Safety Device Library

Release v1.04

Allen-Bradley
by ROCKWELL AUTOMATION

Reference Manual

Original Instructions

Important User Information

Read this document and the documents listed in the additional resources section about installation, configuration, and operation of this equipment before you install, configure, operate, or maintain this product. Users are required to familiarize themselves with installation and wiring instructions in addition to requirements of all applicable codes, laws, and standards.

Activities including installation, adjustments, putting into service, use, assembly, disassembly, and maintenance are required to be carried out by suitably trained personnel in accordance with applicable code of practice.

If this equipment is used in a manner not specified by the manufacturer, the protection provided by the equipment may be impaired.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Throughout this manual, when necessary, we use notes to make you aware of safety considerations.

WARNING: Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.

ATTENTION: Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you identify a hazard, avoid a hazard, and recognize the consequence.

IMPORTANT Identifies information that is critical for successful application and understanding of the product.

These labels may also be on or inside the equipment to provide specific precautions.

SHOCK HAZARD: Labels may be on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.

BURN HAZARD: Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.

ARC FLASH HAZARD: Labels may be on or inside the equipment, for example, a motor control center, to alert people to potential Arc Flash. Arc Flash will cause severe injury or death. Wear proper Personal Protective Equipment (PPE). Follow ALL Regulatory requirements for safe work practices and for Personal Protective Equipment (PPE).

The following icon may appear in the text of this document.

Identifies information that is useful and can help to make a process easier to do or easier to understand.

Chapter 7

DCSTL - Dual Channel Input Stop with Test and Lock (raC_Dvc_DCSTL)

Chapter 8

DCSTM - Dual Channel Input Stop with Test and Mute (raC_Dvc_DCSTM)

Chapter 9

CROUT - Configurable Redundant Output (raC_Dvc_CROUT)

Chapter 10

TSAM - Two Sensor Asymmetrical Muting (raC_Dvc_TSAM)

Chapter 11

TSSM - Two Sensor Symmetrical Muting (raC_Dvc_TSSM)

Chapter 12

FSBM - Four Sensor Bi-directional Muting (raC_Dvc_FSBM)

Chapter 13

THRSe - Two Hand Run Station Enhanced (raC_Dvc_THRSe)

Chapter 14

SMAT - Safety Mat (raC_Dvc_SMAT)

Chapter 15

SFX - Safety Feedback Interface (raC_Dvc_SFX)

Chapter 16

SS1 - Safe Stop 1(raC_Dvc_SS1)

Chapter 17

SS2 - Safe Stop 2 (raC_Dvc_SS2)

STO - Safe Torque OFF (raC_Dvc_STO)

Chapter 18

Safety Group Monitor and Safety Group Members (raC_Opr_SafetyGroupMonitor, raC_Opr_SafetyGroupMember)

Chapter 19

Device Object Libraries Overview

Our Device Object Libraries enable you to easily interface with Rockwell Automation® intelligent devices like drives, motion, network switches, sensors, IO and more. The libraries contain tested, documented, and lifecycle-managed objects which can be used with machine builder, process, and packaged libraries or as standalone components. Device objects include HMI faceplates for FactoryTalk® View ME/SE and Studio 5000 View Designer® software and provide a user interface that seamlessly integrates with the products.

HMI faceplates are standard display files that provide a common user interface. These are HMI pop-up screens used to display detailed information related to a specific instruction or device. In systems that follow ISA 101.1 design guidelines, faceplates are often referred to as Level 4 displays.

Pre-configured Device Objects include an Add-On Instruction Rung/Routine and an HMI Faceplate providing the following benefits:

- Collect, Process and Deliver Data between Smart Devices and Application Logic
- Detailed Device Data Collection and Delivery
- Enhanced Device Status and Diagnostics
- Common Control Interfaces maximizing Flexible Automation Device Selection & Application Code Reuse

Device Object Use Cases:

- Basic Device Maintenance and Diagnostics
- Virtual Device Operations for Startup and Commissioning
- Operator and Program Control for Velocity Machine and Process Applications

Device Object Libraries may be downloaded from the Product Compatibility and Download Center. Search for "Library".

Application Code Manager

Studio 5000® Application Code Manager is a tool that can be used with Device Object Libraries to streamline project and machine development. This bulk coding tool allows you to easily design and standardize functionality with reusable application code.

Enable more efficient project development with reusable libraries of code:

- Quickly create and deploy projects through our Application Content Libraries
- Import Rockwell provided application content libraries to expedite system development

- Build your own reusable code that can be managed and deployed across your entire enterprise
- Easily configure objects in bulk with reusable code to increase application development, no additional programming is necessary
- Consolidate content for Studio 5000 Logix Designer®, FactoryTalk® View Studio, FactoryTalk® Alarms & Events, FactoryTalk® Historian to configure an object a single time and generate content for each of those software packages.

See the section on Using the Library with Application Code Manager for more details.

Other Application Code Libraries

This Device Object Library may be used in harmony with other Application Code Libraries including other Device Object Libraries (Network, IO, IO-Link, Safety Device Libraries) or Application Libraries (PlantPAx® Process Objects library, Machine Builder Libraries). All libraries are intended to follow similar design philosophies to provide a consistent experience for operators and maintenance staff.

A complete list of Application Code Libraries from Rockwell Automation® follows.

Item	Description
PlantPAx® Process Library	Rockwell Automation® Library of Process Objects provides application templates, Endress + Hauser library objects, Application Code Manager library objects, and tools and utilities for PlantPAx® DCS applications. Includes the following: • Graphics for built-in instructions • HMI images and Help files • Logix diagnostic objects • Process objects • Control strategies • Sequencer objects • PlantPAx® Configuration Tools for Tags, Alarms and Historian • Color Change • Historian -- Asset Framework template and objects
Machine Builder Libraries	Tested, documented and life-cycle managed library objects and faceplates for use with Studio 5000® Application Code Manager for use primarily with OEM and discrete machine applications.
Common Application Libraries	Commonly used application library objects and faceplates for use with Studio 5000® Application Code Manager including basic functions like unit conversion and data collection.
Independent Cart Technology Libraries	ICT Libraries for iTRAK® and MagneMotion® including MagneMover® Lite™ and QuickStick® for Studio 5000® Application Code Manager
I/O Device Library	Provides objects for Rockwell Automation® 1756, 1769, 1734, 1794, 1738, 1732E, 1719, 5069, 5094 I/O modules including pre-configured status and diagnostic faceplates
IO-Link Device Library	Provides IO-Link master and sensor objects including pre-configured status and diagnostic faceplates
Network Device Library	Provides objects for Stratix® switch and Device Level Ring network objects
Power Device Library	Provides objects for discrete, velocity, motion, and power monitor devices
Safety Device Library	Provides safety objects to interface with safety I/O
Electrical Protection Device Library	Provides a standard to represent protection devices within your electrical distribution system

Libraries can be accessed from the Product Compatibility and Download Center.

Software and Firmware Upgrades

When you update software or firmware revisions, we recommend that you verify the impact on performance and memory utilization before implementing the upgrade on the production system. For FactoryTalk® View or ControlLogix® platforms, we recommend that you review the release notes and verify the impact of the upgrade on performance and memory utilization.

You can also verify the compatibility of the upgrade with the installed software and operating systems in use on your system. See the Product Compatibility and Download Center.

Rockwell Automation® Services and Support

System Support offers technical assistance that is tailored for control systems. Some of the features include the following:

- Highly experienced team of engineers with training and systems experience
- Use of online remote diagnostic tools
- Access to otherwise restricted TechConnect℠ Knowledgebase content
- 24-hour, 7 days per week, 365 days per year of phone-support coverage upgrade option

For more information, contact your local distributor or Rockwell Automation® representative or see http://www.rockwellautomation.com/support.

You can view or download publications at http://www.rockwellautomation.com/literature. To order paper copies of technical documentation, contact your local Allen-Bradley distributor or Rockwell Automation® sales representative.

Rockwell Automation® Safety Device Library

The Safety Device Library is a tested, documented, and life cycle managed object library. The Device Library provides pre-configured status and diagnostic faceplates and AOI sets for Rockwell Automation® safety instructions and devices. The Safety Device Objects may be used with Machine Builder, Process, and Packaged Libraries or as standalone components. Safety Device Library add-on instructions objects collect, process, and deliver data between hardware devices and application logic.

