence/Sequence2D component.

The Sequence2D object type is also available at Sequence Functional Component Object Type in Object Type Structure: Object Types\Functional Planning\Sequence Programming\Sequences\Sequence2D.

4. Enter a valid name (for example, 1TT2SeqCtrl_Seq1) and description, in the **New Component Name** dialog. Keep the default value for the Data Flow Order.
5. Click **OK**.
   The graphical block symbol for this Function Component appears on the diagram with minimum ports. By default, most of the ports are hidden.

To unhide the ports:

a. Right-click the required symbol and click **Show Hidden Ports...** from the available context menu.

b. In the **Show Hidden Ports** dialog, select the required ports which must be made visible or select the **All Hidden Ports** check box.

c. Click **OK**.

Alternative to Step 3:

1. Include Structure Browser 1 and ensure that **Object Type Structure** is shown.
2. Navigate to the Sequence Functional Component Object Type in Object Type Structure: *Object Types\Functional Planning\Sequence Programming\Sequences\Sequence*.
3. Click the required object type (Sequence\Sequence2D), keep the mouse button pressed and drag the object type on to the Function Diagram drawing area, then release the mouse button.

In a sequence, the initial step is defined by enabling **InitialStep** in its **Aspect Properties** dialog.

Editing Overview Diagram

Follow the steps to open and edit the overview diagram of a sequence:

1. Right-click the sequence symbol.

2. Click **Goto Child Diagram** from the available context menu. The grid-structured overview diagram appears and is shown as a workbook tab (if workbook mode, as given by default, is switched on). By default, a first step symbol is inserted.
3. Click the Step1 symbol and drag it into another cell, for example B2.
4. To rename Step1 to Fill:
   a. Right-click the step symbol.
   b. Click **Aspect Properties...**.
   c. Edit the name available in the **Component** tab of the **Aspect Properties** dialog and click **OK**.
5. To add the transition to follow step Fill:
   a. Select the required port.
   b. Right-click and select **Insert Transition** from the available context menu.
   c. In the **New Component Name** dialog enter transition name, for example Filled, Number and Description or accept the available defaults.
   d. Press ESC to quit insertion mode.
6. To add the next step to follow the transition Filled:
   a. Select the required port.
   b. Right-click and select **Insert Step** from the available context menu.
   c. In the **New Component Name** dialog, provide Name, for example Agitate, Number and Description or accept the available defaults.
   d. Press ESC to quit insertion mode.
7. Repeat Step 5 and Step 6 to add a transition Agitated and a step Drain and repeat Step 5 to add a final transition Drained.
8. Connect the output port of each step to the input port of the next transition and follow the steps to connect the output port of the final transition to the input port of the first step:
   a. Click the output port, keep the mouse button pressed.

b. Drag the mouse pointer to the input port.

c. Release the mouse button.

9. Click **Save**.

Simultaneous sequence branches must begin and end with a transition.

In SPL Transition diagram the connection lines become red when incompatible data types are connected to the input and output ports. For example, the connection lines become red when the input diagram reference of type **RealIO** is connected to a port of type **Real**.

It is recommended not to copy/paste configured steps/transitions/actions across the same sequence. If such configured steps/transitions/actions are copy/pasted, the connection links needs to be renamed.

As an alternative to editing single steps and transitions, user can use the **Create multiple transitions/steps** button of the sequences toolbar.

As an alternative to connecting steps and transitions by mouse, right-click a step, a transition, or a jump and select **Goto/Connect...**, then connect using the **Goto/Connect to** dialog. Use this especially when connecting a transition with a jump to a step.

Editing Step Detail Diagram

Add action code to a step detail diagram, as in a normal Function Diagram.

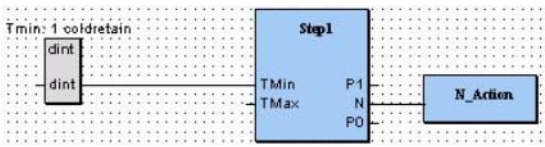
A step has three outputs of data type Boolean to initiate corresponding actions:

N, P1, and P0, to initiate N Action, P1 Action, and P0 Action respectively.

- N action qualifier stands for Non-stored, and means that the action code is executed continuously, while the step is active.
  Output N is true while the step is active.
- P1 action qualifier stands for Pulse rising edge, and means that the action code is executed once, when the step becomes active.
  Output P1 is true when the step becomes active.
- P0 action qualifier stands for Pulse falling edge, and means that the action code is executed once, when the active step becomes inactive.
  Output P0 is true when the step becomes inactive.

For each step, there can be zero or one associated P1 action, zero or one associated N action and zero or one associated P0 action.

Connecting Diagram parameters to TMin or TMax of a step causes inconsistency in the sequence, hence add local variables instead of Diagram parameters.



Right-click and select **New Action** from the respective context menu to add action to P0 or P1 or N.

In SPL based diagrams, connecting an Output Diagram Reference having a Communication Variable to the Step Output port of a sequence, results in an error during configuration data generation.

In the simplest case the output N is connected to a variable or diagram reference.

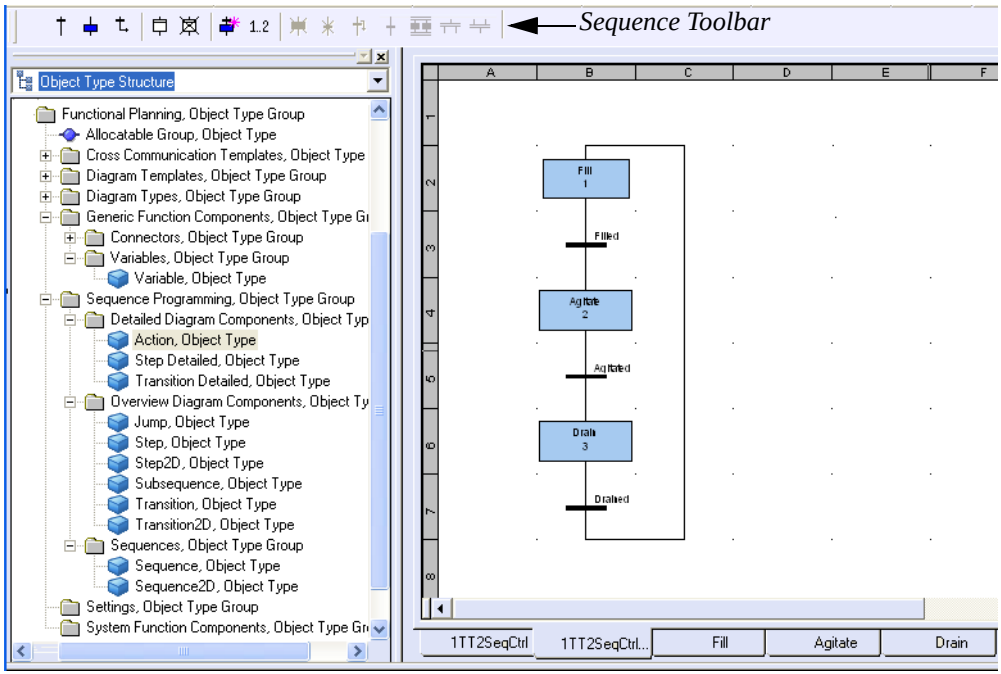


Figure 14. Sequence Overview Diagram, Sequence Toolbar

Diagram references can be connected to Communication Variables. Action code is generated into CBM Structured Text (ST) code.

Limitations:
In all kinds of detailed diagrams implicit type casts are not allowed. For action code in detail diagrams, only the following components are allowed:

- Diagram Reference.
- Variable.
- Action.

To create and edit a detail diagram of a step:

1. Double-click the corresponding step symbol in the overview diagram.
   A diagram named according to the step, appears as a further workbook tab. By default, it contains the step symbol.
2. Edit the diagram by inserting and connecting diagram references, or variable components.
3. Or **Insert** > **Object...** and connect the Action diagram component from Object Type Structure: Object Types\Functional Planning\Sequence Programming\Detailed Diagram Components\Action.
   Within these diagrams, the user can create logic for the different step phases.

Data flow order within a detail diagram is not supported and is not displayed. This order is calculated automatically and cannot be modified.

If diagram references and diagram variables are added within a step or transition, the following default naming convention is used:

- SequenceName_StepName_Action_PortType
- SequenceName_Transition_PortType

Editing Action Diagram

User can add action code in an action diagram, as in a normal Function Diagram. Action code is generated into CBM Structured Text (ST) code. Action diagrams are sub-diagrams.

Limitations:
Maximum number of action blocks per step is three. Also, the action diagrams do not support the following:

- Aspect Objects.
- CBM_Signals.
- Control Modules.

To create and edit an action diagram:

1. Double-click an action diagram component inserted into a step detail diagram (refer to Editing Step Detail Diagram). An empty Function Diagram named according to the step and action component (for example Fill_Action) appears.
2. In this Function Diagram user can edit the control logic for this action, for example the N action of step Fill.
   The action code has the same restrictions when the it is edited in the parent step detail diagram.

Connection to Step output can be made through N Port of Step block and not through N action diagram.

Editing Transition Detail Diagram

Add transition logic in a transition detail diagram, as in a normal Function Diagram.

Limitations:
In the detail diagram graphical symbols are used for the transition logic, but the allowed set of components is limited and an error message appears informing the user that the selected blocks are not supported, due to the fact that transition logic results in a Boolean expression, which is defined in Control Builder M as ST code.

To create and edit a detail diagram of a transition:

1. Double-click the corresponding transition symbol in the overview diagram.
   A diagram named according to the transition appears as a further workbook tab. By default, it contains the transition symbol.
2. Insert and connect allowed components from Object Type Structure: Object Types\Control System\AC 800M/C Connect\System Functions\Functions.
   If the user tries to insert an incompatible component, an error message appears.

SPL does not support error navigation.

Deleting a Sequence

To delete a sequence, select and delete the related component in the main diagram. The corresponding overview diagram and the detail diagrams are also deleted automatically.

Navigating Between the Sequence Diagrams

User can navigate between different diagrams by any of the following methods:

- Right-click the menubar/toolbar area of the Function Diagram and click **SFC OverView** from the available context menu. A sequence tree structure appears, and user can click the desired sequence component to access it directly.

For information regarding **SFC OverView** menu option, refer to *System 800xA Engineering, Engineering Studio Function Designer (3BDS011224*).*

- Use the workbook tabs in the set of the opened diagrams.
- Right-click the blank space of a diagram and select **Goto Parent Diagram** to switch to the parent diagram.
- Right-click a component on the diagram representing a child diagram and select **Goto Child Diagram** to switch to the child diagram.
- Right-click the diagram reference and select **Goto Reference**.

To achieve maximum performance in a Function Diagram of more than 2MB size, it is recommended to have:

- Less than 60 steps per SFC.
- Less than 60 transitions per SFC and per step.

Allocating, Generating Configuration Data, and Testing

The complete sequence is available in the Function Diagram. Allocate, generate and download configuration data, and test the configuration of the Function Diagram containing the sequence as described in earlier sections.

Additional test functions available in the sequences toolbar are:

- Enables / Disables execution of the actions associated to the step.
- Enables / Disables state transition.
- Forces next step to become active.
- Forces previous step to become active.

These test functions are also available as a context menu on a step or transition.

Printing Documentation

Ensure the following while printing sequences:

1. The aspect property "**Number**" of a Step should always be an integer.
2. The aspect property **Number** of each Step should be entered in ascending order. The printing order of sequences depend on this aspect property. Set a value for this property to match the required printing order of the sequence. The aspect property **Number** of a Transition does not have any effect on the printing order of the sequence.

Use normal Function Diagram print functions to print sequences.

Step detail and transition detail are printed together in one diagram containing the transition with the transition logic and the step with the action code.

Section 3 Configuration

This section describes how to use Function Designer and Bulk Data Manager to configure process functions or process sequences required to implement an application based on a given process model.

Process Model

A process model results from plant analysis and design. It can be formulated as a process flow diagram, as shown in figure below or as a sequence flow diagram as shown in figure in Process Sequence Model and additional descriptions (refer to Table 3table below).

The process model used as an example describes a tank with two control loops, two valves, an agitator, and a level monitoring. It can be implemented based on control module types and function block types from the AC 800M Connect standard object type libraries and based on a project specific object type.

The process model used here is simplified and serves as a thread through the tutorial. The tutorial only shows implementation of some parts of it. Names not compliant to IEC 61131 are used to show name mapping between Function Designer and Control Builder M.

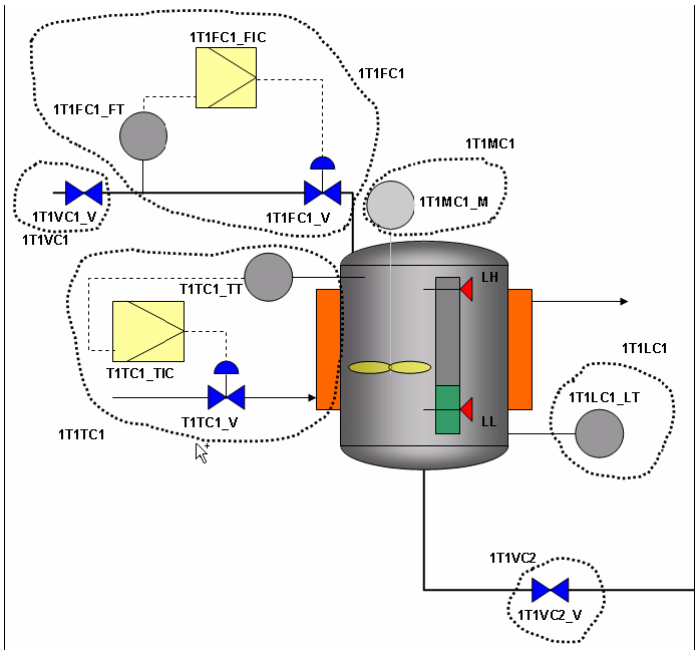


Figure 15. Process Flow Diagram Tank1

Table 3. Process Functions Tank1

Function		Object		Object Type (Library)
Name	**Description**	**Tag Name**	**Description**	
1T1VC1	Filling valve control	1T1VC1_V	Inlet block valve	ValveUni (ProcessObjExtLib)
1T1FC1 Filling flow control		1T1FC1_FT	Flow transmitter	AnalogInCC (Control - StandardLib)
		1T1FC1_V	Flow control valve	AnalogOutCC (Control - StandardLib)
		1T1FC1_FIC	Flow controller	PidCC (Control - StandardLib)

Table 3. Process Functions Tank1 (Continued)

Function		Object		Object Type (Library)
Name	**Description**	**Tag Name**	**Description**	
1T1VC2	Draining valve control	1T1VC2_V	Outlet block valve	ValveUni (ProcessObjExtLib)
1T1TC1 Heating temperature control		1T1TC1_TT	Temperature transmitter	AnalogInCC (Control-StandardLib)
		1T1TC1_V	Temperature control valve	AnalogOutCC (Control-StandardLib)
		1T1TC1_TIC	Temperature controller	PidCC (Control-StandardLib)
1T1MC1	Agitator motor control	1T1MC1_M	Agitator	AgitatorUni (Own library)
1T1LC1	Level control	1T1LC1_LT	Level transmitter	SignalInReal (SignalLib)

Configuring Process Functions

Follow the steps to implement the process functions identified and listed in he table above:

1. Create a project in the Control Structure and include the necessary controllers and I/O hardware units.
2. If necessary, create parent objects in the Functional Structure, and insert Function Diagrams to the objects.
3. Configure these Function Diagrams individually, or use an existing configured diagram as a typical for copying, according to the requirement.
4. If required, create user-defined object types in the Object Type Structure. Include them in the user-defined object type libraries in the Library Structure and use them in the Function Diagrams.

5. Allocate the process functions:
   a. Allocate the Function Diagrams to the required applications.
   b. Allocate the signals to the compatible I/O boards.
6. Generate configuration data.
7. Download the process functions to the controller and test them.
8. Print the documentation.

Preparing Control Structure

User must create a project and open Control Builder M on this project before working on Function Designer.

Extending the project with I/O board hardware units can be done at any time before the user allocates the signals.

Creating Control Project

Creating Control Network object:

Follow the steps to create a Control network object:

1. In Engineering Workplace, right-click the root object of Control Structure and click **New Object...** from the context menu.
2. In the Object Type Structure tree of the New Object dialog navigate to Object Types\Control System\AC 800M/C Connect\Control Types.
3. Click **Control Network**.
4. Accept or change the **Name** and enter an object **Description**.
5. Click **Create** to create the Control Network object.

Configuring Control Network Object:

Follow the steps to configure a Control Network object:

1. Select the Control Network object in Control Structure.
2. Right-click the **OPC Data Source Definition** aspect and click **Config View.**

3. In the Connectivity tab of the **<control network name>:OPC Data Source Definition** dialog click **New...**.
4. In the **New Service Group** dialog enter a unique service group name (or accept the default if it fits).
5. In the **Add Service Provider** dialog select the computer name of the connectivity server and click **OK**.
6. In the **New Service Group** dialog click **OK**.
7. In the **OPC Data Source Definition** dialog click **Apply**.
8. To exit the dialog click **Close** on window level.

Creating Project Object:

Follow the steps to create a project object:

1. In Engineering Workplace, right-click the created Control Network object of Control Structure to access the context menu and click **New Object...**.
2. Click AC800M from the list of offered project template object types.
3. Accept or change the **Name** and enter an object **Description**.
4. Click **Create** to create the project object.

Alternatively user can create a new project from within Control Builder M.