The Safety Device Library includes Add-On Instructions (AOIs) and HMI Faceplates for safety applications including Dual Channel Input Stop (DCS/DCSTL/DCSTM), Configurable Redundant Output (CROUT), Two Sensor Asymmetrical/Symmetrical Muting (TASM/TSSM), Four Sensor Bi-directional Muting (FSBM), Two Hand Run Station Enhanced (THRSe) and Safety Mat (SMAT). These safety applications use Allen-Bradley safety IO modules. Safety Group Monitor and Safety Group Member instructions are also provided to easily group up to 32 safety I/O devices for monitoring and event logging.

These Safety Device Library add-on instructions are to be used as supplemental functionality to the pre-defined Safety instructions available in safety class GuardLogix® and Compact GuardLogix® controllers such as DCS, DCSTL, DCSTM, CROUT, SMAT, TSAM, TSSM, FSBM, THRSe, SS2, SS1, SFX etc. The safety class controllers and pre-defined Safety instructions are mandatory when using the Safety Device Library instructions.

Additionally, this release of the Safety Device Library v1.04 includes updated versions of safety class I/O modules files for use with Studio 5000® Application Code Manager. These are required when using some Safety Device Library instructions in Application Code Manager. These I/O module files are normally found in the IO Device Library which is also available as a free download from the Product Compatibility and Download Center.

This document includes the functional requirements of the safety instruction and safety group monitor device objects.

The Safety Device Library may be downloaded from the Product Compatibility and Download Center. Search for Safety Device Library.

Compatibility

Compatible Software

- Studio 5000 Logix Designer® v30.01 or later for PAC Application Development
- Studio 5000® Application Code Manager v4.0 or later and later for bulk code configuration
- Studio 5000 View Designer® v5.02 and later for PanelView™ 5000 Application Development
- FactoryTalk® View Studio v10 and later for PanelView™ Plus Application and FactoryTalk® View SE Application Development

Compatible Hardware

- PanelView™ 5500/5510/5310 with v5 or later firmware
- PanelView™ Plus with v10 or later firmware
- GuardLogix® 5580/GuardLogix® 5570/Compact GuardLogix® 5370/ Compact GuardLogix® 5380 controller with v30.001 or later firmware
- Safety device e.g. e-stop, light curtain, or safety gate, safety mat, gate lock, etc
- Safety IO Modules; detailed compatibility chart below.

I/O Module	Module Definition Input Status	Module Definition Output Data/ Muting Lamp	DCS	DCSTL	DCSTM	THRSe	TSAM	TSSM	FSBM	SMAT	CROUT
1732ES-IB12XOB4	Pt. Status Muting	Safety	X	X	X	X					X
	Pt. Status - Muting - Test Output	Combined	X	X	X	X	X	X	X	X	X
1732ES-IB12XOBV2	Pt. Status Muting	Safety	X	X	X	X					X
	Pt. Status - Muting - Test Output	Combined	X	X	X	X	X	X	X	X	X
1732ES-IB16	Pt. Status - Muting - Test Output	Test	X	X	X	X	X	X	X	X	
1732ES-IB8XOB8	Pt. Status Muting	Safety	X	X	X	X					X
	Pt. Status - Muting - Test Output	Combined	X	X	X	X	X	X	X	X	X
1732ES-IB8XOBV4	Pt. Status Muting	Safety	X	X	X	X					X
	Pt. Status - Muting - Test Output	Combined	X	X	X	X	X	X	X	X	X
1734-IB8S	Pt. Status	None	X	X		X					
	Pt. Status-Power-Muting-Test	Test	X	X	X	X	X	X	X	X	
1734-OB8S	Pt. Status-Readback	Safety									X
1734-OBV2S	Pt. Status-Readback	Safety									X
1791ES-IB16	Pt. Status	None	X	X		X					
	Pt. Status - Muting - Test Output	Test	X	X	X	X	X	X	X	X	
1791ES-IB8XOBV4	Pt. Status Muting	Safety	X	X	X	X					X
	Pt. Status - Muting - Test Output	Combined	X	X	X	X	X	X	X	X	X
1756-IB16S	-	-	X	X		X					
1756-OBV8S	-	-									X
5069-IB8S/K	-	None	X	X	X	X					
	-	Muting Lamp Points Point 02 and 03-	X	X	X	X	X	X	X	X	
5069-OBV8S	-	-									X

I/O Module	Module Definition Input Status	Module Definition Output Data/ Muting Lamp	DCS	DCSTL	DCSTM	THRSe	TSAM	TSSM	FSBM	SMAT	CROUT
5094-IB16S/XT	-	-	X	X	X	X					
5094-OB16S/XT	-	-									X
5094-OW4IS/XT	-	-									X

Summary of Changes

This publication contains the following new or updated information. This list includes substantive updates only and is not intended to reflect all changes.

Topic	Page
Reference manual entirely reformatted and re-written	all
Added New Safety Motion Instruction Device Object - SFX, SS1, SS2, STO	193- 250

Footprint

Each instruction requires memory footprint within the Logix controller. The following characteristics apply:

- **Definition:** Estimated memory required to store the object definition, including all dependents
- **Instance:** Estimated memory required per object instantiated.
- **Execution (L84ES):** Estimated execution time / scan footprint evaluated in 1756-L84ES.

Device Object Footprint

Device Object	Standard Definition (kB)	Safety Definition	Safety Instance (kB)	Execution (µs)
raC_Dvc_DCS	1.90	5.37	1.18	13
raC_Dvc_DCSTL	1.91	5.99	2.28	14
raC_Dvc_DCSTM	1.75	5.91	2.18	13
raC_Dvc_CROUT	1.85	5.24	1.82	16
raC_Dvc_TSAM	1.92	5.96	1.86	25
raC_Dvc_TSSM	1.90	5.97	1.86	21
raC_Dvc_FSBM	2.01	6.13	2.1	21
raC_Dvc_THRSe	1.86	5.36	1.86	16
raC_Dvc_SMAT	1.80	5.25	1.75	16
raC_Dvc_SS2	3.12	10.58	1.6	42
raC_Dvc_SS1	2.61	7.86	1.13	33
raC_Dvc_SFX	2.61	8.10	0.86	34
raC_Dvc_STO	2.99	6.26	0.54	33
raC_Opr_SafetyGroupMonitor	3.56	12.30	3.92	85
raC_Opr_SafetyGroupMember	2.63	7.30	2.89	14

Additional resources

For a complete understanding of Rockwell Automation® safety applications it is recommended to review the following documents.

Resource	Description
GuardLogix® Safety Application Instruction Set, publication 1756-RM095	This manual describes the Rockwell Automation® GuardLogix® Safety Application Instruction Set (e.g. DCS, CROUT, TSSM, TSAM, etc), which is type-approved and certified for safety-related function in applications up to and including Safety Integrity Level (SIL) 3 according to IEC61508, and Performance Level, PLe (Cat.4), according to ISO13849-1.
Rockwell Automation® Safety Function Documents	Collection of safety function examples provided to minimize risk and speed up development. Examples include set-up and wiring, bill of materials, wiring schemes, configuration, verification and validation plan, and calculation of performance level.
ControlLogix 5580 and GuardLogix® 5580 Controllers User Manual, publication 1756-UM543	Provides information on how to install, configure, and program ControlLogix and GuardLogix® 5580 controllers in the Logix Designer application.
CompactLogix™ 5380 and Compact GuardLogix® 5380 User Manual, publication 5069-UM001.	Provides information on how to install, configure, and program CompactLogix and Compact GuardLogix® 5380 controllers in the Logix Designer application.
GuardLogix® 5570 Controllers User Manual, publication 1756-UM022.	Provides information on how to install, configure, and program the GuardLogix® 5570 controllers in the Logix Designer application.
GuardLogix® 5570 Controllers Reference Manual, publication 1756-RM099.	Contains detailed requirements for how to achieve and maintain SIL 3 with the GuardLogix® 5570 controller system in a Logix Designer application.
Compact GuardLogix® 5370 Controllers User Manual, publication 1769-UM022.	Provides information on how to install, configure, and program Compact GuardLogix® 5570 controllers in the Logix Designer application.
GuardLogix® 5580 and Compact GuardLogix® 5380 Controller Systems Safety Reference, publication 1756-RM012.	Describes the development, operation, and maintenance of GuardLogix® 5580 and Compact GuardLogix® 5380 controller-based safety systems in the Logix Designer application.