Creating and Connecting Applications and Tasks

By default, the project template used above includes an application connected to a task of the included Controller.

If required additional applications, controllers and tasks can be created. Each application has to be connected to a controller and also to a task of the controller.

For more information, refer to *System 800xA Configuration (3BDS011222*)* and *System 800xA Control AC 800M Getting Started (3BSE041880*)*.

Extending Control Structure

Follow the sub-sections to insert a hardware library into the project, to connect this hardware library to the controller and to extend the hardware sub-tree with I/O board objects from this hardware library:

Inserting Hardware Library into Project:

Follow the steps to insert a hardware library into a project:

1. In Engineering Workplace, select the project object in Control Structure.
2. Right-click the Project aspect and click **Config View** from the list of aspects.
3. In the **Hardware Libraries** tab click **Insert**.
4. In the **Select a Library** dialog navigate to and click the required hardware library, for example S800IoModulebusHwLib x.x-x to provide S800 I/O board hardware unit types in this project.
5. Click **OK**.
6. Click **Apply**.

Connecting Hardware Library to Controller:

Follow the steps to connect a hardware library to controller:

1. In Engineering Workplace, navigate to the controller object in Control Network\Project\Controllers of the Control Structure.
2. Right-click the Controller aspect and click **Config View** from the context menu.
3. In the **Libraries** tab click **Connect**.
4. In the **Select a Library to Connect** dialog click the required hardware library, for example S800IoModulebusHwLib x.x-x to provide S800 I/O board hardware unit types in this controller.
5. Click **OK**.

Creating I/O Board Hardware Unit:

1. In Engineering Workplace, right-click the ModuleBus object below the controller in the Control Structure and click **New Object...** from the context menu.
2. In the Object Type Structure tree of the **New Object** dialog navigate to Object Types\Control System\AC 800M/C Connect\Libraries\Hardware.
3. Click the required board object type, for example AI810.
4. Accept or change the **Name** and enter an **Description**.
5. Click **Create** to create the I/O board object.

Repeat steps 1 to 5 to create all required I/O board hardware units.

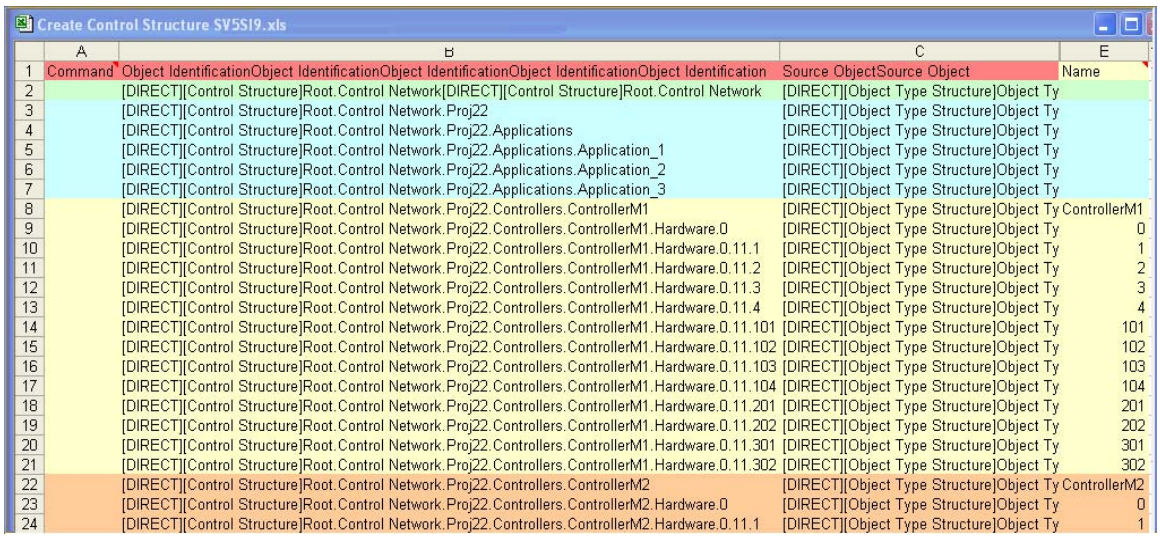
Alternatively user can create new I/O board hardware units repeatedly from within Control Builder M.

Building the Control Structure with Bulk Data Manager

In a real plant many controllers and I/O boards are needed. It is profitable to configure a Bulk Data Manager excel worksheet similar to the figure below, which creates the project, the controllers and the I/O board hardware units by a single action.

It is always recommended to click **Enable Macros** in the **Security Warning** dialog that appears on opening a Bulk Data Manager excel worksheet.

For more information on how to configure such a worksheet, refer to *System 800xA Engineering, Engineering Studio Function Designer (3BDS011224*).*



Create Control Structure SV5SI9.xls

	A	B	C	E
1	Command	Object IdentificationObject IdentificationObject IdentificationObject IdentificationObject Identification	Source ObjectSource Object	Name
2		[DIRECT][Control Structure]Root.Control Network[DIRECT][Control Structure]Root.Control Network	[DIRECT][Object Type Structure]Object Ty	
3		[DIRECT][Control Structure]Root.Control Network.Proj22	[DIRECT][Object Type Structure]Object Ty	
4		[DIRECT][Control Structure]Root.Control Network.Proj22.Applications	[DIRECT][Object Type Structure]Object Ty	
5		[DIRECT][Control Structure]Root.Control Network.Proj22.Applications.Application_1	[DIRECT][Object Type Structure]Object Ty	
6		[DIRECT][Control Structure]Root.Control Network.Proj22.Applications.Application_2	[DIRECT][Object Type Structure]Object Ty	
7		[DIRECT][Control Structure]Root.Control Network.Proj22.Applications.Application_3	[DIRECT][Object Type Structure]Object Ty	
8		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM1	[DIRECT][Object Type Structure]Object Ty	ControllerM1
9		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM1.Hardware.0	[DIRECT][Object Type Structure]Object Ty	0
10		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM1.Hardware.0.11.1	[DIRECT][Object Type Structure]Object Ty	1
11		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM1.Hardware.0.11.2	[DIRECT][Object Type Structure]Object Ty	2
12		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM1.Hardware.0.11.3	[DIRECT][Object Type Structure]Object Ty	3
13		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM1.Hardware.0.11.4	[DIRECT][Object Type Structure]Object Ty	4
14		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM1.Hardware.0.11.101	[DIRECT][Object Type Structure]Object Ty	101
15		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM1.Hardware.0.11.102	[DIRECT][Object Type Structure]Object Ty	102
16		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM1.Hardware.0.11.103	[DIRECT][Object Type Structure]Object Ty	103
17		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM1.Hardware.0.11.104	[DIRECT][Object Type Structure]Object Ty	104
18		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM1.Hardware.0.11.201	[DIRECT][Object Type Structure]Object Ty	201
19		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM1.Hardware.0.11.202	[DIRECT][Object Type Structure]Object Ty	202
20		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM1.Hardware.0.11.301	[DIRECT][Object Type Structure]Object Ty	301
21		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM1.Hardware.0.11.302	[DIRECT][Object Type Structure]Object Ty	302
22		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM2	[DIRECT][Object Type Structure]Object Ty	ControllerM2
23		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM2.Hardware.0	[DIRECT][Object Type Structure]Object Ty	0
24		[DIRECT][Control Structure]Root.Control Network.Proj22.Controllers.ControllerM2.Hardware.0.11.1	[DIRECT][Object Type Structure]Object Ty	1

Figure 16. BDM Worksheet to Create a Project

Configuring Functional Structure

Creating Upper Level Objects

The upper level objects model the plant above the detail process functions implemented by Function Diagrams.

Creating First Level:

1. In Engineering Workplace, right-click the root object of Functional Structure and click **New Object...** from the context menu.
2. In the Object Type Structure tree of the **New Object** dialog navigate to Object Types\Plant & Mill\Sites and Areas.
3. Click the Site object type.
4. Edit the **Name**, for example Site1, and enter an object **Description**.
5. Click **Create** to create the site object.

Creating Second Level:

1. In Engineering Workplace, right-click the site object in Functional Structure and click **New Object...** from the context menu.
2. In the Object Type Structure tree of the **New Object** dialog navigate to Object Types\Plant & Mill\Sites and Areas.
3. Click the Area object type.
4. Edit the **Name**, for example Area1, and enter an object **Description**.
5. Click **Create** to create the area object.

Creating Third Level:

This object represents the process function Tank1 and serves as the parent object for the objects holding the Function Diagrams:

1. In Engineering Workplace, right-click the site object in Functional Structure and click **New Object...** from the context menu.
2. In the Object Type Structure tree of the **New Object** dialog navigate to Object Types\Plant & Mill\Process Cells.
3. Click the **Liquid Processing Cell** object type.
4. Edit the **Name**, for example 1T1, and enter an object **Description**.
5. Click **Create** to create the liquid processing cell object 1T1.

Creating Objects for Function Diagrams

Using the liquid processing cell object 1T1 as parent object create an object with Name 1T1FC1 and Object description Flow Control according to the steps described in Creating a new Function Diagram.

Repeat these steps for other objects on this level, except those intended to be created by copying object typicals (1T1VC1 and 1T1VC2), resulting in the subtree shown in the figure below.

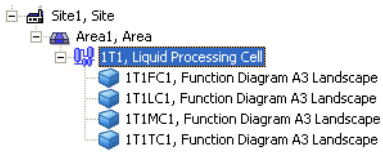


Figure 17. Initially Configured Functional Structure

Building the Functional Structure with Bulk Data Management

In a real plant many objects are needed to model the plant in the Functional Structure. It is profitable to configure a Bulk Data Manager Excel worksheet which creates the upper level objects of the Functional Structure until including the objects holding the Function Diagrams, except those intended to be created by copying object typicals, by a single action.

For information on how to configure such a worksheet, refer to *System 800xA Engineering, Engineering Studio Function Designer (3BDS011224*)*.

Configuring Function Diagrams

To configure the flow control loop 1T1FC1:

1. Double-click the Function aspect of object 1T1FC1 to open the Function Diagram.
2. Insert the required Function Components given in the table below into the Function Diagram according to Inserting Function Components and connect them according to Connecting Function Components.

Table 4. Function Components 1T1FC1

Function Component Object Type	Name	Description	Additional Action
Object Types\Control System \AC 800M/C Connect\Libraries \ControlStandardLib 1.4-4 \Control Module Types\PidCC	1T1FC1 _FIC	Flow controller	
Object Types\Control System \AC 800M/C Connect\Libraries \ControlStandardLib 1.4-4 \Control Module Types\AnalogInCC	1T1FC1 _FT	Flow transmitter	a) Connect port Out to port Pv of 1T1FC1_FIC
Object Types\Control System \AC 800M/C Connect\Libraries \ControlStandardLib 1.4-4 \Control Module Types\AnalogOutCC	1T1FC1_V	Flow control valve	a) Connect port In to port Out of 1T1FC1_FIC
Object Types\Control System \AC 800M/C Connect \CBM_Signals\CBM_AIS	1T1FC1_In	Input flow control	a) Connect to port AnalogInput of 1T1FC1_FT
Object Types\Control System \AC 800M/C Connect \CBM_Signals\CBM_AOS	1T1FC1 _Out	Output flow control	a) Connect to port AnalogOutput of 1T1FC1_V

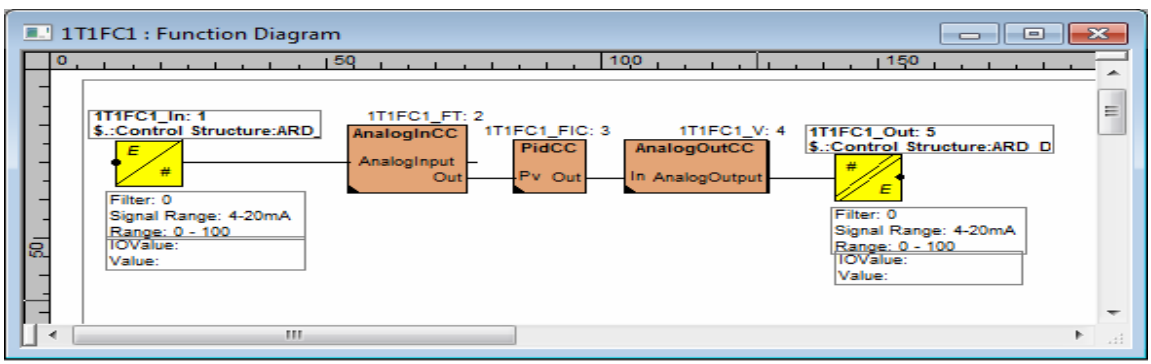


Figure 18. Function Diagram 1T1FC1

The control library versions listed in Table 4 and several other tables in this manual are just examples. Use the latest available versions.

3. Configure Data Flow Order according to Configuring Data Flow Order if needed.
4. Allocate the Function Diagram to Application_1 according to Allocating a Function Diagram.
5. Allocate I/O according to Allocating I/O.
6. Test configuration data generation according to Generating Configuration Data.

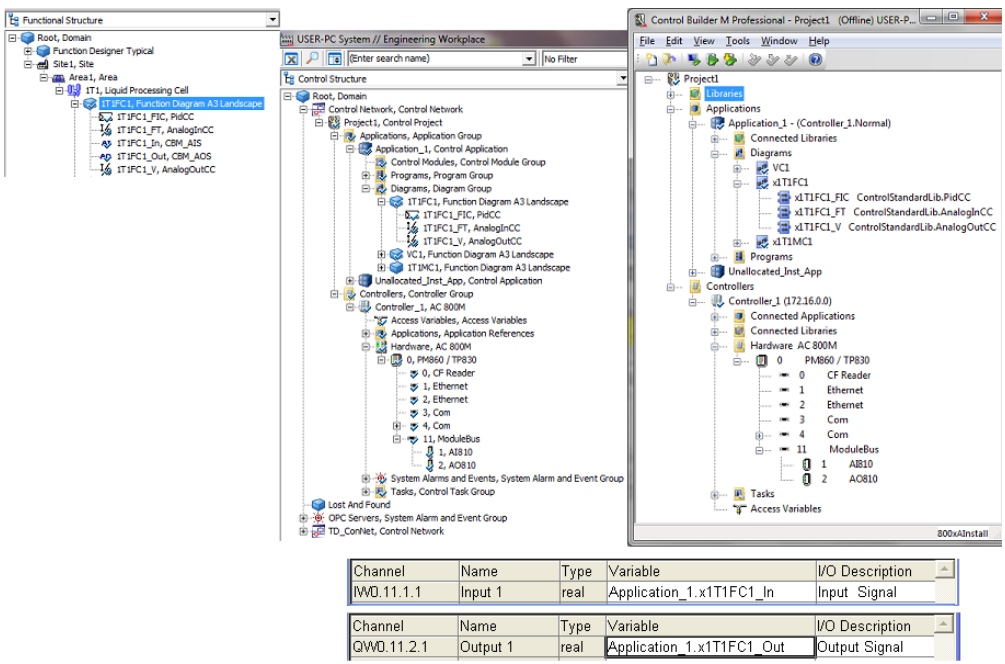


Figure 19. Function Diagram 1T1FC1 in Functional Structure, Control Structure and CBM

Function Diagrams for the analog measurement 1T1LC1 (using Function Component SignalInReal of SignalLib) and the temperature control loop 1T1TC1

(using the same Function Components as for 1T1FC1) can be configured in a similar way.

Working with Typicals

Typicals can be used to implement typical parts of a solution in a project for copying and then adapting them afterwards. In principle every Function Diagram instance can be used as a typical for creating new instances by copying.

Creating a Typical

The valve control function can be configured once and be used as a typical to create the filling and the draining valve control.

1. Under the Root object in the Function Structure create a generic object Function Designer Typical.
2. Below this parent object, create an object of type Function Diagram A3 Landscape, name it VC1 with description Valve Control Typical, see Creating a new Function Diagram.
3. Double-click the Function aspect of object VC1 to open the Function Diagram.
4. Connect the libraries to BasicLib and ProcessObjBasicLib to the Unallocated_Inst_App. Insert the required Function Components given in the table below into the Function Diagram according to Inserting Function Components and connect them according to Connecting Function Components.

In Control Builder M, always browse to **Connected Library** > **Connect Library** and select the latest version of BasicLib.

Table 5. Function Components VC1

Function Component Object Type	Name	Description	Additional Action
Object Types\Control System\AC 800M/C Connect\Libraries\ProcessObjExtLib 2.2-1\Function Block Types\ValveUni	VC1_V1	Inlet block valve	
Object Types\Control System\AC 800M/C Connect\System Functions\Functions\or (bool)	-	-	a) No of inputs: 2 b) Connect to port Ilock1 of VC1_V1
Object Types\Functional Planning\Generic Function Components\Variables\Variable	GSU	Placeholder/ tuning variable	a) Set Variable Properties: Datatype: GroupStartStepConnection Attributes: retain b) Connect to port GroupStartIn of VC1_V1
" " "	IPar	Placeholder/ tuning variable	a) Set Variable Properties: Datatype: UniPar Attributes: retain b) Connect to port InteractionPar of VC1_V1
Object Types\Functional Planning\Generic Function Components\Connectors\Off-Diagram References\Reduced Input Reference	VC1Ilock1	Input diagram ref for interlock	a) Set Variable Properties: Datatype: bool Attributes: retain b) Connect to default Diagram Variable c) Connect to port In1 of or (bool)

Table 5. Function Components VC1 (Continued)

Function Component Object Type	Name	Description	Additional Action
“ “ “	VC1Ilock2	Input diagram ref for interlock	a) Set Variable Properties: Datatype: bool Attributes: retain b) Connect to default Diagram Variable c) Connect to port In2 of or (bool)
Object Types\Functional Planning\Generic Function Components\Connectors\Off-Diagram References\Reduced Output Reference	VC1StatAct	Output diagram ref for interlock	a) Set Variable Properties: Datatype: bool Attributes: retain b) Do not Connect to default Diagram Variable c) Connect to port StatAct of VC1_V1
Object Types\Control System\AC 800M/C Connect\CBM_Signals\CBM_DIS	VC1_FB1	Input FB1	a) Connect to port FB1 of VC1_V1
“ “ “	VC1_FB0	Input FB0	a) Connect to port FB0 of VC1_V1
Object Types\Control System\AC 800M/C Connect\CBM_Signals\CBM_DOS	VC1_Out1	Output Out1	a) Connect to port Out1of VC1_V1

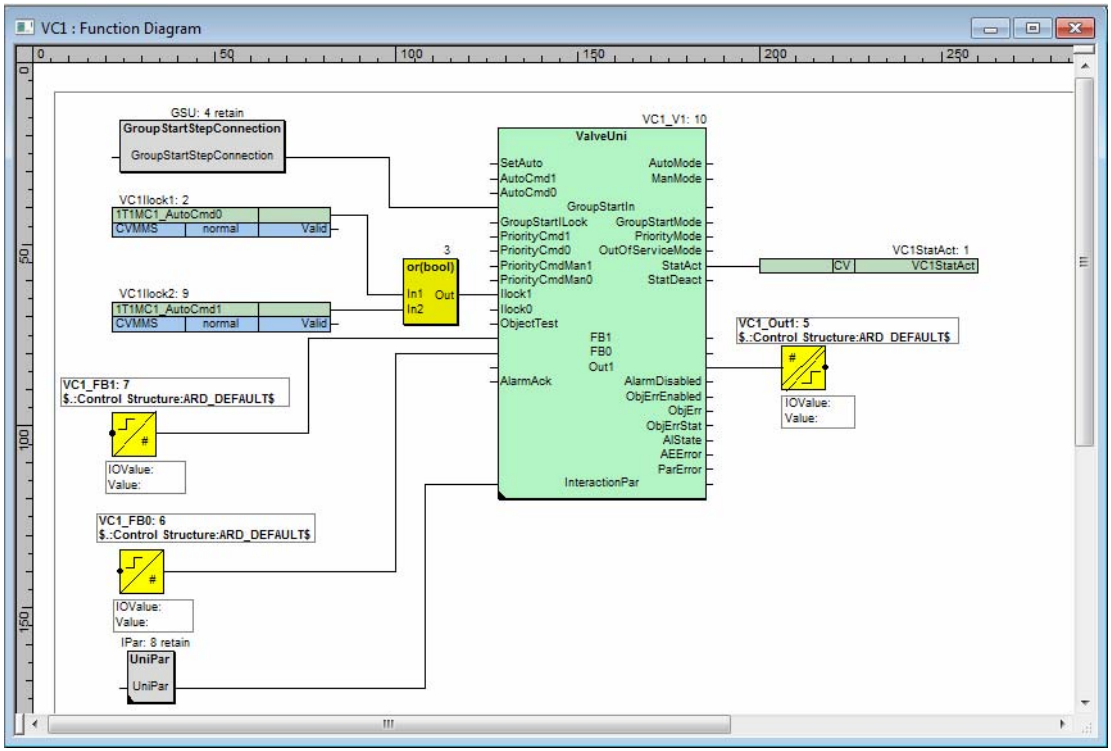


Figure 20. Function Diagram of Typical VC1

5. Configure Data Flow Order according to Configuring Data Flow Order if needed.
6. Do not allocate the Function Diagram to an application as it will be used as a typical for copy actions only.

To test the typical follow the steps given below:

1. Allocate the typical to an application, according to Allocating a Function Diagram.
2. Allocate engineering signals (CBM_Signal objects) of the typical to I/O boards of a Controller, according to Allocating I/O.
3. Generate configuration data, according to Generating Configuration Data.

4. Load the generated configuration data to a controller, according to Downloading Configuration Data.
5. Test the typical, according to Testing Configuration.
6. After testing deallocate the engineering signals:
   a. Right-click on the blank space of the Function Diagram.
   b. Click **Advanced** > **IO Allocation** from the context menu.
   c. In the tree on the left side of the **IO-Allocation** dialog right-click the Boards object.
   d. In the context menu click on **De-allocate**.
7. Unallocate the typical from the application:
   a. Right-click the blank space of the Function Diagram.
   b. Click **Allocatable Group** from the context menu.
   c. In the **Allocatable Group** dialog click **Unallocate**.

Instantiating the Typical

The typical VC1 can be instantiated several times with Bulk Data Management as required.

1. Make sure that in the Function Settings aspect in the Object Type Structure under Object Types\Functional Planning\Settings the value of the property Naming is set to **Pattern** and the value of the property NamePattern is set to *****.

Default Function Setting for Naming = “Pattern”, and NamePattern = “*”.

It is always advisable to provide a single line string for the **NamePattern** field of the **Function Settings** aspect.

2. Open and attach workbook **BDM_for_Function_Diagrams** for start object 1T1 in Function Structure.
3. Drag and Drop the object VC1 from the Functional Structure into the Excel worksheet **Typical**. This gives the user the correct Typical path.
4. Switch to the worksheet **CopyFuD** and configure the first row:

a. Enter name 1T1VC1 into the cell of column **FuD Name**.

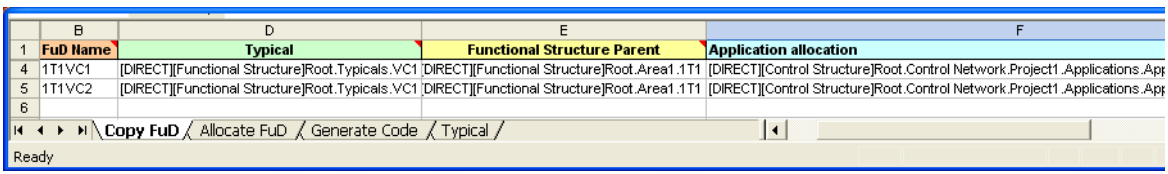
b. Select the cell of column **Typical**. From the offered combo box select the path to typical VC1.

c. Select and right-click the cell of the column **Functional Structure Parent** and click **Insert Object Path...**. Select the Functional Structure and navigate to object 1T1, then click **Object Path**.
If the user has to configure **Application Allocation** during copy, the **Insert Object Path** dialog must not be cancelled.

d. Select the cell of the column **Application Allocation**. In the **Insert Object Path** dialog select Control Structure and navigate to:
..\ Application_1\Control Modules, and click **Object Path**.

e. Click **Cancel** to exit the dialog.

5. Copy the row once as two instances of the typical are required. Change cell FuD Name of the second row to 1T1VC2.



	B	D	E	F
1	FuD Name	Typical	Functional Structure Parent	Application allocation
4	1T1VC1	[DIRECT][Functional Structure]Root.Typicals.VC1	DIRECT][Functional Structure]Root.Area1.1T1	[DIRECT][Control Structure]Root.Control Network.Project1.Applications.App
5	1T1VC2	[DIRECT][Functional Structure]Root.Typicals.VC1	DIRECT][Functional Structure]Root.Area1.1T1	[DIRECT][Control Structure]Root.Control Network.Project1.Applications.App
6				

Copy FuD | Allocate FuD | Generate Code | Typical

Ready

Figure 21. BDM Excel Worksheet to Create two Instances from Typical VC1

6. In the Bulk Data Manager tool bar click **Save all Objects** to create the instances.

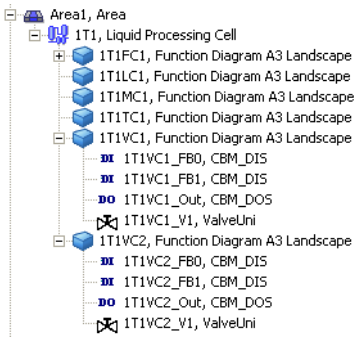


Figure 22. Instantiated 1T1VC1 and 1T1VC2 in Function Structure

7. To check results:

   a. Open the Function aspect of object 1T1VC1 and of 1T1VC2.
   Names in the Function Diagrams are adapted to the new loop name except diagram references and diagram variables which get names derived from VC1.

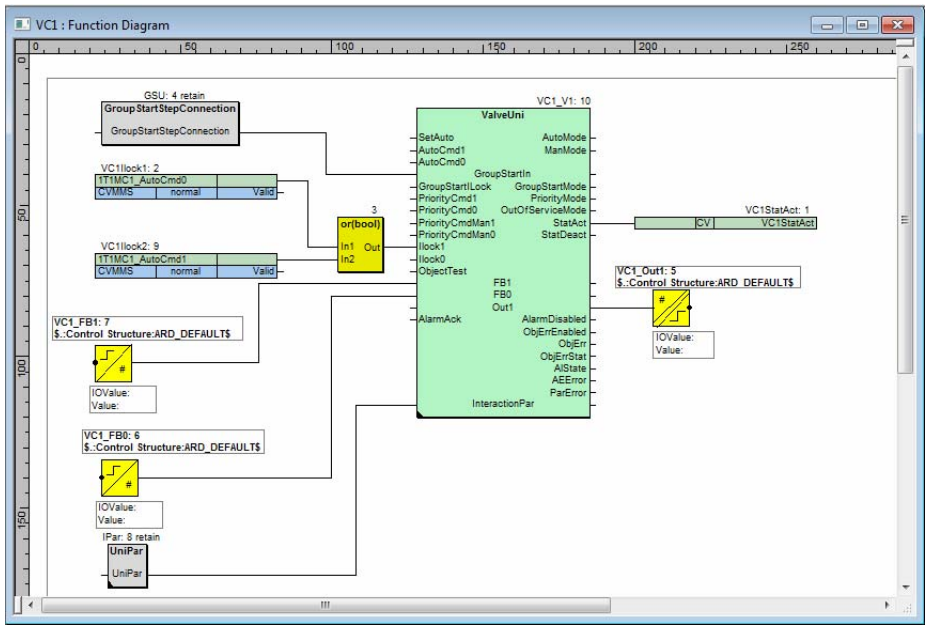


Figure 23. Function Diagram 1T1VC1

b. Generate configuration data (allocation to Application_1 has already been done through the worksheet **CopyFuD**).

Configuring User-defined Object Types

Creating user-defined object types in an user-defined versioned library is used to implement typical solutions that will evolve in a managed way over several versions and several projects.

User should only develop user-defined object types if there are no standard object types available that cover the requirements.

Developing user-defined object types may cause considerable additional effort.

For technical reasons composite object types have to be put into a library, a library extensions or have to belong to a system extension.

If control module types created with a Function Diagram in an user-defined library of an earlier version are restored in 800xA 5.1, then user needs to perform configuration data generation while working with such control module types.

It is always recommended to create Control Module Types and Function Block Types using Function Designer in libraries. Do not create them in applications.

A composite agitator object type including a Function Diagram and a corresponding signal group composite object type is used to show configuration of user-defined object types here. The object types are used to configure the function 1T1MC1 later.

Creating an User-defined Library

To create an user-defined object type library **AgLib**:

1. Open Control Builder M on the required project in Control Structure.
2. In Project Explorer, right-click the Libraries folder object and click **New Library...** from the context menu.
3. In the **New Library** dialog enter **AgLib** as the name of the library.
4. Click **OK** to create AgLib 1.0-0.

The library AgLib 1.0-0 is visible in Project Explorer, in the Library Structure and in the Object Type Structure. Since it is a Control Library it is placed in the Object Type Structure under
Object Types\Control System\AC 800M/C Connect\Libraries\AgLib 1.0-0.

Creating a Data Type

Follow the steps to create an user-defined structured data type IO_SAgitator in AgLib to be used by SimpleAgitator to connect to the I/O signals:

1. In Project Explorer, right-click Libraries\AgLib 1.0-0\New\Data Type... .
2. In the **New Data Type** dialog enter IO_SAgitator as the name of the data type.
3. Click **OK** to create IO_SAgitator.
4. In Project Explorer, right-click Libraries\AgLib 1.0-0\Data Types\IO_SAgitator.
5. Click **Editor**.
6. In the Data Type editor dialog, enter the components as shown in the figure below.

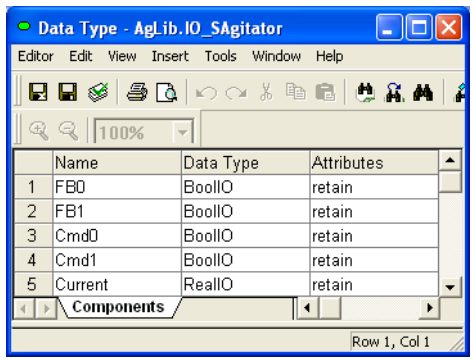


Figure 24. Structured Data Type IO_SAgitator

7. In the editor tool bar click **Save and Close**.

Creating a Control Module Object Type

Follow the below sub-sections to create an user-defined control module object type SimpleAgitator in the library AgLib that provides a Function Diagram which can be used to prepare an empty type and then edit the contents.

Preparing the Control Module Type:

Follow the steps to prepare the Control Module type:

1. In the Object Type Structure in Engineering Workplace, navigate to Object Types\Control System\AC 800M/C Connect\Libraries\AgLib 1.0-0\Control Module Types.
2. Right-click Control Module Types.
3. In the context menu click **New Object...**.
4. In the **New Object** dialog:
   a. Click Control Module Type.
   b. Enter **Name** as SimpleAgitator and an appropriate object **Description**.
   c. Click **Create** to create the type.
5. In Project Explorer of Control Builder M, navigate to the new control module type: <project>\Libraries\AgLib 1.0-0\Control Module Types\SimpleAgitator.
6. Right-click SimpleAgitator.
7. Click **Properties** from the context menu**.** Make sure that **Instantiate as Aspect Object** is selected. (Otherwise the object type would be instantiated as a symbol object and will not be visible in the Functional Structure and Control Structure).

Preparing the Function Diagram:

Follow the steps to prepare the Function Diagram:

1. In the Object Type Structure of Plant Explorer, right-click the new control module type, in the newly created user-defined library.
2. Click **New Aspect...** from the context menu.
3. In the **New Aspect** dialog navigate to and click Function aspect.
4. Click **Create** to add a Function aspect to the Control Module Type object.
5. Right-click the Function aspect and click **Diagram** to open the Diagram view.
6. Click **File > New**.

7. In the Select Master Page Template dialog click an appropriate template, for example A3 Landscape.
8. Click **OK** to create a new empty diagram based on this template.

Empty Function Diagrams can not be downloaded to Control Builder M.

Editing the Function Diagram:

1. Insert the required Function Components provided in the table below into the Function Diagram according to Inserting Function Components and connect them according to Connecting Function Components.