Library Components

The Safety Device Library is a tested, documented, and life cycle managed object library. The Device Library provides pre-configured status and diagnostic faceplates and AOI sets for Rockwell Automation® safety instructions and devices. The Safety Device Objects may be used with Machine Builder, Process, and Packaged Libraries or as standalone components. Safety Device Library add-on instructions objects collect, process, and deliver data between hardware devices and application logic.

Safety Device Instructions

The Safety Device Library includes instructions to interface with safety applications utilizing safety I/O. The instructions included are as follows:

- Dual Channel Input Stop (DCS)
- Dual Channel Input Stop with Test and Lock (DCSTL)
- Dual Channel Input Stop with Test and Mute (DCSTM)
- Configurable Redundant Output (CROUT)
- Two Sensor Asymmetrical Muting (TSAM)
- Two Sensor Symmetrical Muting (TSSM)
- Four Sensor Bi-directional Muting (FSBM)
- Two Hand Run Station Enhanced (THRSe)
- Safety Mat (SMAT)
- Safe Stop 2 (SS2)
- Safe Stop 1 (SS1)
- Safety Feedback Interface (SFX)
- Safe Torque OFF (STO)
- Safety Group Member
- Safety Group Monitor

There are two types of instructions:

- Device (Dvc): instruction used for devices. The device can be logical (e.g. DCS) or physical (e.g. Safety I/O or Sensor).
- Operation (Opr): instructions used for operation or applied code such as sequencing, operational execution, and general application functions. (e.g. Safety Group Monitor)

Safety Device Instructions

Instruction	Version	Instruction Description	Common Safety Applications
raC_Dvc_DCS	1.02	Dual Channel Input Stop	Emergency Stop, Safety Gate, Light Curtain, Area Scanner, Safety Mat, Cable (Rope) Pull Switch
raC_Dvc_DCSTL	1.02	Dual Channel Input Stop with Test and Lock	Safety Gate, Slide Lock
raC_Dvc_DCSTM	1.02	Dual Channel Input Stop with Test and Mute	Light Curtain, Area Scanner, Safety Mat

Safety Device Instructions

Instruction	Version	Instruction Description	Common Safety Applications
raC_Dvc_CROUT	1.02	Configurable Redundant Output	Redundant outputs
raC_Dvc_TSAM	1.02	Two Sensor Asymmetrical Muting	Light Curtain
raC_Dvc_TSSM	1.02	Two Sensor Symmetrical Muting	Light Curtain
raC_Dvc_FSBM	1.02	Four Sensor Bi-directional Muting	Light Curtain
raC_Dvc_THRSe	1.02	Two Hand Run Station Enhanced	Two Hand Run Station
raC_Dvc_SMAT	1.02	Safety Mat	Safety Mat
raC_Dvc_SS2	1.04	Safe Stop 2	Safe Stop 2
raC_Dvc_SS1	1.04	Safe Stop 1	Safe Stop 1
raC_Dvc_SFX	1.04	Safety Feedback Interface	Safety Feedback Interface
raC_Dvc_STO	1.04	Safe Torque Off	Safe Torque Off
raC_Opr_SafetyGroupMonitor	1.02	Safety Group Monitor	Grouping safety devices into a common safety zone
raC_Opr_SafetyGroupMember	1.02	Safety Group Member	Group member of safety group

Library Folders and Files

When you extract the library from the downloaded .zip folder, you will find the following folder and file structure. Note that some items are generalized with *TYPE* (e.g. Dvc, Opr, Tec) and *OBJECT* (e.g. DCS, CROUT, etc). The major and minor versions are represented by X and Y respectively.

Level 1	Level 2	Level 3	File Type	Description
Application Example			**Folder**	**Application Example Files**
	(raC-1_04-VD) raC_Dvc_Safety.vpd		VPD	View Designer Example Project
	SafetyApplication_ACM.xlsx		XLSX	Application Code Manager Project
	SafetyApplication_ME.apa		APA	FT View ME Project Archive
	SafetyApplication.ACD		ACD	Logix Designer Example Project
	SafetyApplication_SE.apa		APA	FT View SE Project Archive
ApplicationCodeManagerLibraries			**Folder**	**Application Code Manager files**
	Attachments (.HZ1 and .txt files)		Folder	ACM Object Attachments
	(RA-LIB)_Device_Asset-Control_Safety_raC_Dvc_*OBJECT*_(X.Y).HSL4		HSL4	ACM Asset-Control Object
	(RA-LIB)_Device_Device_Safety_raC_LD_*OBJECT*_(X.Y).HSL4		HSL4	ACM Device Object
HMI - FactoryTalk View ME			**Folder**	**FactoryTalk® View ME files**
	Displays - gfx		Folder	FT View ME display files
		(raC-X_YY-ME) raC_*TYPE_OBJECT*-faceplate.gfx	GFX	Object Faceplate display
	Global Objects - ggfx		Folder	FT View ME Global Object files
		(raC-X-ME) Graphic Symbols - LIBRARY.ggfx	GGFX	Graphic Symbol/Launch Button global objects
		(raC-X-ME) Toolbox - LIBRARY.ggfx	GGFX	Toolbox global objects
HMI - FactoryTalk View SE			**Folder**	**FactoryTalk® View SE Files**
	Displays - gfx		Folder	FT View SE display files
		(raC-X_YY-SE) raC_*TYPE_OBJECT*-faceplate.gfx	GFX	Object Faceplate display
	Global Objects - ggfx		Folder	FT View SE Global Object files
		(raC-X-SE) Graphic Symbols - LIBRARY.ggfx	GGFX	Graphic Symbol/Launch Button global objects
		(raC-X-SE) Toolbox - LIBRARY.ggfx	GGFX	Toolbox global objects
HMI - ViewDesigner - vpd			**Folder**	**View Designer Files**
	(raC-1_04-VD) raC_*Dvc*_Safety.vpd		VPD	Object faceplate and graphic symbol/launch buttons

HMI FactoryTalk View Images - png			**Folder**	**FT View ME/SE image files**
	images.png		PNG	FTView ME/SE images
Reference Manuals			**Folder**	**Manuals**
	DEVICE-RM500C-EN-P.pdf		PDF	Reference manual
Studio 5000 Logix Designer Files - L5X			**Folder**	**Studio 5000® AOI and RUNG import files**
	raC_*TYPE_OBJECT*_X.YY_RUNG.L5X		L5X	Object rung import
	raC_*TYPE_OBJECT*_X.YY_ROUTINE.L5X		L5X	Object routine import
	raC_*TYPE_OBJECT*_X.YY_AOI.L5X		L5X	Object AOI import
	ReadMe.txt		TXT	Explanation of standard vs PlantPAx® files
Videos			**Folder**	**How-to and Operational Overview Videos**
	How_To_Import_and_Add_Safety_Device_and_Operation_Objects_in_ACM.mp4		MP4	How-to Video
	How_To_Import_and_Configure_Safety_Device_Objects_in_FTViewME.mp4		MP4	How-to Video
	How_To_Import_and_Configure_Safety_Device_Objects_in_LogixDesigner.mp4		MP4	How-to Video
	How_To_Configure_Safety_Device_Objects_in_ViewDesigner.mp4		MP4	How-to Video
	Operational_Overview_of_Safety_Application_Example_in_LogixDesigner		MP4	How-to Video
	Operational_Overview_of_Safety_Device_Object_Faceplate.mp4		MP4	Operational Overview video
ReadMe.txt			TXT	Explanation of setup.cmd
SetUp.cmd			CMD	Application Code Manager setup script to register library

See the files in the *Application Example* folder to see a functional application that uses all of the Safety Device Library instructions. These files are referenced in the Programming Examples for each instruction. The files include a Studio 5000 Logix Designer® controller file, a Studio 5000® Application Code Manager project back-up, and HMI projects for Studio 5000 View Designer®, FactoryTalk® View ME, and FactoryTalk® View SE.