Table 6. Function Components SimpleAgitator

Function Component Object Type	Name	Description	Additional Action
Object Types\Control System \AC 800M/C Connect\Libraries \ProcessObjExtLib 2.2-2 \Function Block Types\MotorUni	SimpleAgitator_Motor	Motor control	
Object Types\Functional Planning\Generic Function Components\Connectors \Diagram Parameters\Input Parameter	Name	IN Name	a) Set Variable Properties: Datatype: String[30] b) Connect to port Name of SimpleAgitator_Motor
" " "	Description	IN Description	a) Set Variable Properties: Datatype: String[40] b) Connect to port Description of SimpleAgitator_Motor
" " "	SetAuto	IN Indicates auto mode	a) Set Variable Properties: Datatype: Bool b) Connect to port SetAuto of SimpleAgitator_Motor

Table 6. Function Components SimpleAgitator (Continued)

Function Component Object Type	Name	Description	Additional Action
“ “ “	AutoCmd1	IN Auto command 1	a) Set Variable Properties: Datatype: bool b) Connect to port AutoCmd1 of SimpleAgitator_Motor
“ “ “	AutoCmd0	IN Auto command 0	a) Set Variable Properties: Datatype: bool b) Connect to port AutoCmd0 of SimpleAgitator_Motor
“ “ “	GrpStartIn	IN Group start in	a) Set Variable Properties: Datatype: GroupStartStepConnection b) Connect to port GroupStartIn of SimpleAgitator_Motor

Table 6. Function Components SimpleAgitator (Continued)

Function Component Object Type	Name	Description	Additional Action
“ “ “	IO	IN IO signal group	a) Set Variable Properties: Datatype: IO_SAgitator b) Connect the components to ports of SimpleAgitator_Motor: IO.FB1->FB1, IO.FB0->FB0, IO.Cmd1->Out1, IO.Cmd0->Out0 and IO.Current->MotorValue The most efficient way is to select the port and to enter IO.<component> in the Connection toolbar´s edit combo box. Splitter and joiner Function Blocks can be used when the user tries to graphically connect the ports having different data types. In the example, component notation is used at the sink port of the connection.
“ “ “	Interaction Par	IN Interaction parameter	a) Set Variable Properties: Datatype: MotorUniPar b) Connect to port InteractionPar of SimpleAgitator_Motor
Object Types\Functional Planning\Generic Function Components\Connectors \Diagram Parameters\Output Parameter	AutoMode	OUT Indicates auto mode	a) Set Variable Properties: Datatype: Bool b) Connect to port AutoMode of SimpleAgitator_Motor

Table 6. Function Components SimpleAgitator (Continued)

Function Component Object Type	Name	Description	Additional Action
“ “ “	ManMode	OUT Indicates manual mode	a) Set Variable Properties: Datatype: Bool b) Connect to port ManMode of SimpleAgitator_Motor
Object Types\Control System\AC 800M/C Connect \System Functions\Functions\concat	-	-	a) Connect port String1 to port Name of SimpleAgitator_Motor b) Connect port String2 to string ‘in ManMode’.
Object Types\Control System \AC 800M/C Connect\Libraries \AlarmEventLib 1.3-0 \Function Block Types\AlarmCond	SimpleAgitator_Alarm	Alarm condition motor control	a) Connect port Signal to port ManMode of SimpleAgitator_Motor b) Connect port SrcName to port Name of SimpleAgitator_Motor c) Connect port Message to port Out of concat

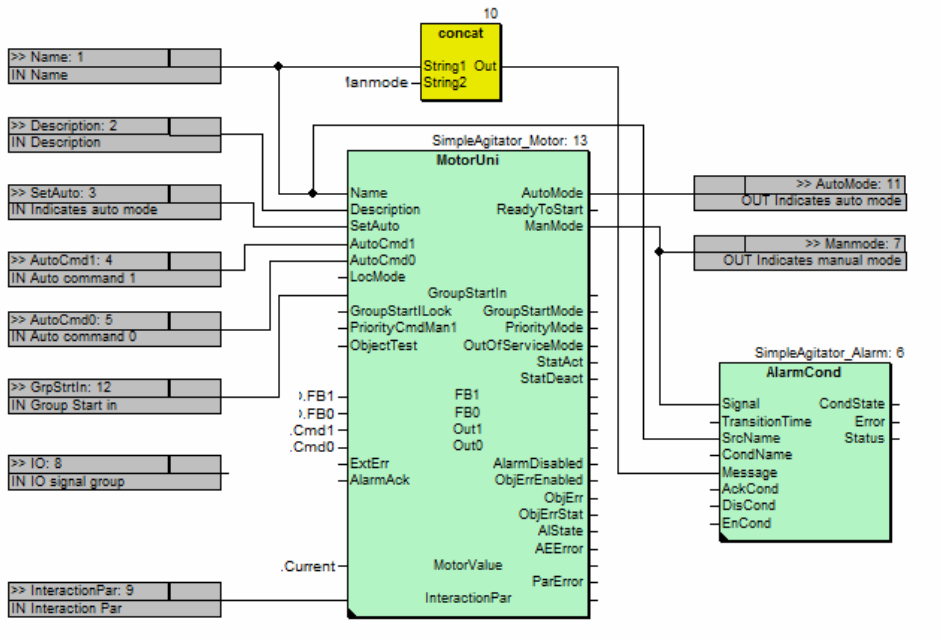


Figure 25. Function Diagram of Object Type SimpleAgitator

2. To configure non-mandatory port connections on instance level for the parameters **GrpStartIn**, **AutoMode**, and **ManMode**, set the initial value for these parameters to default:

   a. Right-click the required port.

   b. Click **Aspect Properties...** from the context menu.

   c. In the **All** tab of the **Aspect Properties** dialog, click on the Value cell of the **Value.Initval** row.

   d. Enter value as default.

   e. Click **Apply** and click **OK**.

3. To make the concat function run in an user-defined code block that is executed only once after download, introduce an additional property:

   a. Right-click the concat symbol.

b. Click **Aspect Properties...**from the context menu.

c. Select **All** tab and right-click the gray area around.

d. Click **Add Property** from the context menu.

e. Enter Name NewCodeBlock.

f. In the new row, click the **Data Type** cell.

g. Enter **String** from the pick list.

h. In the new row, click the **Access Type**.

i. Enter **Read&Write** from the pick list.

j. In the new row, click the **Value cell**.

k. Enter value **Start_CodeBlock**.

l. Click **Apply** and click **OK**.

To delete a block containing sub-diagrams (eg. Sequence) configured in a Control Module Type, delete the existing diagram parameters within the sub-diagrams before deleting the blocks.

Finalizing Control Module Object Type

Follow the steps to finalize the Control Module object type:

1. To provide an instance specific name according to naming rules to instances of SimpleAgitator_Motor:

   a. In the Object Type Structure of Engineering Workplace, right-click SimpleAgitator_Motor and click **New Aspect...** from the context menu.

   b. In the **New Aspect** dialog, navigate to Aspect System Structure\Basic Property Aspect\Basic Property Name and select **Name**.

   c. Click **Create.**

   d. Double-click the new aspect to open the Config View.

   e. Edit Name as SimpleAgitator_Motor.

   f. Click **Apply**.

   g. Click **Close** on window level to exit.

2. To instantiate the SimpleAgitator type place the SimpleAgitator_Motor instance in Functional Structure, a formal instance of the Functional Structure has to be configured:
   a. Right-click the Formal Instance List object of SimpleAgitator.
   b. Click **Insert Object...**.
   c. In the Object Type Structure shown in the **Insert Object** dialog navigate to and click SimpleAgitator_Motor.
   d. Click **Insert**.
   e. Click **Close**.

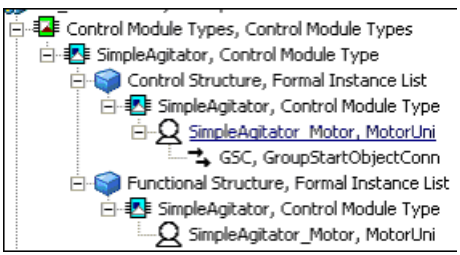


Figure 26. Object Type SimpleAgitator

3. Generate or edit Data Flow Order according to Configuring Data Flow Order.
4. Generate Configuration Data according to Generating Configuration Data. Note that allocation to an application is not needed for types.
5. Follow the steps to test the SimpleAgitator type:
   a. Create a test object with a Function Diagram according to Creating a new Function Diagram.
   b. Insert the SimpleAgitator type into the Function Diagram according to Inserting Function Components.
   c. Connect the ports mandatory to connect according to Connecting Function Components.
   d. Allocate the Function Diagram according Allocating a Function Diagram.

e. Generate configuration data according to Generating Configuration Data.

f. Download and test the application according to Downloading and Testing the Application.

Creating a Signal Group Object Type

User creates a Signal Group object type to group I/O signals of structured data type IO_SAgitator.

While creating a signal group object, select the following in the **Aspect Control** tab of the Object Type Definition aspect:

- **Inherit to All Instances** check box for the Function aspect.
- **Copy to all instances** check box for the Function Parameter aspect.

Preparing the Object Type

1. In the Object Type Structure of Engineering Workplace, navigate to Object Types\Control System\AC 800M/C Connect\Libraries\AgLib 1.0-0.
2. Right-click AgLib 1.0-0 and click **New Object...** from the context menu.
3. In the **New Object** dialog:
   a. Click Object Type Group.
   b. Enter Name as SignalGroups.
   c. Click **Create**.
4. Right-click the SignalGroups folder object.
5. In the **New Object** dialog:
   a. Click Object Type.
   b. Enter Name as IOSAgitator.
   c. Click **Create**.
6. Right-click IOSAgitator object.
7. In the **New Object** dialog:
   a. Verify that the **Formal Instance List** is selected in the **Common** tab.

b. Verify that Functional Structure is selected in the Instantiation Structure drop-down.

c. Enter Name as Functional Structure.

d. Click **Create**.

Extending the Object Type with Control Builder Name

To support a IEC 61131 compatible variable name in CBM a Control Builder Name aspect is required:

1. Right-click the IOSAgitator signal group object.
2. Click **New Aspect...**.
3. In the **New Aspect** dialog navigate to Basic Property Aspects\Basic Property Name and select Control Builder Name.
4. Click **Create.**
5. Right-click the IOSAgitator Type Definition aspect and click **Config View** from the context menu.
6. In the **Aspect Control** tab of the **Type Definition** dialog select the Control Builder Name row and select the **Copy to all instances** check box.
7. Click **Apply**.
8. Click **Close** on window level to exit.

Configuring Signal Information

1. Right-click the IOSAgitator signal group object.
2. Click **New Aspect...**.
3. In the New Aspect dialog navigate to and click on Parameter Manager aspect category CBM_SignalInformation.
4. Click **Create.**
5. Right click on the IOSAgitator Type Definition aspect.
6. Click **Config View**.
7. In the **Aspect Control** tab of the **Type Definition** dialog, select the CBM_SignalInformation row and select the **Copy to all instances** check box.

8. Click **Apply**.
9. Click **Close** on window level to exit.
10. Right-click the CBM_SignalInformation aspect.
11. Click **Main View**.
12. Click the option **Connect to Application Global Variable of Structured Data Type (Signal Group)**.
13. Enter Variable Data Type IO_SAgitator.
14. Click **Apply**.
15. Click **Close** on window level to exit..

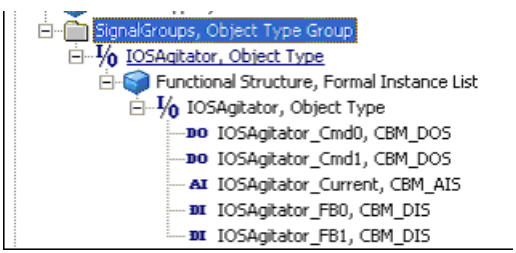


Figure 27. Object Type IOSAgitator

Extending the Object Type with Formal Instances

1. Right-click the IOSAgitator object below the Functional Structure object and click **New Object...** from the context menu.
2. In the **New Object** dialog:
   a. Navigate to Object Types\ Control System\ AC 800M/C Connect\ CBM_Signals.
   b. Click **CBM_DOS**.
   c. Enter Name as Cmd0 and an appropriate Object description. This is the name that will be stored in the Relative Name aspect of the object.
   d. Click **Create** to create the signal object.

3. Right-click the new signal object and click **New Aspect...** from the context menu.
4. In the **New Aspect** dialog navigate to and click a Basic Property Name aspect category Name. This is the name aspect to hold the name template for instances.
5. Click **Create.**
6. Double-click the new aspect to open the **Config View**.
7. Edit Name to hold IOSAgitator_Cmd0.
8. Click **Apply**.
9. Click **Close** on window level to exit.
10. Repeat Step 1 to Step 9 for
    CBM_DOS with Relative Name Cmd1 and Name IOSAgitator_Cmd1,
    CBM_AIS with Relative Name Current and Name IOSAgitator_Current,
    CBM_DIS with Relative Name FB0 and Name IOSAgitator_FB0, and
    CBM_DIS with Relative Name FB1 and Name IOSAgitator_FB1.

For each IO signal, the CBM_SignalInformation aspect has to be set as **Connect to Application Global Variable** . Do not change the given Variable Data Type.

Copying a Graphical Symbol.

1. In the Object Type Structure navigate to Object Types\Control System\AC 800M/C Connect\CBM_Signals.
2. Click **CBM_DIS** object type.
3. Right-click Function Parameters aspect.
4. Click **Copy**.
5. Click IOSAgitator object.
6. Right-click in the aspect area and click **Paste** from the context menu.
7. Then repeat Step 2 to Step 6 for the Function aspect.
   This copies the pre-configured component view of the CBM_DIS object type as the base for adaptations.

8. Right-click Function Parameters aspect and click **Details...** from the context menu.
9. In the **Aspect Info** tab of the **Details** dialog, click **Add...**.
10. Click **Auto-Instantiate** aspect from the selection list.
11. Click **OK** to exit the **Details** dialog.

Changing the Graphical Symbol.

Follow the steps to edit the symbol and the labels in the Component view of Function Designer:

1. Right-click the copied Function Aspect and click **Component** from the context menu.
2. Right-click on the lower label and click **Delete**.
3. Right-click the upper label and click **Component Properties...**from the context menu.
4. In the **Field** tab select and delete the second reference.
5. Click **OK**.
6. In the symbol select and delete: # symbol, polyline and input circle.
7. In the rectangle above the diagonal insert a text I, below a text O.
8. Right-click the background of the drawing and click **Aspect Properties...** from the context menu.
9. In the **Aspect Properties** dialog, click the first row dealing with IN port.
10. Shift and click the last row dealing with IN port.
11. Right-click the selected area and click **Remove Property** from the context menu.
12. In the **Aspect Properties** dialog, right-click the gray area around.
13. Click **Add Property** from the context menu.
14. Update the Aspect Properties dialog, for the values indicated in Figure 28.

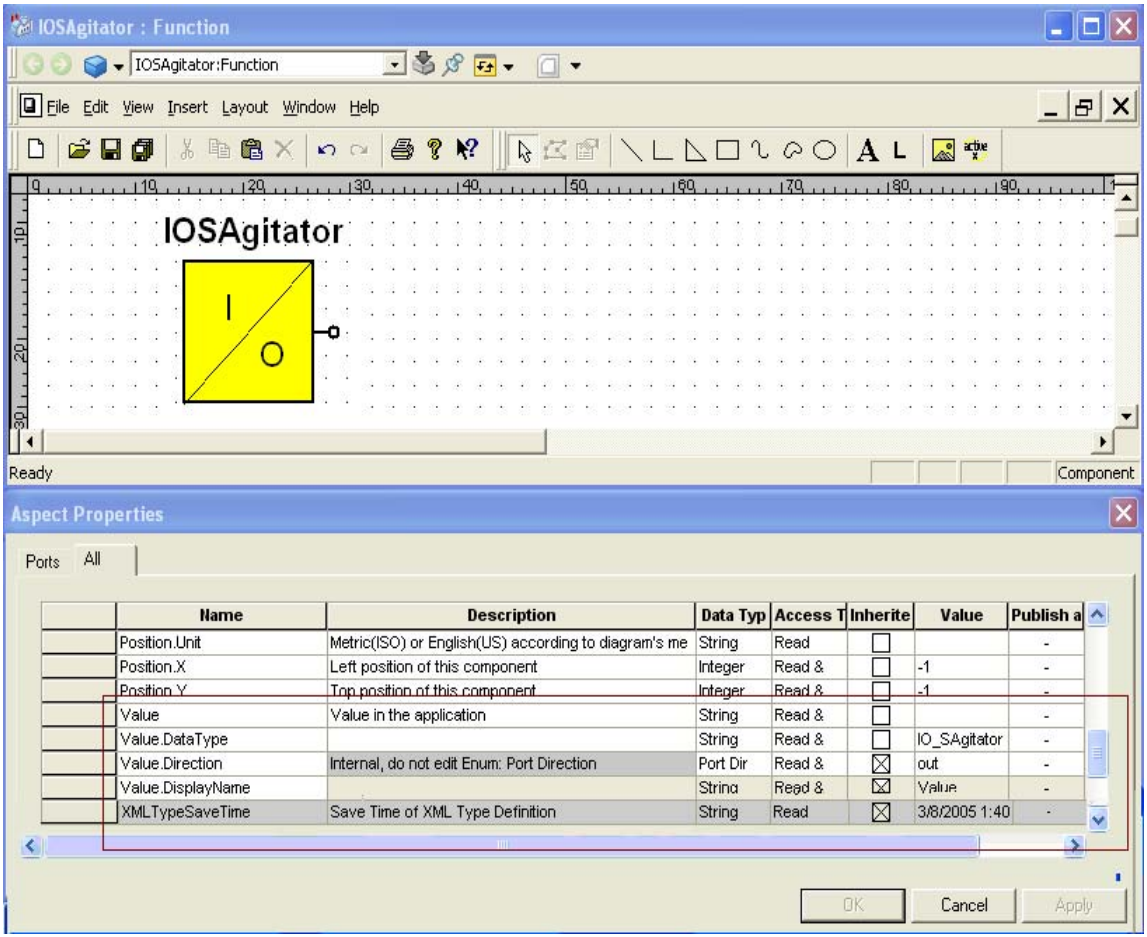


Figure 28. Graphical Symbol for IOSAgitator

15. Click **OK**.
16. In the Function Designer window click **Save**.
17. Click **Close** on window level to exit.

Adding a Function Diagram

Follow the steps to add a Function Diagram:

1. Right-click the copied Function Aspect and click **Diagram View** from the context menu.
2. Click **File > New**.
3. In the **Select a Master Page Template** dialog, select a template, for example A3 Landscape.
4. Click **OK**. All CBM_Signal objects are displayed automatically in the Function Diagram. The objects can be arranged as required, refer to the figure below.
5. Click **Save**.
6. Click **Close** on window level to exit.

While Instantiating the signal groups in Function Structure, default considered for Diagram concept is **Local Variable for Diagram** instead of **Local Variable on Diagram of Structured Data Type** and default considered for Single Control Module is **Application Global Variable** instead of **Application Global variable of structure data type**.

Do not perform **Generate Configuration Data** for this Function Diagram.