Visualization Files

Each Add-On Instruction has associated visualization files that provide a common user interface. The Safety Device Library supports three HMI options each with their own files supplied:

- FactoryTalk® View ME (Machine Edition)
- FactoryTalk® View SE (Site Edition)
- Studio 5000 View Designer®

FactoryTalk® View Visualization Files

You must import these files in the following order:

- Images (.png files)
- Global Objects(.ggfx file type)
- HMI faceplates (.gfx file type)

File Type Abbreviations	FactoryTalk® View SE	FactoryTalk® View ME	Description
Images (.png)	All .png files in the *HMI FactoryTalk View Images - png* folder. **IMPORTANT:** FactoryTalk® View application renames PNG files when they are imported with a .bmp file extension, but the files retain a .png format.		Common icons that are used in the Global Objects and standard displays for all objects.
Global objects (.ggfx)	(raC-1-SE) Graphic Symbols - Safety Device.ggfx	(raC-1-ME) Graphic Symbols - Safety Device.ggfx	Graphic symbols or launch buttons used to open faceplate displays from other displays.
	(raC-1-SE) Toolbox - Safety Device.ggfx	(raC-1-ME) Toolbox - Safety Device.ggfx	Common objects used across multiple device faceplates.
Standard displays (.gfx)	(raC-1_XX-SE) precedes name of the display.	(raC-1_XX-ME) precedes name of the display.	e.g. (raC-1_01-SE) raC_Dvc_DCS-Faceplate.gfx

Global object files contain Graphic Symbols that are created once and referenced multiple times on multiple displays in an application. When changes are made to a global object, all instances in the application are automatically updated.

Global objects serve two purposes:

- Toolbox files contain common elements that are used to build faceplate displays.
- Graphic Symbols files contain device symbols or launch buttons that you can use to build your application displays. Select the symbol to open the corresponding faceplate display.

Standard display files, commonly called faceplates, provide a common user interface.

Studio 5000 View Designer® Visualization Files

A single View Designer project file is supplied which contains faceplates and launch buttons for the Safety Device Library. This file is found in the *HMI - ViewDesigner - vpd* folder. In the VPD file you will find the required screens inside of the *User-Defined Screens* folder.

Display Type	View Designer Screen	Description
Screen	Toolbox	Graphic symbols or launch buttons used to open faceplate/pop-up displays from other displays.
Pop-Up	raC_Dvc_ precedes name of the pop-up.	Faceplate display for specific device. e.g. raC_Dvc_DCS_FP

Basic Faceplate Attributes

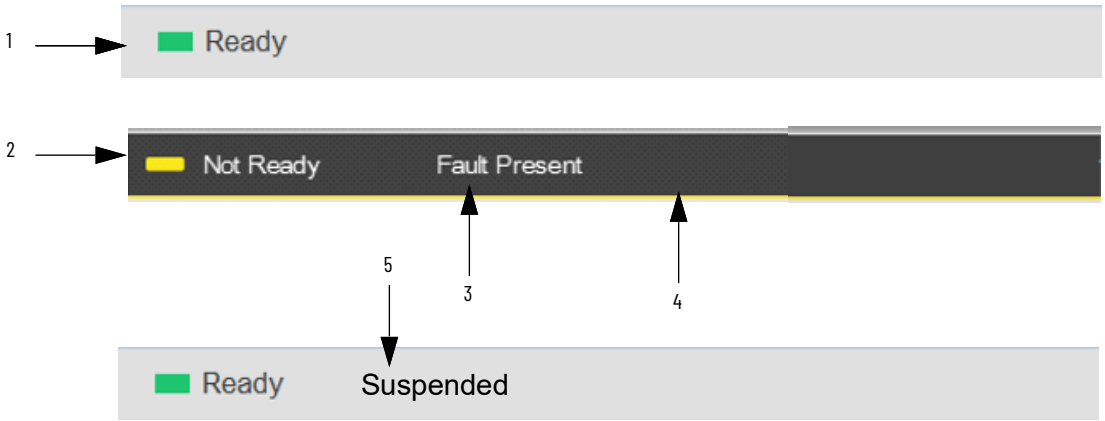
Faceplates consist of tabs, and each tab consists of one or more pages. The Home tab is displayed when the faceplate is initially opened. The faceplate provides the means for operators, maintenance personnel, engineers, and others to interact with a device or instruction instance, which includes a view of its status and values. Select the appropriate icon on the left of the faceplate to access a specific tab. This section provides an overview of the faceplate attributes that are common across the objects. More details are supplied in the individual section for each object.

Common Status Banner

At the top of all device object faceplates there is a common status banner which provides the following information:

- Ready (green LED icon) or Not Ready (yellow LED icon) status
- Faulted (banner will show Not Ready with fault message)
- Safety Demand (banner will show Not Ready with Safety Demand
- Input Suspended (banner will show green LED icon and Ready with Suspended

.



Item	Description
1	Ready state displays green LED icon and grey background.
2	Faulted state shows yellow LED icon and Not Ready status.
3	Not Ready Reason: “Fault Present” or “Safety Demand”
4	Faulted state shows yellow border around banner.
5	Input Suspended state displays green LED icon and “Suspended”.

Faceplate Navigation

All device object faceplates have navigation tabs on the left side of the faceplate. Navigation tabs may vary based on device type. The active tab will show as a light grey, while an inactive tab will show as a dark grey.


Active Tab


Inactive Tab

The common tabs are shown below. The Extensions Tab is used to navigate to an associated instruction faceplate such as the DCS instruction related to TSSM, TSAM, or FSBM faceplates.


Home Tab


Extensions Tab

Basic Launch Button Attributes

All Safety Device Library objects share the same HMI launch button.

A black 'X' icon is shown on the top left of the button when the device or group is in a Not Ready State due to a fault or diagnostic present. A black octogan icon is shown on the bottom left of the button when the device has a safety demand present.

Input device status	Launch button state
Ready State	
Not Ready State - Fault/Diagnostic Present	
Not Ready Safety Demand	

The button has a text label which is set to the instruction type (e.g. "DCS") + the instance's *.Inf_DisplayName* tag.

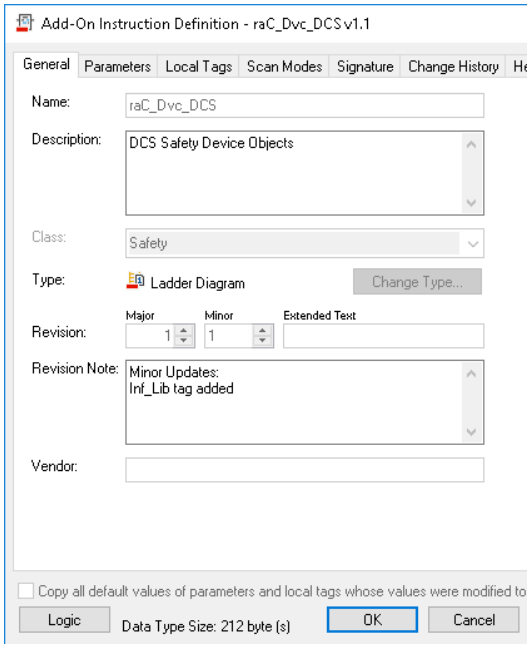
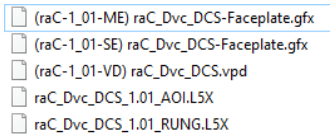
Scope: raC_Dvc_DCS Show: All Tags

Data Context: MainEstop (SafetyZone1)

Name	Usage	Value	Style	Data Type
Inp_DCS	InOut	{...}		DCI_STOP
Out_Ctrl_Inf	InOut	{...}		raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	InOut	{...}		raC_UDT_Itf_SafetyDevice_Sts
EnableIn	Input	1	Decimal	BOOL
Inp_Suspend	Input	0	Decimal	BOOL
_bSts	Local	0	Decimal	DINT
_DCS	Local	{...}		DCI_STOP
_EF	Local	0	Decimal	BOOL
_ET	Local	1	Decimal	BOOL
_OCS_Initialized	Local	1	Decimal	BOOL
_ONS	Local	0	Decimal	DINT
Inf_DisplayName	Local	'Gate Switch'		STR0024
Inf_Lib	Local	'raC-1_01'		STR0016
Inf_Type	Local	'raC_Dvc_DCS'		STR0032
Wrk_Name	Local	''		STR0024

Library Versions

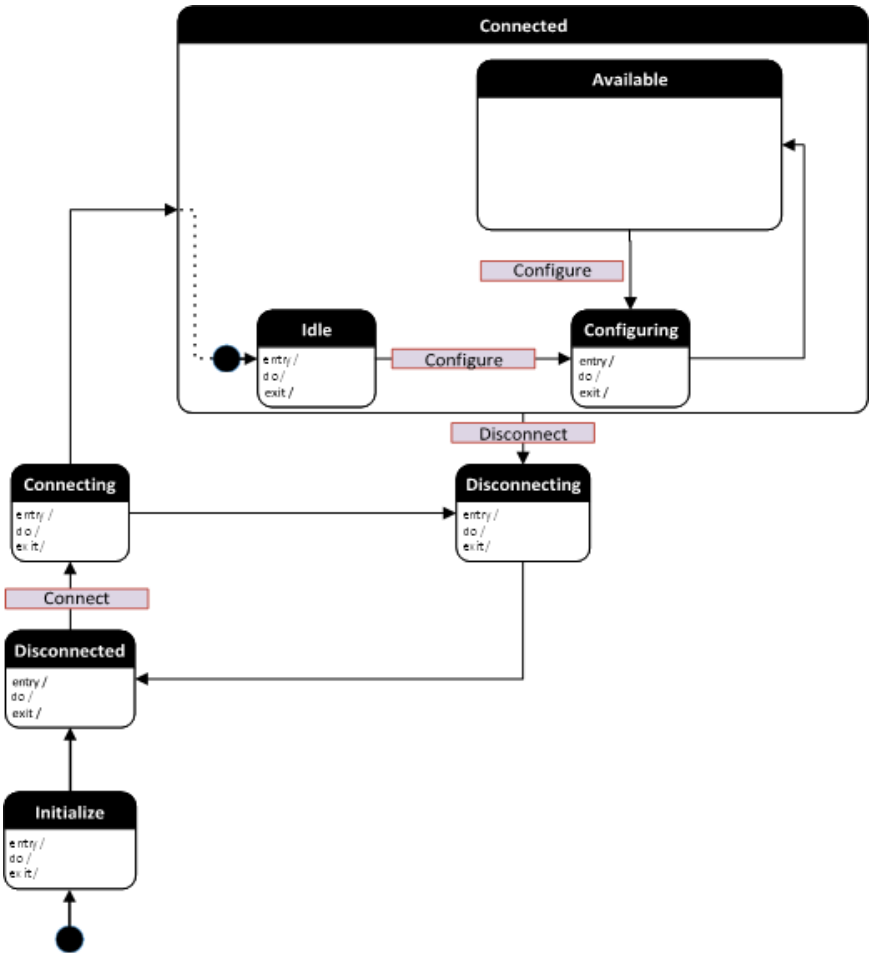
Each library object has a revision x.yy where: x is the Major Revision number and yy is the Minor Revision number. Each release of the library comes with release notes that describe the changes that were made since the last release. You can find the revision number of the object in a number of locations as shown in the following table.

Component	Example
The Add-On Instruction in Logix Designer application has revision information visible when the instruction is selected in the Controller Organizer.	
The Add-On Instruction Definition General tab shows the revision number along with basic revision notes. Refer to the release notes for complete revision notes.	
The revision number is shown in the file names for GFX, VPD, ACM.HSL4, AOI.L5X, and RUNG.L5X files.	

State Model

The following section will discuss the state model for Device Object. The figure below shows the core logic states.

By default, each state is active for a minimum of 256us to allow for evaluation of state outside of the ADO instance in the user program



Interfaces

Device object interfaces are intended to provide the application programmer a class based harmonized interface for interacting with the device object from user code. Standard control interfaces are used for passing Information (Inf), Settings (Set), Commands (Cmd) and Status (Sts). At this time the Safety Device Library only uses Sts and Inf type interfaces.

The following Safety Common Control Interface tags are the primary device program tags to read and write to when interfacing to safety devices. The value of using these tags in your specific application code is that you may use a number of different safety devices such as DCS, CROUT, etc without having to update your application device interface tags. The primary function of these interfaces within the Safety Device Library is to pass individual instruction information to the Safety Group Monitor instruction. The following interfaces apply to all Safety Device Library instructions.

Interface Class	Object Class	Interface Type	Interface Name (UDT)
Control	Safety	Information	raC_UDT_ItfAD_SafetyDevice_Inf
		Status	raC_UDT_Itf_SafetyDevice_Sts

raC_UDT_Itf_SafetyDevice_Sts

This is the Safety Device Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data. The table below shows member names, descriptions and tag data types. This information is passed from the safety instruction to the Safety Group Monitor.

Input	Description	Data Type
bSts	Bit overlay (Visible) covering all subsequent Boolean members. (Unused)	DINT
DeviceSuspended	Device suspended status (Inp_Suspend = 1)	BOOL
SafetyDemand	Safety demand status ON (1): Safety demand is present on the device object OFF (0): No safety demand	BOOL
ResetRequired	Reset Required status ON (1): Device requires reset signal OFF (0): No reset required	BOOL
DiagnosticPresent	Diagnostic status ON (1): A diagnostic is present in the device object OFF (0): Device is operating normally	BOOL
FaultPresent	Fault Status ON (1): A fault is present in the device object OFF (0): Device is operating normally	BOOL
Muted	Device Muted status (Available for DCSTM)	BOOL

raC_UDT_Itf_SafetyDevice_Inf

This is the Safety Device Common Control Interface User-Defined Data Type for device information. The table below shows member names, descriptions and tag data types. This information is passed from the safety instruction to the Safety Group Monitor.

Input	Description	Data Type
Type	Provides device object type information e.g., raC_Dvc_DCS, raC_Dvc_DCSTL, raC_Dvc_DCSTM, raC_Dvc_CROUT, raC_Dvc_TSAM, raC_Dvc_TSSM, raC_Dvc_FSBM, raC_Dvc_THRSe, raC_Dvc_SMAT	STR0032
DisplayName	Provides object display name	STR0024
eClass	Provide instruction type information eClass ID: Instruction Type 0: DCS 1: DCSTL 2: DCSTM 3: CROUT 5: SafetyGroupMonitor 6: TSAM 7: TSSM 8: FSBM 9: THRSe 10: SMAT 11:SS2 12: SS1 13:SFX 14:STO	DINT

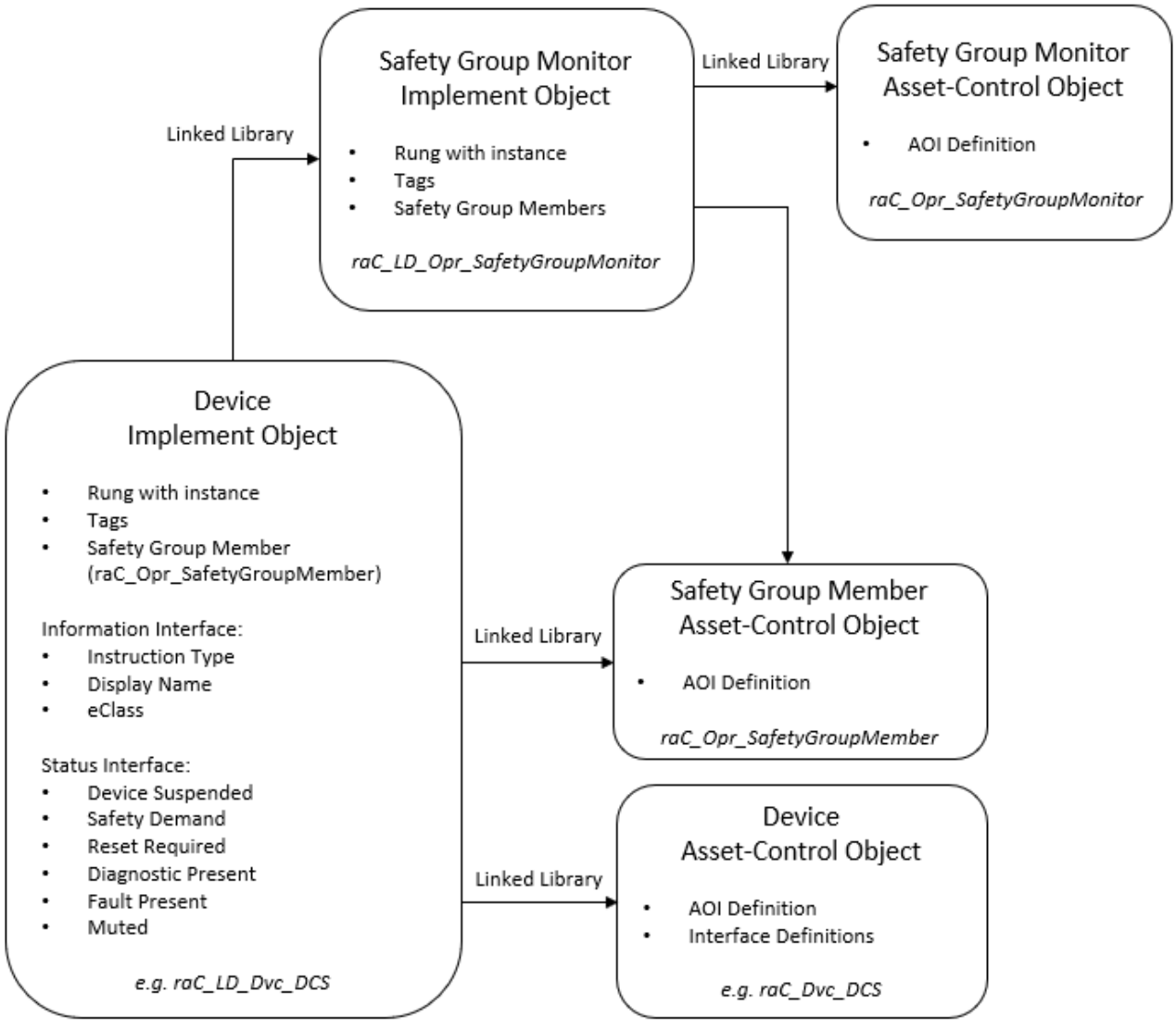
Application Code Manager

Architectural Overview

Device libraries, as with most Application Code Libraries are divided into 2 logical groups: either Asset-Control Object or Device Implement Object.