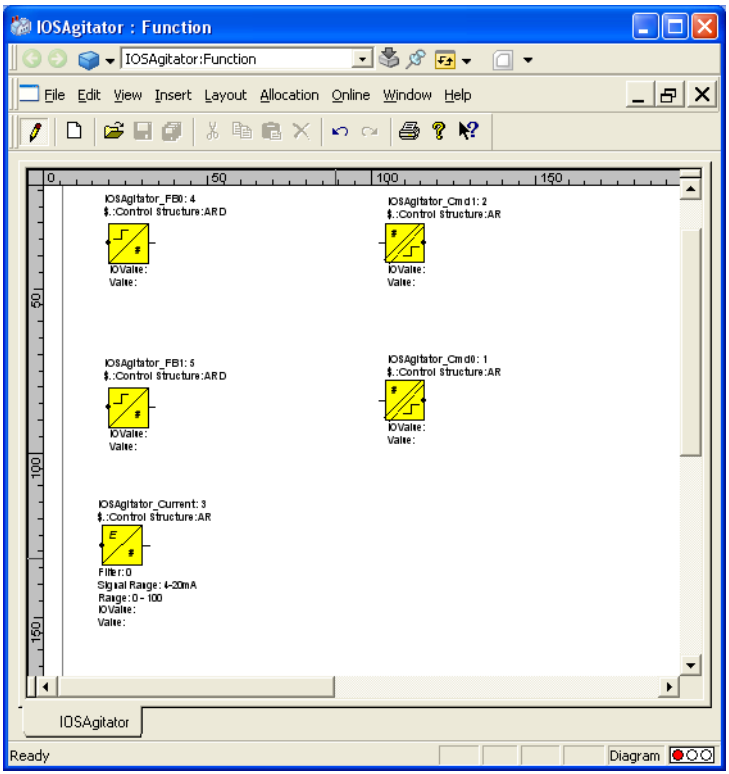


Figure 29. Function Diagram of Object Type IOSAgitator

Completing the Function

Creating the Agitator

Follow the steps to configure the agitator function 1T1MC1 based on the user-defined object types SimpleAgitator and IOSAgitator:

1. Insert the required Function Components provided in the table below into the Function Diagram according to Inserting Function Components and connect them according to Connecting Function Components.

Table 7. Function Components Agitator 1T1MC1

Function Component Object Type	Name	Description	Additional Action
Object Types\Control System \AC 800M/C Connect\Libraries \AgLib 1.0-0 \Control Module Types\SimpleAgitator	1T1MC1 _M	Agitator	
Object Types\Control System \AC 800M/C Connect\Libraries \AgLib 1.0-0 \SignalGroups\IOSAgitator	1T1MC1 _IO	Signalgroup for Agitator	a) Connect to port IO of 1T1MC_M
Object Types\Functional Planning\Generic Function Components\Connectors\Off-Diagram References\Reduced Input Reference	SetAuto	Diagram ref SetAuto	a) Set Variable Properties: Datatype: bool, Attributes: retain b) Connect to Diagram Variable 1T1MC1_SetAuto c) Connect to port SetAuto of 1T1MC1_M
" " "	AutoCmd1	Diagram ref AutoCmd1	a) Set Variable Properties: Datatype: bool, Attributes: retain b) Connect to Diagram Variable 1T1MC1_AutoCmd1 c) Connect to port AutoCmd1 of 1T1MC1_M
" " "	AutoCmd0	Diagram ref AutoCmd0	a) Set Variable Properties: Datatype: bool, Attributes: retain b) Connect to Diagram Variable 1T1MC1_AutoCmd0 c) Connect to port AutoCmd0 of 1T1MC1_M

Table 7. Function Components Agitator 1T1MC1 (Continued)

Function Component Object Type	Name	Description	Additional Action
“ “ “	Interaction Par	Diagram ref Interaction Par	a) Set Variable Properties: Datatype: MotorUniPar, Attributes: retain b) Connect to Diagram Variable 1T1MC1_InteractionPar c) Connect to port InteractionPar of 1T1MC1_M
Object Types\Functional Planning\Generic Function Components\Connectors\Off-Diagram References\Reduced Output Reference	AutoMode	Diagram ref AutoMode	a) Set Variable Properties: Datatype: bool, Attributes: retain b) Connect to Diagram Variable 1T1MC1_AutoMode c) Connect to port AutoMode of 1T1MC1_M
“ “ “	ManMode	Diagram ref ManMode	a) Set Variable Properties: Datatype: bool, Attributes: retain b) Connect to Diagram Variable 1T1MC1_ManMode c) Connect to port ManMode of 1T1MC1_M

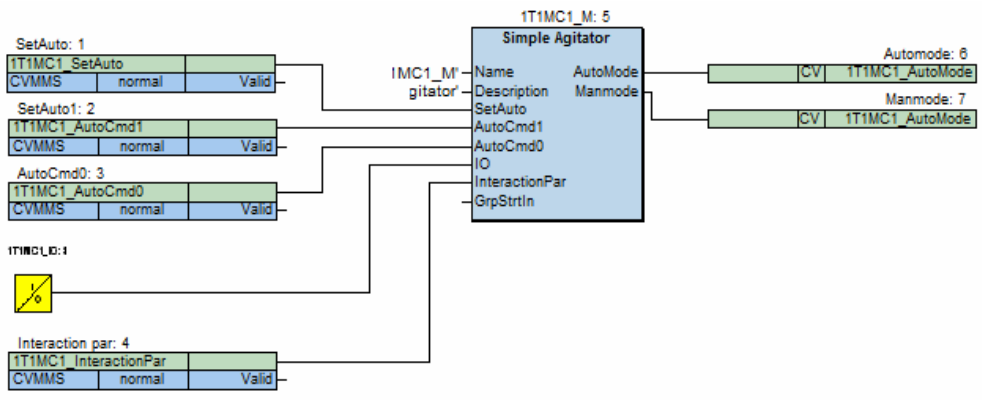


Figure 30. Function Diagram Agitator 1T1MC1

2. Configure Data Flow Order according to Configuring Data Flow Order if needed.
3. Allocate the Function Diagram to Application_1 according to Allocating a Function Diagram.
4. Test configuration data generation according to Generating Configuration Data.

Interlocking Valve Control

The two valve control functions require interlocks:

- 1T1VC1 shall be able to open if 1T1VC2 is closed.
- 1T1VC2 shall be able to open if 1T1LC1LT.GTH is true.

To configure interlock connections:

1. Double-click Function aspect of object 1T1VC2 to open the Function Diagram.
2. Right-click port StatDeact, connect it to a new **Reduced Output Diagram Reference** named **StatDeact** and connect this reference to Communication Variable 1T1VC2_StatDeact.
3. Also open Function Diagram 1T1VC1 using **File > Open**.

4. Right-click **Reduced Input Diagram Reference** named **Ilock1** and connect it to existing Communication Variable 1T1VC2_StatDeact.
5. Also open Function Diagram 1T1LC1 using **File > Open**.
6. Right-click port GTH, connect it to a new **Reduced Output Diagram Reference** named **GTH** and connect this reference to Communication Variable 1T1LC1_GTH.
7. Click the workbook tab 1T1VC2, and switch to the opened Function Diagram 1T1VC2.
8. Right-click **Reduced Input Diagram Reference** named **Ilock1** and connect it to existing Communication Variable 1T1LC1_GTH.
9. Use the workbook tabs to switch between the three open diagrams to verify if the cross-reference information is available in the interconnected Diagram References.

Reusing the Function

User can copy tank 1T1 and rename it to 1T2, to provide a second tank function.

Copying and Renaming the Function

1. In Engineering Workplace, right-click 1T1 object in Functional Structure and click **Copy**.
2. Right-click object Area1 and click **Paste**.
3. Click the new object to select it.
4. Add a Function aspect.
5. Double-click Name aspect.
6. Change Name 1T1 to 1T2 and adapt Object description.
7. Click **Apply**.
8. Click **Close** on window level.

All sub-objects are renamed automatically according the naming scheme and Control Builder Name aspects are updated.

If more than one copy of the Function Diagram is required, it is recommended to use Bulk Data Manager. Use either the pre-configured worksheet BDM_for_Function Designer.xls (see Instantiating the Typical) or configure a worksheet according to the user requirements.

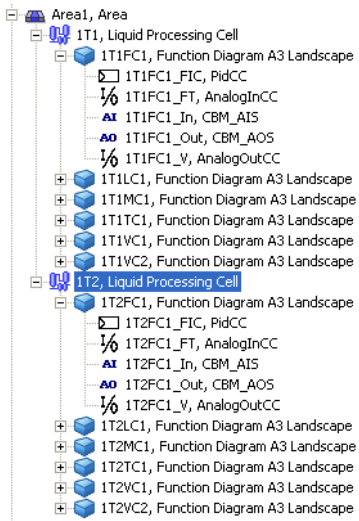


Figure 31. Function 1T1 Copied and Renamed to Function 1T2

Reconfigure Diagram References

If all the copied Diagram References are connected to Communication Variables, then by default, the pasted Diagram References and Communication Variables also remain connected.

To reconnect these copied diagram references in the corresponding diagrams:

1. Right-click the diagram reference symbol in the diagram and click **Disconnect From Diagram Variable**.
2. Right-click the diagram reference symbol again and click **Connect To Diagram Variable...** from the context menu.

3. In the **Connect To Diagram Variable** dialog:
   a. Accept or change the name of the Communication Variable.
   b. Click **OK**.

Allocating the Function Diagrams

Besides allocating the Function Diagrams individually as described in Allocating a Function Diagram, user can also allocate them in bulk using Bulk Data Manager.

1. Open and attach workbook BDM_for_Function_Diagrams for start object Area1 in Functional Structure.
2. Switch to worksheet **Allocate FuD**. Enter value 0 into the first cell of column Allocated for filtering unallocated diagrams.
3. Drag & drop object Area1 onto the worksheet.
   The result is a list of all unallocated Function Diagrams.
4. For every listed Function Diagram enter the path to the application into the cell of the Application Allocation row:
   a. Right-click the cell and click **Insert Object Path...** from the context menu.
   b. Select Control Structure.
   c. Navigate to the corresponding application, click **...<Application>.Diagrams** (for newly created diagrams).
   d. Click **Object Path**, and the focus jumps to the next cell.
   e. Continue to click the **Object Path** until a new application has to be selected. Then continue with Step c. Allocate Function Diagrams of 1T1 to Application_1 and of 1T2 to Application_2.
   f. Click **Cancel**.
5. To allocate click **Save all Objects** in Bulk Data Manager toolbar**.**

Allocating the Signals

1. Open the IO Allocation tool in the Control Structure on the Controller object for which the I/O Boards are attached to. (See Allocating I/O how to open the tool.) The available boards are displayed in the left pane of the window.
2. From the Functional Structure drag Area1 on the right pane of the IO Allocation window. All engineering signals (CBM_Signal objects) are read.
3. Drag and Drop one or several signal objects from the grid of the right pane of the window to the Boards object or to an I/O board object or to a channel stub of an I/O board object in the left pane of the window.
   After Drop the allocation is shown on both sides.
4. Click **File** > **Exit** to close the I/O Allocation dialog**.**

Configuring Interlock Connections

Interlocking Functions

Configure interlocks between the functions 1T1 and 1T2 with the following conditions:

- Only one inlet valve 1T1VC1_V1 or 1T2VC1_V1 should be able to open at a time.
- The outlet valve 1T1VC2_V should be locked if the level at 1T2LC1_LT is reached.
- The outlet valve 1T2VC2_V should be locked if the level at 1T1LC1_LT is reached.

1. Implement these interlocks similar to the Interlocking Valve Control, but according to Table 8.table

Table 8. Interlocks of 1T1 and 1T2

Function Diagram	Output Diagram Reference	Input Diagram Reference	Communication Variable
1T1VC1		Ilock2	1T2VC1_StatAct
1T2VC1	StatAct		1T2VC1_StatAct
1T2VC1		Ilock2	1T1VC1_StatAct
1T1VC1	StatAct		1T1VC1_StatAct
1T1VC2		Ilock2	1T2LC1_GTH1
1T2LC1	GTH		1T2LC1_GTH1
1T2VC2		Ilock2	1T1LC1_GTH
1T1LC1	GTH		1T1LC1_GTH

2. Ensure the following in the interlock connections:
   - The cross reference field of the input diagram references connected to the output diagram references of a Function Diagram which is allocated to another application, shows an additional blue CVMMS marker as shown in Figure 32.

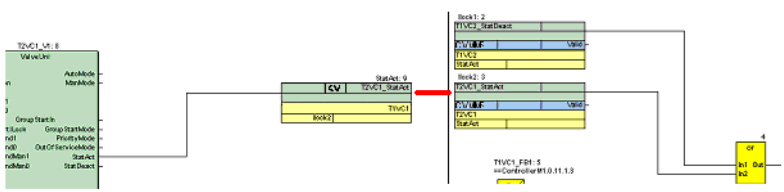


Figure 32. Interconnection Requiring CVMMS Communication

To establish MMS communication between Function Diagrams created prior to Engineering Studio 800xA 5.1, refer to MMS Communication in *System 800xA Engineering, Engineering Studio Function Designer (3BDS011224*)*.

To establish communication between a restored diagram and a newly created diagram, refer to **Connection between Restored Diagrams (created upto 800xA 5.0 SP2) and New Diagrams (created from 800xA 5.1 onwards)** in *System 800xA Engineering, Engineering Studio Function Designer (3BDS011224*)*.

Generating Configuration Data

Besides generating configuration data for the Function Diagrams individually as described in Allocating a Function Diagram, user can generate configuration data in bulk using Bulk Data Manager.

1. Open and attach workbook BDM_for_Function_Diagrams for start object Area1 in Functional Structure, see Opening and Attaching BDM_for_Function_Diagrams.
2. Switch to worksheet **Generate Code**.
3. Drag & drop object Area1 onto the worksheet.
   The result is a list of all Function Diagrams below and including the start object showing status indications for them: Generated, Allocated, I/O Allocated, (Modification) Status.
4. To generate configuration data check if diagrams are allocated, if yes:
   a. Select (multiple) cell(s) in Column K (Generated).
   b. Right-click the selected cell(s).
   c. Click **Aspect Commands**.
   d. Click **Generate Configuration Data** or **Generate Configuration Data (Full Build)**.
   e. Wait for the progress indicator box to disappear.

To re-check the status clear the data lines and drag & drop the diagrams again onto the worksheet.

Downloading and Testing the Application

Download and test the application according to Downloading Configuration Data and Testing Configuration. For examples based on functions 1T1 and 1T2 see Test and Commissioning.

The following Function Designer features can be used to test the diagrams:

- Subscribe for live data in the diagrams.
- Follow the signal flow through diagram references.
- Navigate between parent and child diagrams.
- Watch Window to watch and force the variable properties.
- Force menu item to force variables of certain data types.

Printing Documentation

Print the Function Diagram documentation according to Printing Documentation.

Assume the user has created a contents diagram in object Area1. All diagrams of the area including the leading contents diagram can be printed by performing the following steps on the Function aspect in object Area1:

1. Click **File > Print...** to open the Print dialog.

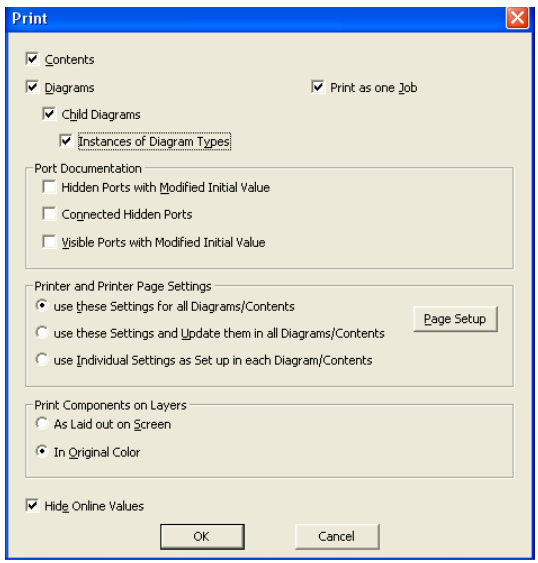


Figure 33. Print Dialog

2. Select the **Contents** check box, if the user has to print the generated contents diagram as the first diagram.
3. Select the **Diagrams** check box, if the user has to print all top level diagrams.
4. Select the **Child Diagrams** check box, if the user has to print all child diagrams.
5. Select the **Print as one Job** check box.
6. Select the **Instances of Diagram Types** check box, if the user requires to additionally print the Function Diagrams of the used types with instance specific information.
7. Select the Port Documentation check boxes (**Hidden Ports with Modified Initial Value**, **Connected Hidden Ports**, **Visible Ports with Modified Initial Value**) as required. This will generate and print additional pages with information for these kinds of ports.

8. Select the **Hide Online Values** check box, if the user wants to mask the online values from the printed Function Diagrams.
9. Click **OK** to open the **Print** dialog for the printer.
10. Click **OK** to print or click **Cancel** to discard printing.

Procedure to print template information in signal group

Template information of an instantiated signal group object may be inconsistent while printing function diagram. To avoid this perform the following steps:

1. Create the Signal Group Object Type.
2. Create a document aspect for **IOSAgitator** and name it as Function Diagram Document.

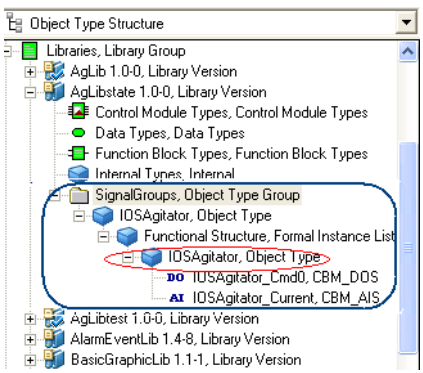


Figure 34. Template information in signal group

3. Right click **Function Diagram Document** in aspect list, and select **Main View**. Enter all the necessary information.
4. In Function Structure, create a new diagram template . The document aspect with name Function Diagram Document is automatically added.
5. Right click Function Diagram Document in aspect list, and select Main View. Enter all the necessary information.

This information should be same as the information updated in Object Type Structure in **step 3**.