Asset-Control Objects contain the asset definition of an object and any associated content which belongs to the asset. This includes controller tags, add-on instructions, data types, and attachments such as HMI content and documentation. These are found under the *(RA-LIB) Device > Asset-Control* folder and have names like *raC_Dvc_xxxx* where *xxxx* is the device name.

Device Implement Objects contain an instance of an asset-control object and provide all related configuration of the asset. The Device implement type is the application code (e.g. programming rung). This includes the required controller tags, programs, modules, and FactoryTalk® View ME/SE symbols. These are found under the *(RA-LIB) Device > Device* folder and have names like *raC_LD_Dvc_xxxx* where *xxxx* is the device name. LD stands for ladder logic.


Safety Group Monitor
Implement Object
• Rung with instance
• Tags
• Safety Group Members
raC_LD_Opr_SafetyGroupMonitor
Linked Library
Safety Group Monitor
Asset-Control Object
• AOI Definition
raC_Opr_SafetyGroupMonitor
Linked Library
Device
Implement Object
• Rung with instance
• Tags
• Safety Group Member
(raC_Opr_SafetyGroupMember)
Information Interface:
• Instruction Type
• Display Name
• eClass
Status Interface:
• Device Suspended
• Safety Demand
• Reset Required
• Diagnostic Present
• Fault Present
• Muted
e.g. raC_LD_Dvc_DCS
Linked Library
Safety Group Member
Asset-Control Object
• AOI Definition
raC_Opr_SafetyGroupMember
Linked Library
Device
Asset-Control Object
• AOI Definition
• Interface Definitions
e.g. raC_Dvc_DCS

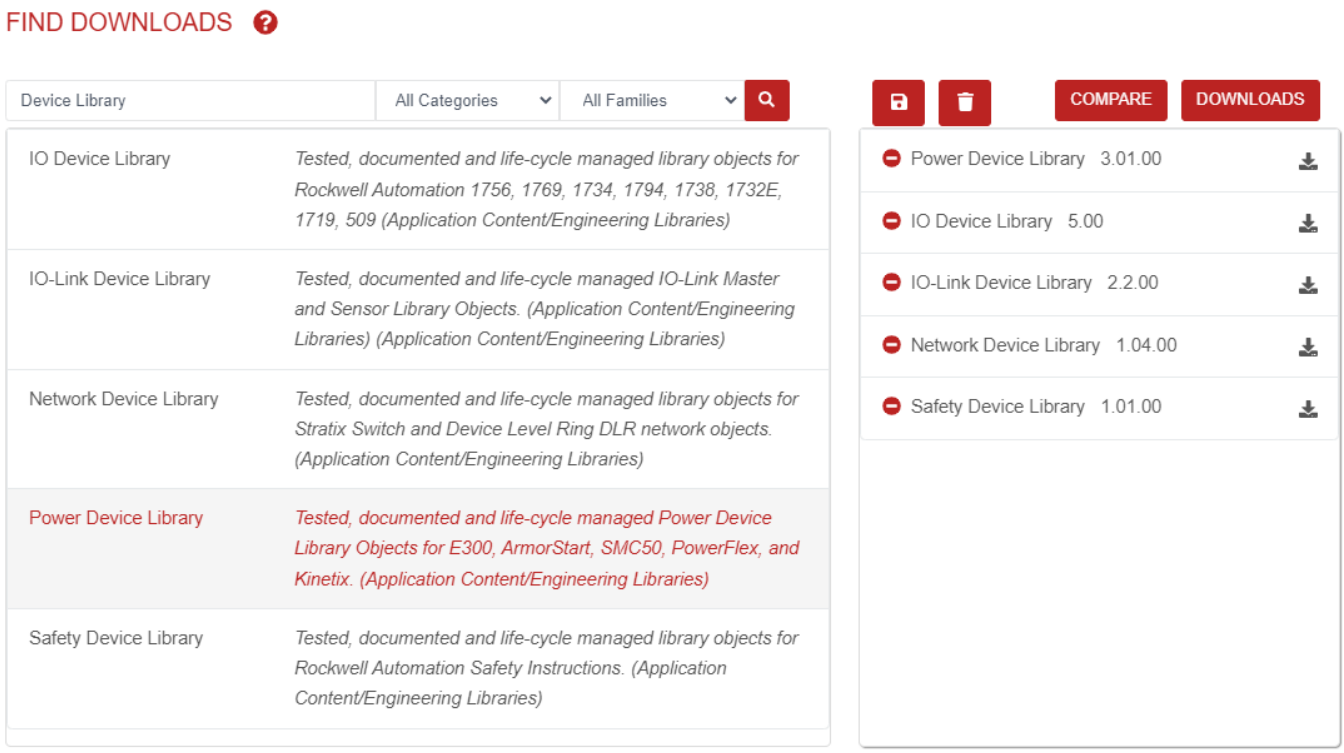
Using the Library

Install the Library

Download the Library

For the latest compatible software information and to download the Rockwell Automation® Library, see the Product Compatibility and Download Center.

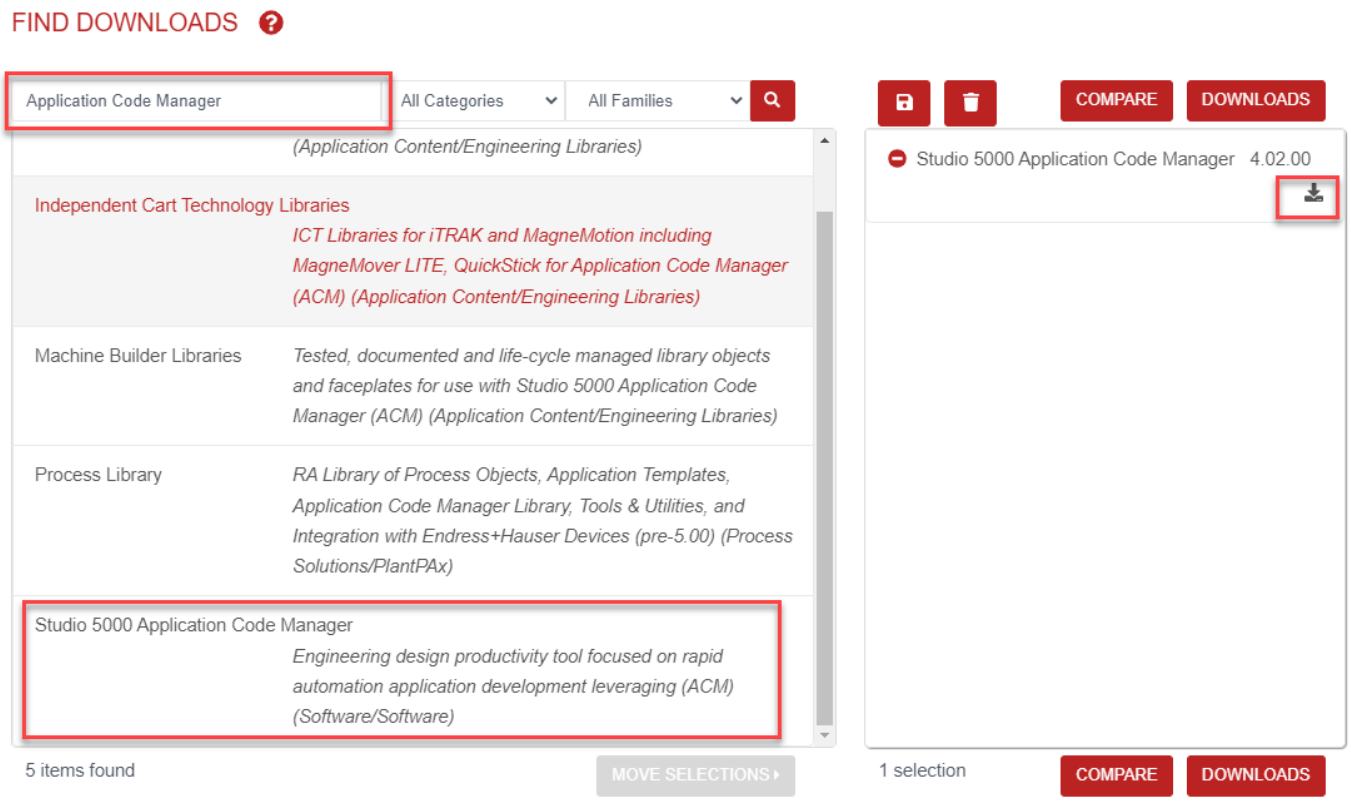
Search "Device Library" or filter on Application Content to quickly find the library.