6. Drag and drop the signal group from the Object Type Structure to the Diagram.
7. The correct information is updated in parent and child diagram.

Process Sequence Model

If the process shown in Process Modelas a sequence of process steps, this results in a sequence flow diagram as shown in the figure below.

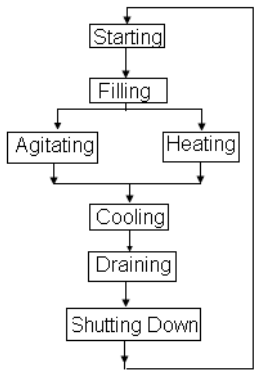


Figure 35. Sequence Flow Diagram Tank 1

User can implement this flow diagram as the overview diagram of a sequence in Function Designer.

Configuring Process Sequences

asic information on sequence support in Function DesignerTypes of Sequences.

Configuring the Function Diagram

Follow the steps to create a Function Diagram as a container of the sequence for Tank1:

1. Create an object, for example 1T1Seq1, below object 1T1 according to Creating a new Function Diagram.
2. Open the Function Diagram and insert an IEC 61131-3 sequence component according to Inserting a Sequence\Sequence2D Component.
3. Show hidden ports of the sequence component according to Showing / Hiding Ports.
4. Add diagram references according to Ports on Different Function Diagrams as required to control the sequence from other diagrams.

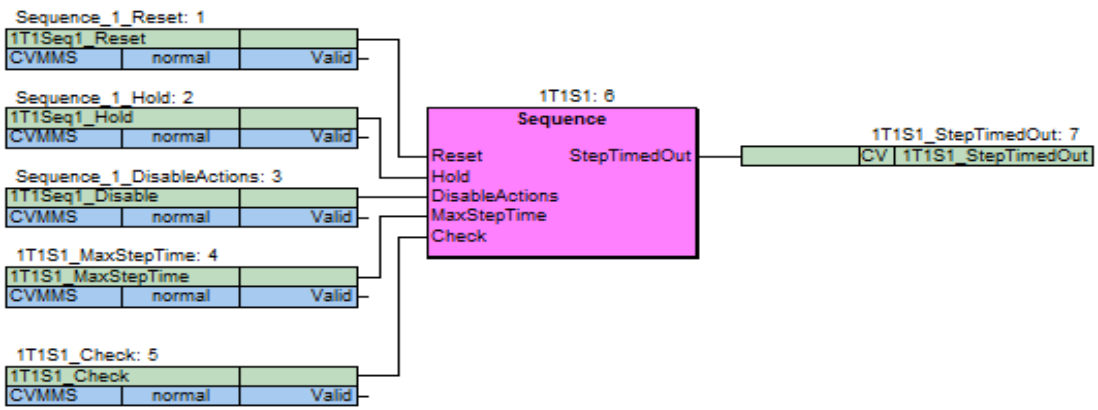


Figure 36. Function Diagram with IEC61131-3 Sequence

5. Allocate the diagram to Application_1 according to Allocating a Function Diagram. By this operation the complete sequence is allocated to the application as required.

Configuring the Sequence

The sequence example is not configured completely in this manual. Besides the overview diagram as an example only step Filling and transition filled are shown.

Overview Diagram

To implement the overview diagram according to the process sequence flow diagram shown in the figure in Process Sequence Model:

1. Open and edit the sequence 1T1_S1 in the Function Diagram 1T1Seq1 according to Editing Overview Diagram. The figure below shows an example design for the sequence.
2. Perform an initial check for the sequence by generating configuration data according Generating Configuration Data.
3. The sequence can be accessed from the Project Explorer of Control Builder M by opening the editor on the corresponding Diagram x1T1Seq1.

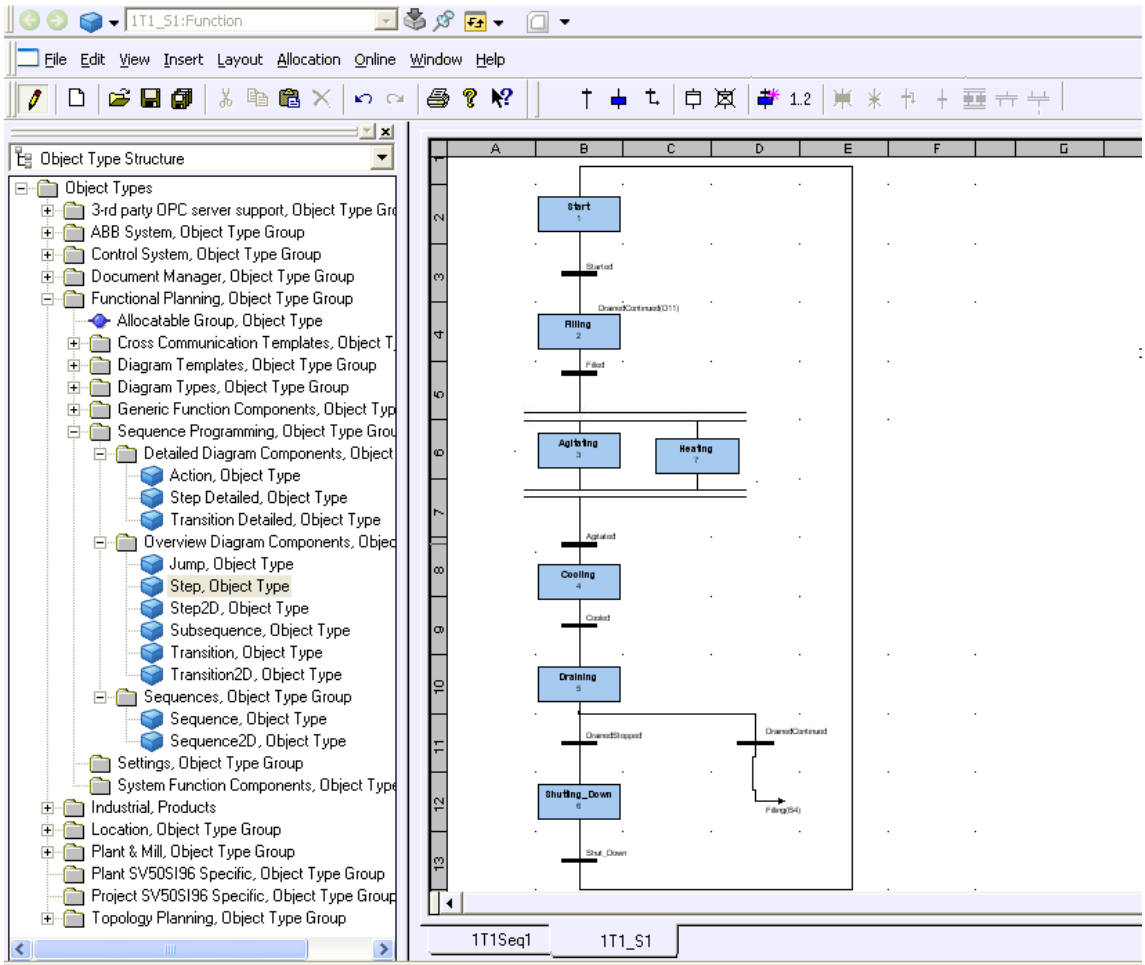


Figure 37. Sequence Overview Diagram in Function Designer

Steps

User can configure a step detail diagram for step Filling according to Editing Step Detail Diagram.

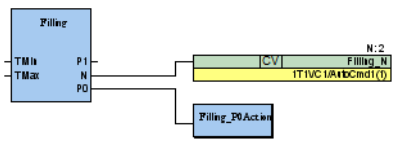


Figure 38. Step Detail Diagram for Step Filling

Actions

The intended actions are to set command AutoCmd1 of valve T1VC1_V1 in N action and to reset AutoCmd1 in P0 action. For the N action in our example user can use a diagram reference in the step detail diagram.

For the P0 action user has to configure an action diagram for P0 action according to Editing Action Diagram.

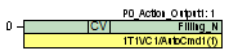


Figure 39. Action Diagram Filling_P0Action for Step Filling

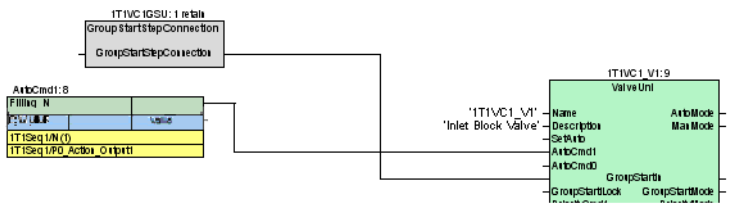


Figure 40. N and P0 Action of Step Filling Connected to AutoCmd1

Due to restrictions complex action procedures must be implemented in the normal Function Diagrams.

Transitions

For transition Filled user can configure a transition detail diagram according to Editing Transition Detail Diagram. In our example this connects just the GTH output of 1T1LC1_LT.

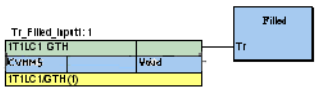


Figure 41. Transition Detail Diagram Filled

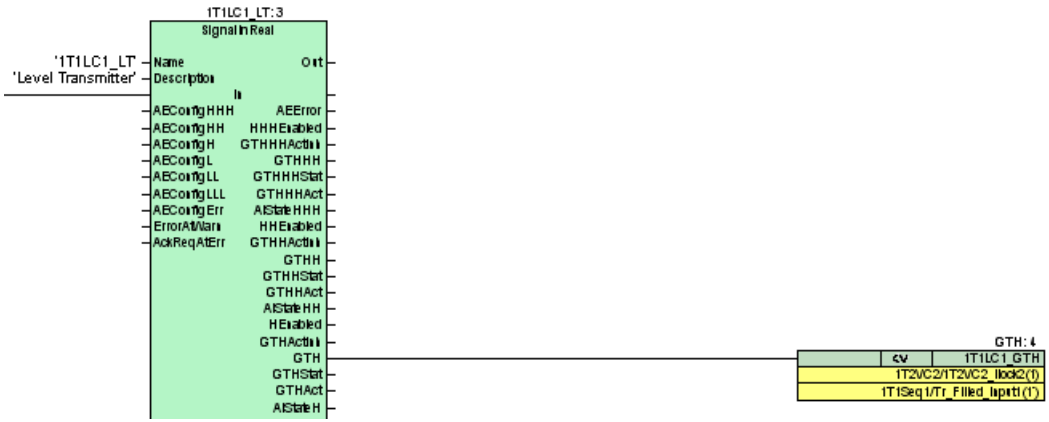


Figure 42. Source for Transition Filled

Reusing a Complete Sequence

To provide the same sequence for the second tank function user can copy 1T1Seq1 and rename it as 1T2Seq1.

Copying and Renaming the Sequence

1. In Engineering Workplace, right-click 1T1Seq1 object in Functional Structure and click **Copy** from the context menu.
2. Right-click on object 1T2 and click **Paste** from the context menu.

3. Click the new object to select it.
4. Double-click **Name** aspect.
5. Change Name 1T1Seq1 to 1T2Seq1 and adapt Object description.
6. Click **Apply**.
7. Click **Close** on window level.

The sub-object 1T1_S1 is renamed automatically according the naming scheme and Control Builder Name aspect is updated.

If more than one copy is required, it is recommended that user copies and renames the function using Bulk Data Manager. Use the pre-configured worksheet BDM_for_Function Designer.xls (see Instantiating the Typical) or configure a user-defined worksheet.

Reconfigure

For all copied diagram references, the diagram variables of the source are still connected. User has to re-connect these diagram references in the sub-diagrams of the sequence according to Reconfigure Diagram References. It might also be necessary to adjust transition logic or action code. Finally re-allocate and re-generate the configuration data.

Downloading and Testing the Application

Download sequences as part of an application of the project according to Allocating, Generating Configuration Data, and Testing. For examples based on 1T1Seq1 see Test and Commissioning.

Use the following basic Function Designer functionalities for testing sequences:

- Subscribe for live data in the sequence related diagrams.
- Follow the signal flow through the diagram references.
- Navigate between parent and child diagrams.

Additional test functions available in the sequences toolbar are:

- Enables / Disables execution of the actions associated to the step.
- Enables / Disables state transition.

- Forces next step to become active.
- Forces previous step to become active.

These test functions are also available as a context menu on a step or transition.

- Use the Watch Window to watch and force variable properties.

Printing Documentation

Printing Diagrams and Sequences

Follow the steps to print all diagrams of the sequence:

1. Open diagram 1T1Seq1.
2. Click **File > Print**.
3. In the **Print** dialog ensure that the check boxes Diagrams and Child Diagrams are selected.
4. Ensure a fitting Page Setup is set.
5. Click **OK** in the Function Designer **Print** dialog, and also in the subsequent Windows **Print** dialog.

All diagrams and child diagrams are printed as one batch, starting with the Function Diagram containing the sequence, continued with the sequence overview diagram and followed by the step and transition related diagrams. Transition and step details are merged into one diagram each.

Creating and Printing Table of Contents

Follow the steps to create and print a leading table of contents:

1. Open diagram **1T1Seq1**.
2. Click **File > Create Contents**.
3. Select either radio option Full Contents or Overview Contents.
4. Select a contents template.
5. Click **OK**.

6. Leave the workbook tab showing the contents diagram open.
7. Click **File** > **Print**.
8. Ensure that the **Contents** check box is selected.
9. Continue with Step 3 of Printing Diagrams and Sequences.

A leading contents diagram is printed together with the sequence diagrams.

User needs to provide Functional Designation, Diagram Description and Page Comment in the diagrams before printing contents diagram.

Process Sequence2D Model

The process model used as an example describes a motor with a simple control logic implemented using sequence2D. It can be implemented based on control module types and function block types from the AC 800M Connect standard object type libraries and based on a project specific object type.

The process model used here is simplified and this tutorial shows implementation of some parts of it. Names not compliant to IEC 61131 may be used to indicate name mapping between Function Designer and Control Builder M.

Configuring Sequence2D Processes

Types of Sequences provides basic information on sequence support in Function Designer.

Configuring the Function Diagram

Sequence2D

Perform the following to create and configure the **Sequence2D** function block:

1. Create a parent object (ex: Motor Control) in Engineering Workplace, according to Creating a new Function Diagram.
2. Select **Sequence2D** from the **SPL** tab according to Inserting a Sequence\Sequence2D Component.

3. Select **SFC2DHeader** from the **Object Types** tab in the **Insert Objects** dialog by navigating to the following path:
Object Types\Control System\AC 800M/C Connect\ Libraries\SeqStartLib 1.2-3\Control Module Types
4. Click **Apply** in the **Insert Objects** dialog.
5. Accept or change the **Name** in the **New Component Name** dialog.
6. Click **OK**.
7. Click **Cancel** in the **Insert Objects** dialog.
8. Connect the **SFC2DHeader** and **Sequence2D** Function Blocks as shown in Figure 11.
9. Double-click **Sequence2D** to open the **Motor Control/Sequence2D_1** diagram in a workbook tab.
10. Reserve the opened diagram if it is unreserved.
11. Add two transitions (**Tr101** and **Tr102**) and complete the workflow as shown in Figure 43.

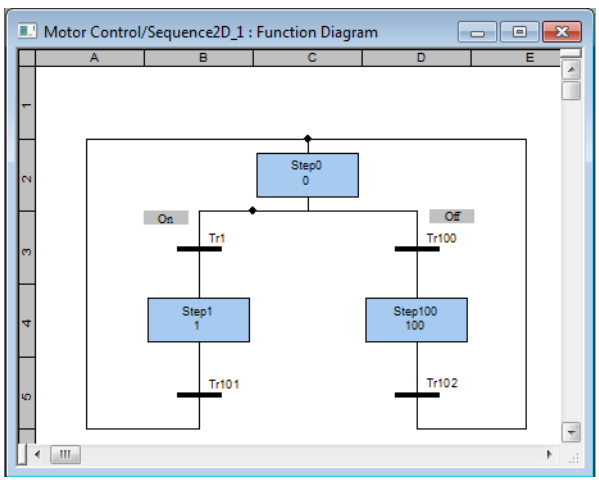
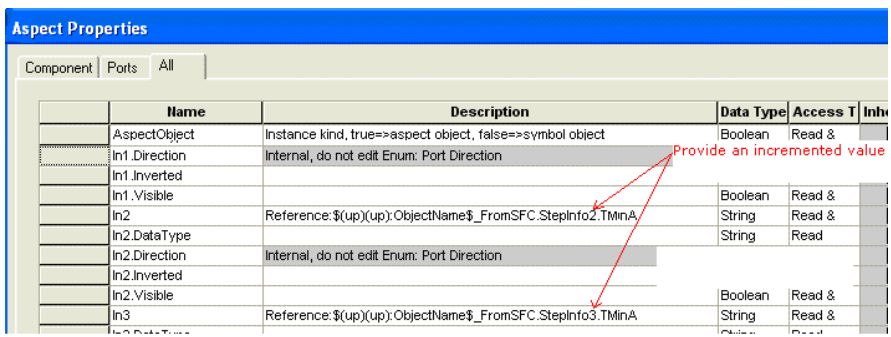


Figure 43. Overview Diagram

12. Double-click the transitions and set the following conditions:
    a. For transitions **Tr1** and **Tr100** - set the **In2** input of the **and(bool)** Function Blocks to 1.
    b. For transitions **Tr101** and **Tr102 -** set the **In2** of the **and(bool)** Function Blocks to 0.

In a simultaneous sequence where more than one Step is connected to the transition, the **AND** block needs to consider signals of all Steps connected to the transition. Hence, configure the Transition manually as explained below:

1. Double-click and open the Transition (ex. **Tr102**).
2. Right-click the **AND** block and select **Number Of Inputs…**.
3. In the **Number Of Inputs** dialog, increase the number of input ports in accordance with the number of Steps available in the simultaneous sequence.
4. Right-click the **AND** block and select **Aspect Properties**.
5. In the **Aspect Properties** dialog, select the **All** tab.
6. Copy and paste the description available for **In1** value to the other **Input** ports after incrementing the reference values as indicated in figure below. Click **OK** in the **Aspect Properties** dialog.



7. Double-click **Step1**, and configure the ON sequence of the motor.
8. Double-click the **Action** block and configure the following:

a. Click **Insert** > **Object** and select **TOn** from the following path: *Object Types\Control System\AC 800M/C Connect\ Libraries\BasicLib 1.6-5\Function Block Types*

b. Click **Apply** in the **Insert Objects** dialog.

c. Accept or change the **Name** in the **New Component Name** dialog.

d. Click **OK**.

e. Click **Cancel** in the **Insert Objects** dialog.

f. Configure the following for **TOn** as shown in Figure 44:

1.Connect the **In** input of **TOn** to the **StepName.X** port.

2.Set the required time delay (ex: 10s) for the **PT**.

3.Right-click the **Q** port and select **New Diagram Output Reference...** from the context menu.

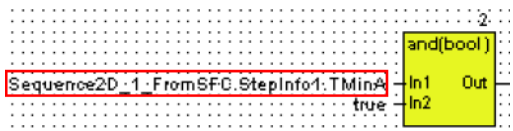
i. Accept or change the diagram reference **Name** in the **Variable Properties** dialog.

ii. Click **Connect...** to display the **Connect to Diagram Variable** dialog.

iii. Select the required communication variable from the list of variables, and click **OK**.

iv. Click **OK** in the **Variable Properties** dialog.

Default connection **Sequence2D_1_FromSFC.StepInfo1.TMinA** should not be deleted in a Transition2D.



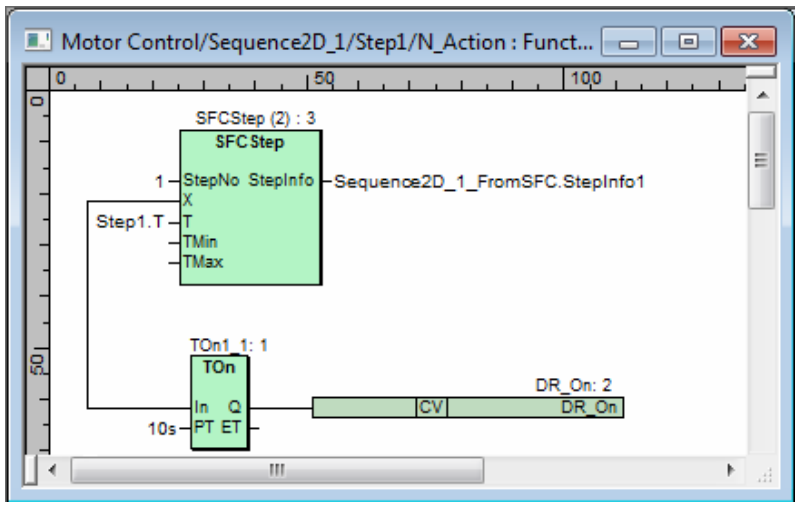
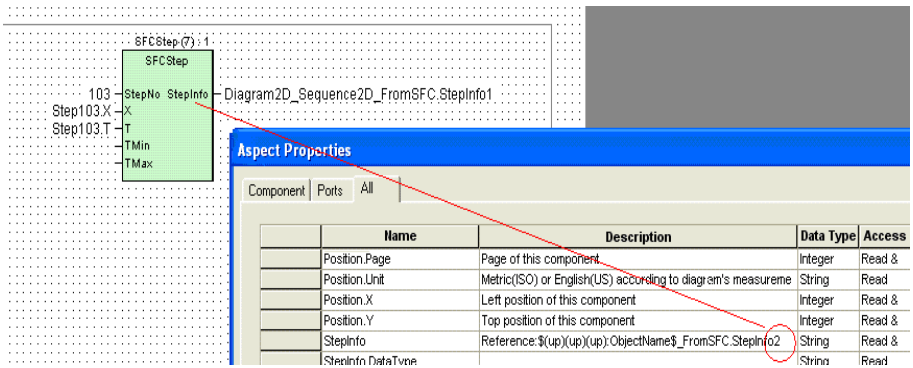


Figure 44. Configuration of Timer

In a simultaneous sequence, the **StepInfo** of the SFCStep object needs to be configured manually as explained below:

1. Right-click **SFCStep** object and select **Aspect Properties**.
2. Select the **All** tab.
3. In the **Description** column of **StepInfo** property, increment the reference value as indicated in figure below:



4. Click **Apply** and click **OK**.

22. Double-click **Step100** and repeat Step 8 to configure the OFF sequence of the motor.

Motor Logic

Perform the following to configure the Motor logic:

1. Create an object (ex: **Motor**) in Engineering Workplace, according to Creating a new Function Diagram.
2. Insert **MotorUniM** Function Block according to Inserting Function Components.
3. Right-click **MotorUniM**, and select **Show Hidden Ports…** from the context menu.

4. Select the required ports such as FB0, FB1, SetAuto, etc in the **Show Hidden Ports** dialog.
5. Click **OK**.
6. Set the value of **SetAuto** as 1.
7. Right-click **Out1** and select **New IO Signal DOS...** from the context menu.
8. Accept or change the **Name** in the **New Component Name** dialog.
9. Click **OK**.
10. Repeat Step 7 to Step 9 to configure **Out2**.
11. Right-click **FB1** and select **New IO Signal DIS...** from the context menu.
12. Accept or change the **Name** in the **New Component Name** dialog.
13. Click **OK**.
14. Repeat Step 11 to Step 13 to configure **FB2** as shown in Figure 45.

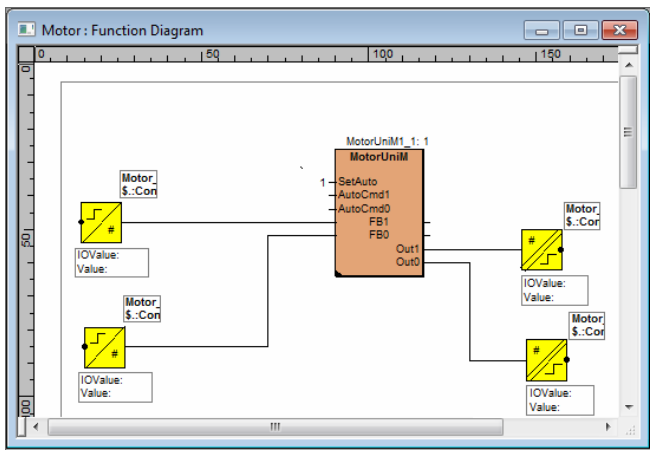


Figure 45. Configurations for MotorUniM Function Block

15. Right-click **AutoCmd0** port of **MotorUniM** Function Block and select **New Diagram Input Reference** from the context menu.

16. Accept or change the diagram reference **Name** in the **Variable Properties** dialog.
17. Click **Connect...** to display the **Connect to Diagram Variable** dialog.
18. Select the required communication variable from the list of variables, and click **OK**.
19. Click **OK** in the **Variable Properties** dialog.
20. Right-click **AutoCmd1** port and select **New Diagram Input Reference** from the context menu.
21. Repeat Step 16 to Step 19 to configure **AutoCmd1** port as in Figure 46.

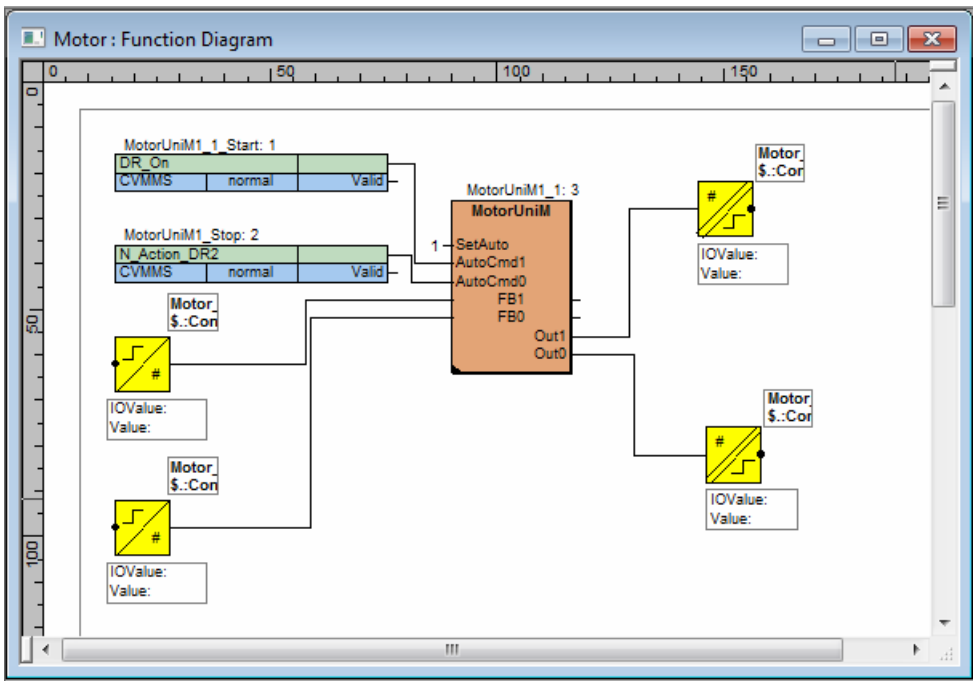


Figure 46. Complete Configuration of MotorUniM Function Block

IO Allocation

Allocate the Function Diagram to an application in Control Builder M according to Allocating a Function Diagram.

Insert the required Hardware Libraries/IO Boards according to Extending Control Structure. Allocate IO signals to the Function Diagram according to Allocating I/O. The allocated Function Diagram appears as shown in Figure 47.

Generate configuration data according to Generating Configuration Data.

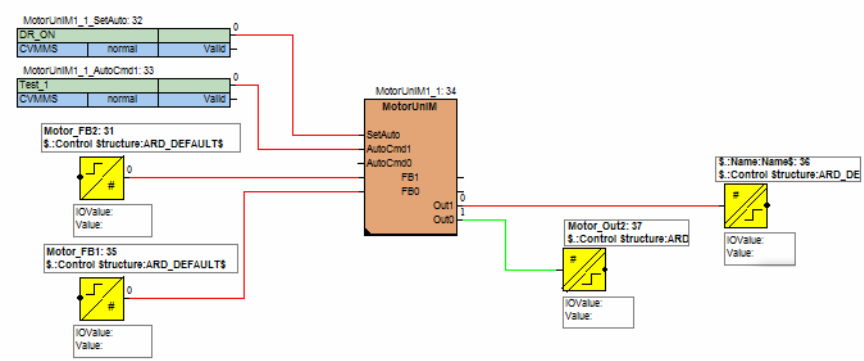


Figure 47. Sequence2D Controlled Motor Logic Function Diagram

Downloading and Testing the Application

Download sequences as part of an application of the project according to Allocating, Generating Configuration Data, and Testing. To test and commission the application, refer to the details available in Test and Commissioning. To print the sequence2D Function Diagram, refer to the details available in Printing Documentation.

Section 4 Test and Commissioning

Downloading to SoftController:

User has already generated configuration data. Use SoftController to perform a first download and test of the application:

1. In the opened project of Project Explorer, right-click on **Controller_1** and click **System Identity**.
2. Every controller must have a unique system identity.
   Enter 127.0.0.1:2 as the unique system identity of Controller_1.
   Or enter <IP address of the relevant workplace>:2.
3. Click **OK**.
4. Right-click Controller_1 again.
5. Click **Simulate Hardware**.
6. Start SoftController x.x.x:
   Start > All Programs > ABB Industrial IT 800xA > Control and IO > SoftController x.x > SoftController x.x.
   Or double-click the SoftController x.x.x shortcut icon if it exists on the desktop.
7. In the **SoftController** window, click **Start**.
8. Double-click the **Ethernet** hardware unit.
9. In the **Settings** workbook tab enter 127.0.0.1 as the value for the IP address.
10. In Project Explorer, click **Download Project and Go Online**.
11. Click **Continue** in the required dialogs, to load the project.

Follow the steps to start OPC server and connect the controller to it:

1. To start OPC Server for AC 800M x.x.x click
Start > All Programs > ABB Industrial IT 800xA > Control and IO > OPC Server for AC 800M x.x.x > OPC Server for AC 800M x.x.x.
Or double-click the OPC Server for AC 800M x.x.x shortcut icon if it exists on the desktop.
2. In the **Data Access** tab of OPC Server Configuration dialog enter the Controller Identity (127.0.0.1:2 or <IP address of the relevant workplace>:2, refer to example above).
3. Click **Connect**.
4. Repeat Step 2 and Step 3 for the **Alarm and Event** tab of **OPC Server Configuration** dialog.

Displaying Live Values

Follow the steps to show live values in diagram 1T1VC1:

1. In Engineering Workplace, navigate to and double-click on the Function aspect of object 1T1VC1 to open the diagram.
2. In Function Designer, click **Online > Subscribe for Live Data All Output Ports** to get online values.
3. Right-click 1T1VC1_V1.
4. Click MainFaceplate to open the faceplate of component 1T1VC1_V1.

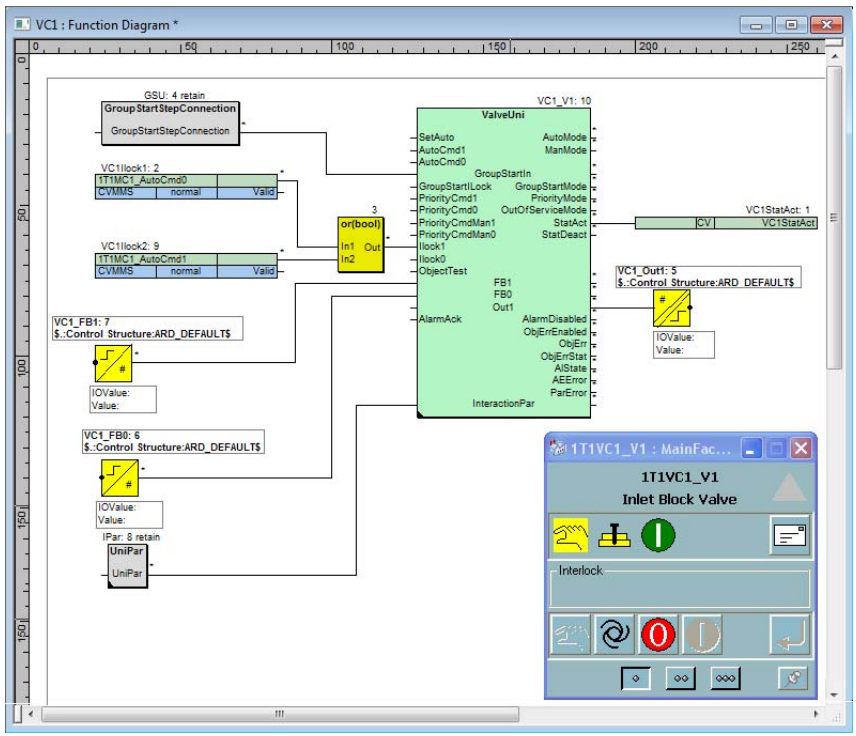


Figure 48. Function Diagram Online, MainFaceplate of 1T1VC1_V1 Opened

5. Click **Activate valve** to open it.
6. Right-click Diagram Reference 1T1VC1_StatAct.
7. Click **Goto Reference** to open diagram 1T2VC1.
8. Check that Ilock1 is true because 1T1VC1_V1 is opened and the MMS communication is working.
9. Switch back to diagram 1T1VC1 using the corresponding workbook tab.
10. Close 1T1VC1_V1 using the MainFaceplate and check again if now 1T2VC1_V1 valve can be opened.

Using Watch Window

The Watch Window of Function Designer allows to observe online values of variables (OPC values) for the opened diagrams and also to observe online values which are not included in the open diagrams. It also allows to set or force values of these variables.

Preparing the Window and Watching Values

In the diagram 1T1VC1 displaying online values already:

1. Click **Online > Watch Window**.
2. Re-position the window and resize the columns according to ones requirements.
3. Click **Add** in the toolbar of the Watch Window.
4. In the dialog **Add Variable to Watch Window**:
   (This is an example on how to receive variables not related to the opened diagram.)
   a. In the Functional Structure of the structure browser in the left pane navigate to and select object 1T1FC1_FIC.
   b. In the right pane navigate to and select InteractionPar.FacePlate.ExternalSp.
   c. Delete the string except for InteractionPar
   d. Click **Apply** and click **Close**.
   e. Click to Save Watch Window Data to any Function Online Data aspect in the toolbar of the Watch Window.
   f. In the **Save To Function Online Data** dialog, enter for example WW1T1VC1 as the aspect name and click **OK**.
5. Click **File > Open** and open diagram 1T1FC1 additionally.
6. Click **Open** (Load Watch Window with data from Function Online Data aspect).
7. In the **Open Function Online Data Aspect** dialog, navigate to WW1T1VC1 aspect and click **OK**.