Download & Install Studio 5000® Application Code Manager

Studio 5000® Application Code Manager is free to install from Rockwell Automation®'s Product Compatibility and Download Center.

Search "Application Code Manager" and select the item to download.



Extract the downloaded .zip file by running the *4.xx.00-Studio5000_ACM-DVD.exe* executable file. This will extract a new folder containing a *Setup.exe* file which can be run to begin product installation.

Follow the prompts from the splash screen until installation is complete. Note that a SQL server is required for Application Code Manager. SQL Server Express is offered for free and is included in the Application Code Manager installer.

Register Libraries in Studio 5000® Application Code Manager

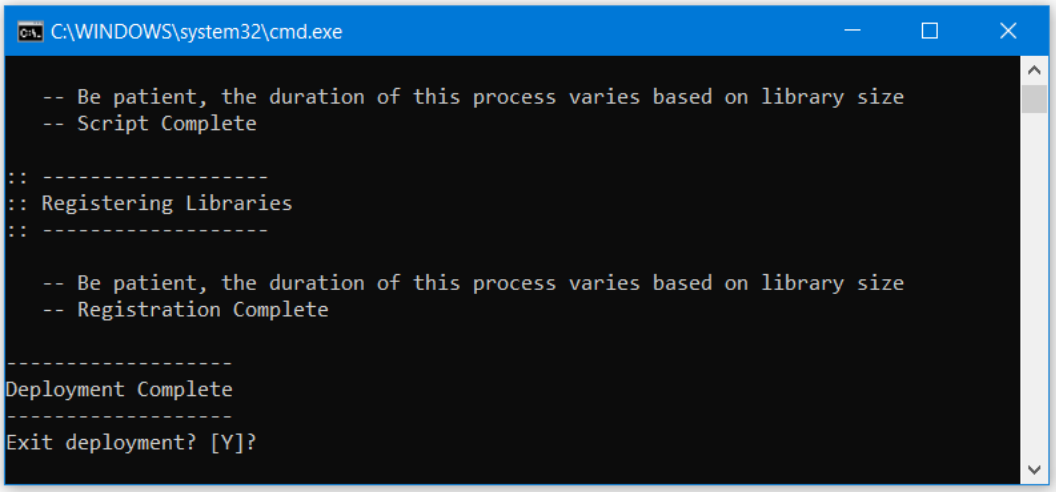
It is recommended that you use Studio 5000® Application Code Manager or the Studio 5000® "Import Library Objects" Plug-In Wizard to import device library objects into a Logix 5000 controller project. To use the library in Application Code Manager you must first register the libraries.

Using Studio 5000® Application Code Manager is not mandatory although it is highly recommended to reduce the likelihood of configuration errors and simplify the workflow. Alternatively, you can import the RUNG.L5X or ROUTINE.L5X files directly into a Studio 5000 project.

The *Lite* version of Studio 5000® Application Code Manager is free of charge and can be downloaded from the Product Compatibility and Download Centre. None of the features included in the Standard (paid) version are required to use Device Object Libraries.

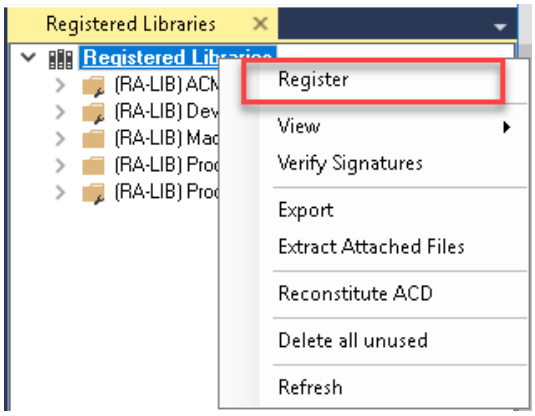
Register Complete Library Automatically

To automatically register the entire library, find and run the *setup.cmd* file in the root folder of the library files. You will see a windows console appear as the script runs. When it is complete it will display “Deployment Complete”. Enter “Y” to exist the console.



Register Individual Library Objects Manually

As an alternative to registering the entire library using the setup.cmd script, you can manually register one or multiple library objects in Studio 5000® Application Code Manager. Open up Application Code Manager and view the Registered Libraries panel on the right. Right-click on *Registered Libraries* and select *Register*. Browse to the *ApplicationCodeManagerLibraries* folder within the library files and select any HSL4 files that you would like to register. Note you may select more than one at a time. Once you complete registering the desired objects they will be shown under the *(RA-LIB) Device* solution folder.



Importing Logic into Studio 5000 Projects

There are multiple methods to using the logic in a Studio 5000® Application. For projects that are being developed from scratch using Application Code Manager along with other Application Code Libraries such as the PlantPAx® Process Objects Library or the Machine Builder Library, you can continue to use the Device Object Libraries in Application Code Manager. For existing applications where devices are being added, it is recommended to use the Studio 5000® Plug-In "Import Library Objects" Wizard. Alternatively you can import the RUNG.L5X or ROUTINE.L5X files into your program and configure them manually.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following video for this section: "How_To_Import_and_Configure_Safety_Device_Objects_in_LogixDesigner"

It is not recommended to simply import the AOI.L5X files and attempt to build your own logic rung. Doing so will increase the likelihood of configuration errors and likely miss logic that is required outside of the Add-On Instruction.

AOI files should only be imported when updating an existing application from a previous version of a Device Object Library to a newer one.

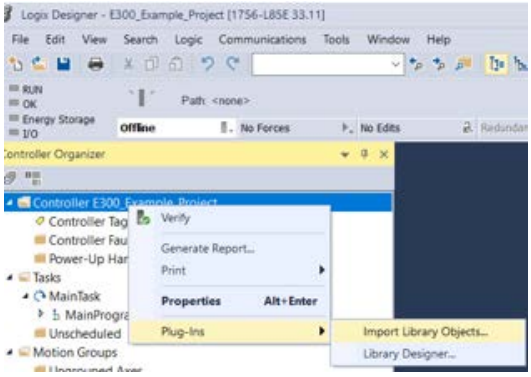
Below is a table to capture recommendations on when to use which tool or workflow when importing and configuring device objects.