8. Add additional variables:
   (This is an example on how to retrieve variables related to the opened diagram).
   a. Right-click the output port of the input signal object 1T1FC1_In and click Add to Watch Window.
   b. Do the same for the input port of the output signal object 1T1FC1_Out.
9. Navigate to the value cells of the three structured variables which are now included into the Watch Window and have a look at the current values.
10. Right-click PID controller 1T1FC1_FIC.
11. Click MainFacePlate and observe the set point value SP.

Setting Variable Values

Follow the steps to set the variable values:

1. In the Watch Window:
   a. Navigate to 1T1FC1::InteractionPar.Faceplate.SPManValue.
   b. In the **Prepared Value** cell of this row enter a changed setpoint value.
   c. Click **Activate**.
   d. Click **Yes** in the Warning query box to perform the write operation.
2. Observe whether the setpoint value is updated in the faceplate.

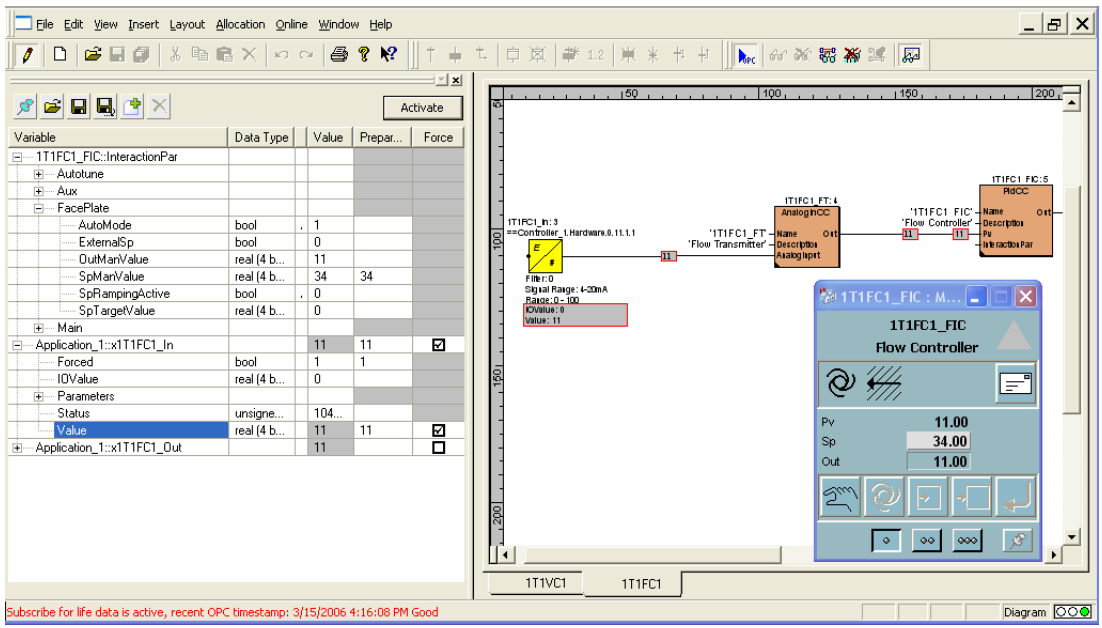


Figure 49. Watch Window and Function Diagram 1T1FC1 On-Line

3. In the Watch Window:
   a. Navigate to the cell **Prepared Value** for x1T1FC1_In.Value.
   b. Enter a changed value.
   c. Select the **Force** check box in the cell to the right of the **Prepared Value**.
   d. Click **Activate**.
   e. Click **Yes** in the Warning query box to perform the write operation.
4. Observe if the value is updated in the Function Diagram 1T1FC1.
5. To end the Watch Window session:
   a. Click left most **Save (**Save Watch Window Data to the Function Online Data aspect from which it was loaded**).**

b. Click **Close** on Watch Window level.

CBM Views

Online editors of Control Builder M can be invoked from the Function Diagram using the context menu of a Function Component.

Example: Force I/O signal in diagram 1T1FC1 in online mode.

1. Right-click the signal object 1T1FC1_In.
2. Click **Control Structure**.
3. In the structure browser of the Control Structure, click the I/O board object.
4. In the aspect list of the Control Structure window, double-click the Hardware Unit aspect of the board object.
5. Use the **Status** tab of the hardware editor window to force the I/O value.

Process Sequences On-Line

Displaying Live Values

User can subscribe for live data in Function Designer for the following:

- the sequence overview diagram, see Figure 50the first figure below.
- the step sub-diagram, see the second figure below.
- the action sub-diagram, see the third figure below.
- the transition sub-diagram, see the fourth figure below.
- the non-sequence diagrams (for example 1T1VC1 and 1T1LC1).

and view the online values representing the current status of the sequence and their sources and sinks.

To follow the signal flow use **Goto Reference**, **Goto Child Diagram**, and **Goto Parent Diagram** (context) menu items.

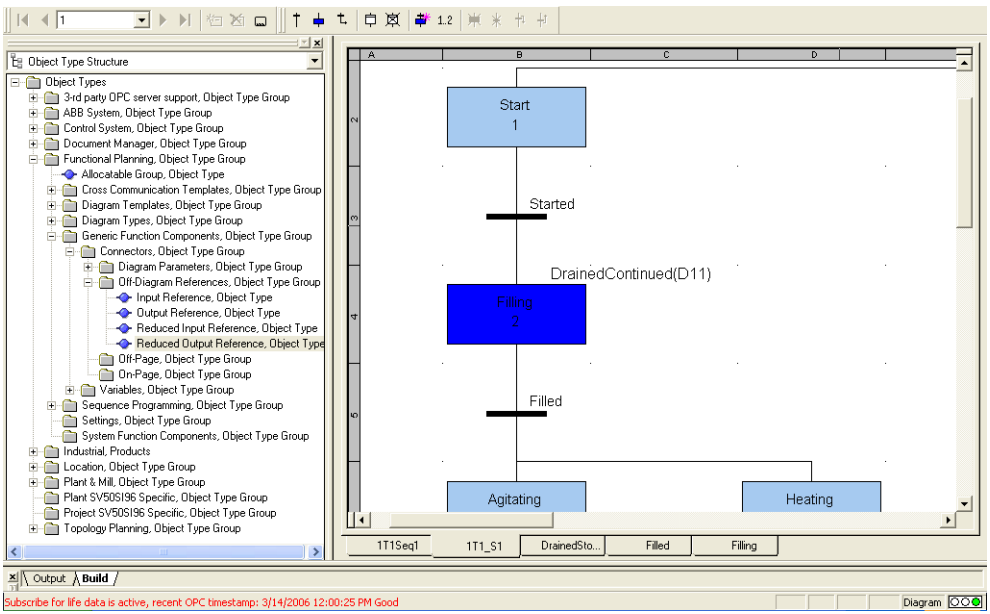


Figure 50. Overview Diagram On-line, Step Filling Active

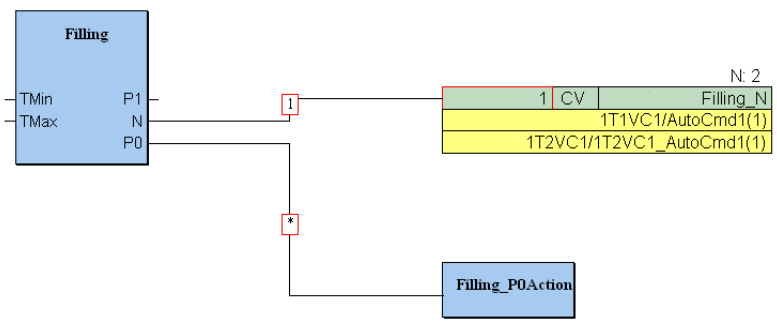


Figure 51. Step Filling, Detail Diagram Online, N Action

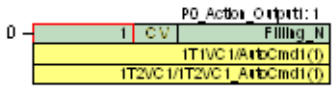


Figure 52. Step Filling, P0 Action Diagram Online

Figure 53. Transition Filled, Detail Diagram Online

Single-Stepping a Sequence

Follow the steps to single step a sequence in overview diagram:

1. Select the transition after the active step.
2. Click **Force Forward** in the Sequence toolbar
   or right-click the transition and click **Force Forward** from the available context menu. The step following the transition becomes active.
3. Repeat the previous two steps for the next transitions to step through the sequence.

Using **Force Backward** user can go back in the sequence, starting on the transition before the active step.

User can also use the Watch Window to step through a sequence: Add the transition variables to the Watch Window and force the corresponding variable components from this window, see the figure below.

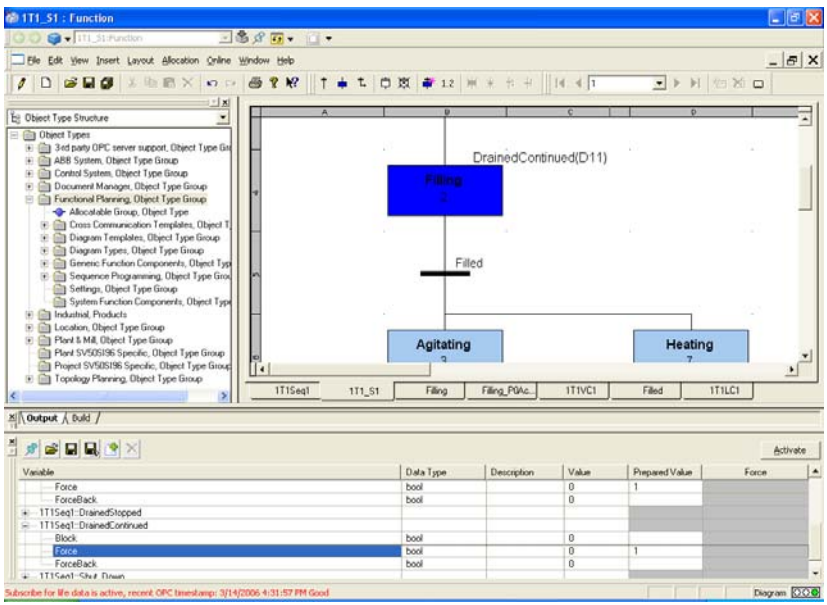


Figure 54. Watch Window Used to Step Through Sequence

Stopping a Sequence

To disable the actions of a selected step user can use the **Disable Actions** context menu item or the corresponding sequence toolbar button, see the figure below.

To stop the sequence, user can block a transition by using the **Block Transition** context menu or the corresponding sequence toolbar button on the selected transition, see the figure below.

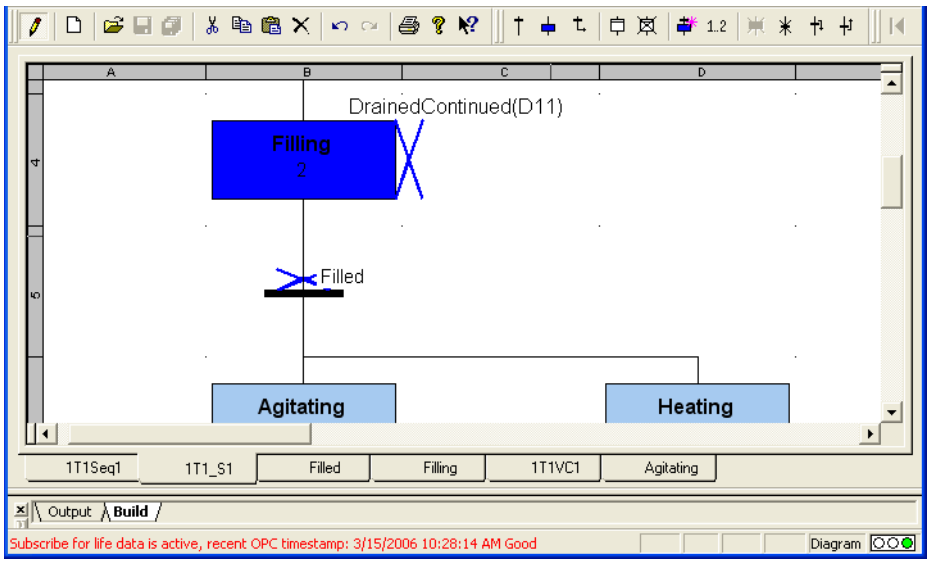


Figure 55. Step with Disabled Actions and Blocked Transition

Index

A

Action 46, 56
Action diagram 59, 117
Allocatable Group 35
Aspect Properties 90

B

BDM_for_Function_Diagrams 43, 79
Block Transition 140
Bulk Data Manager 43, 63, 69, 72, 106

C

CBM_Signal object 37
Child Diagram 34, 61
component
    context menu 22
Component View 22
Connecting the ports 30
Connection 30
Contents Diagram 41
Control module object type 84
Control Structure 66
Copy and rename 104, 118
Cross reference information 32

D

Data Flow Order 36
diagram
    function 21
Diagram states 44
Diagram Variable 31
Diagram view 14
Disable Action 140
Disable action 62, 119
Download and test 119, 129, 131
drag and drop 22

E

Engineering portal 13

F

Force 62, 120, 139
Force a value 40
Force Backward 139
Force Forward 139
Function aspect 14, 20, 72
Function Component 14, 26, 72, 75, 86, 100
Function Designer 13, 20, 23 to 24, 63
Function Diagram 14, 20, 25, 41, 46, 70 to 72, 75, 83
Functional Planning 13
Functional Structure 70, 72

G

Generate Configuration Data 39
Generate configuration data 109
Graphical Symbol 96

H

Hidden port 28

I

IEC 61131-3 Sequence 46
Input Parameter 32
Input Reference 32
IO Allocation 37, 107

L

Live Data 40

O

Object type 82
Object type library 83
Object Type Structure 85
Online value 137
Output Diagram Reference 33 to 34
Output Parameter 32
Output Reference 32
Overview diagram 49, 113, 115

P

P0 Action 56
P1 Action 56
Page 29
Page Comment 29
Page reference 33
Parent Diagram 34, 61
Port Documentation 41
preview window 20
Print documentation 41, 120
Process Function 63
Process function 65
Process model 63
Process Sequence 63
Process sequence 113

S

Sequence 46, 113
Signal Group Object Type 93
Signal Information 94
Step 46, 50
Step detail diagram 56, 117
System Extension 18

T

Transition 46, 50
Transition detail diagram 60, 118
Transition with jump 51
Typical 75

V

Variable Properties 32

W

Watch Window 40, 120, 134, 140

Revision History

This section provides information on the revision history of this User Manual.

The revision index of this User Manual is not related to the 800xA 5.1 System Revision.

The following table lists the revision history of this User Manual.

Revision Index	Description	Date
-	First version published for 800xA 5.1	June 2010
A	Updated for 800xA 5.1 Rev A	May 2011
B	Updated for 800xA 5.1 Rev B	June 2012
C	Updated for 800xA 5.1 FP4	February 2013
D	Updated for 800xA 5.1 Rev D	December 2013

Updates in Revision Index A

The following table shows the updates made in this User Manual for 800xA 5.1 Rev A.

Updated Section/Sub-section	Description of Update
Section 2, Working with Function Diagram	Added Information note on Function Designers project.
Section 2, Connecting Function Components	Added in information note about blob size.

Updated Section/Sub-section	Description of Update
Section 2, Navigating Through Function Diagram	Added new information note on how diagram parameters can be used.
Section 2, Configuring Data Flow Order	Added information note on allocation of Function Diagram.
Section 2, Printing Documentation	Added a new section *Printing Documentation*

Updates in Revision Index B

The following table shows the updates made in this User Manual for 800xA 5.1 Rev B.

Updated Section/Sub-section	Description of Update
Section 2, Working with Function Diagram	Added Information note regarding connecting diagram parameters.
Section 2, Working with Function Diagram	Added information note regarding SFC2DHeader
Section 3, Configuration	Added procedure to Procedure to print template information in signal group.
Section 3, Configuration	Added information regarding Default connections.

Updates in Revision Index C

The following table shows the updates made in this User Manual for 800xA 5.1 FP4.

Updated Section/Sub-section	Description of Update
Section 2, Basic Operation	Added information regarding Auto Sort Order.
Section 3, Configuration	Modifed workflow for creating a signal group object type.

Updates in Revision Index D

The following table shows the updates made in this User Manual for 800xA 5.1 Rev D.

Updated Section/Sub-section	Description of Update
Section 2, Overview Diagram	Added new information for Reference Diagram explaining the SPL error.

Contact us

ABB AB
Control Technologies
Västerås, Sweden
Phone: +46 (0) 21 32 50 00
e-mail: processautomation@se.abb.com
www.abb.com/controlsystems

ABB Automation GmbH
Control Technologies
Mannheim, Germany
Phone: +49 1805 26 67 76
e-mail: marketing.control-products@de.abb.com
www.abb.de/controlsystems

ABB S.P.A.
Control Technologies
Sesto San Giovanni (MI), Italy
Phone: +39 02 24147 555
e-mail: controlsystems@it.abb.com
www.abb.it/controlsystems

ABB Inc.
Control Technologies
Wickliffe, Ohio, USA
Phone: +1 440 585 8500
e-mail: industrialitsolutions@us.abb.com
www.abb.com/controlsystems

ABB Pte Ltd
Control Technologies
Singapore
Phone: +65 6776 5711
e-mail: processautomation@sg.abb.com
www.abb.com/controlsystems

ABB Automation LLC
Control Technologies
Abu Dhabi, United Arab Emirates
Phone: +971 (0) 2 4938 000
e-mail: processautomation@ae.abb.com
www.abb.com/controlsystems

ABB China Ltd
Control Technologies
Beijing, China
Phone: +86 (0) 10 84566688-2193
www.abb.com/controlsystems

3BDS100968-510 D

Power and productivity
for a better world™