Tool/Workflow	Description of when to use	Software Requirements
Application Code Manager (full application)	Project is developped from scratch using Application Code Manager along with PlantPAx® or Machine Builder libraries.	Studio 5000 Logix Designer® Studio 5000® Application Code Manager (Lite)
Studio 5000 Plug-In "Import Library Objects" Wizard	Application Code Manager is installed but not required for the entire project. Application has already been developed but some Device Objects need to be added.	Studio 5000 Logix Designer® Studio 5000® Application Code Manager (Lite)
Import ROUTINE.L5X or RUNG.L5X File	Application Code Manager is not installed. Application has already been developed but some Device Objects need to be added. Familiar with rung import workflow.	Studio 5000 Logix Designer®
Import AOI.L5X File	Updating existing application that contains an older version of a Device Object AOI.	Studio 5000 Logix Designer®

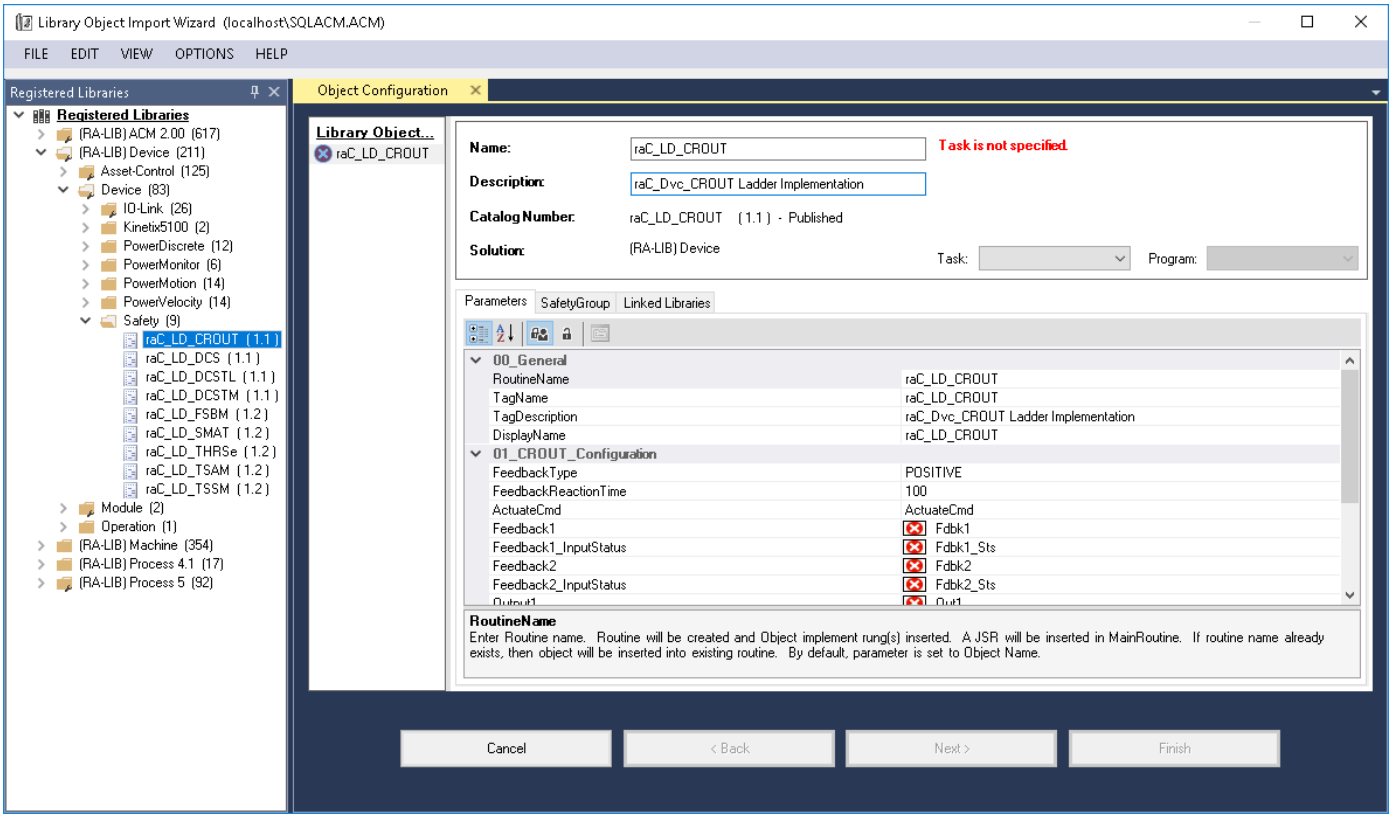
Import Library Objects Wizard

The most simple way to import a Device Object into an existing application is to use the Studio 5000® Plug-In "Import Library Objects" wizard. This plug-in requires Application Code Manager to be installed but does not require it to be open or have a project created.

Right click on an item (e.g. Controller, Task, Program, etc) in the Controller Organizer and select *Plug-Ins > Import Library Objects...*



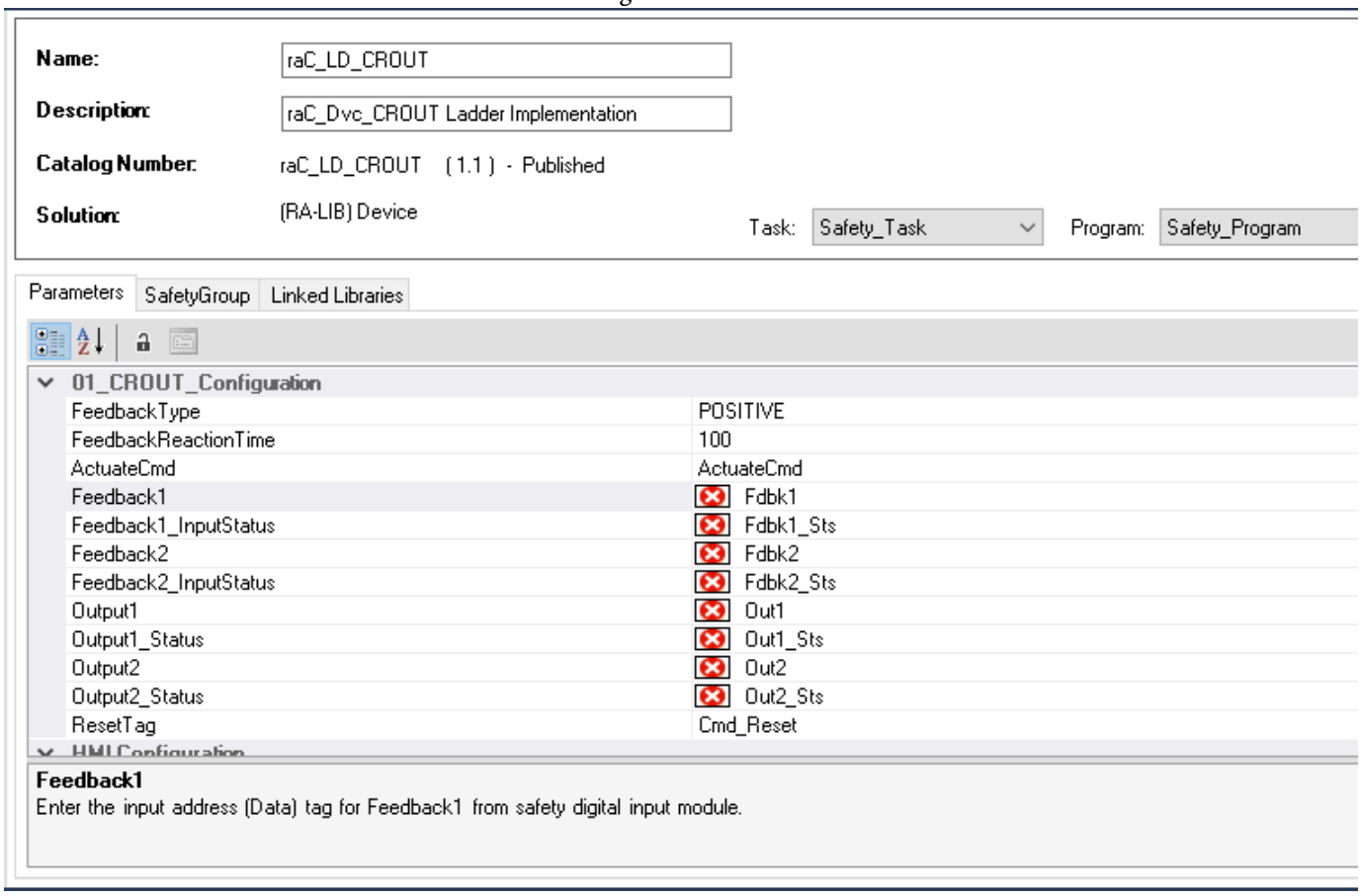
This will launch a small wizard version of Application Code Manager inside of your Studio 5000 Logix Designer® Project. In the Registered Libraries panel on the left, find your desired object under *Registered Libraries* > *(RA-LIB) Device* > *Device* and drag it into the Library Object(s) list in the Object Configuration Tab.



Perform the following configuration:

- Enter a **name** and **description**. Maximum name length can be 22 characters. Note that other parameters such as the RoutineName, TagName, etc will auto-complete based on these fields.
- Assign the **Task** and **Program**.
- Assign a **DisplayName** which is used as the faceplate title and object tag description.

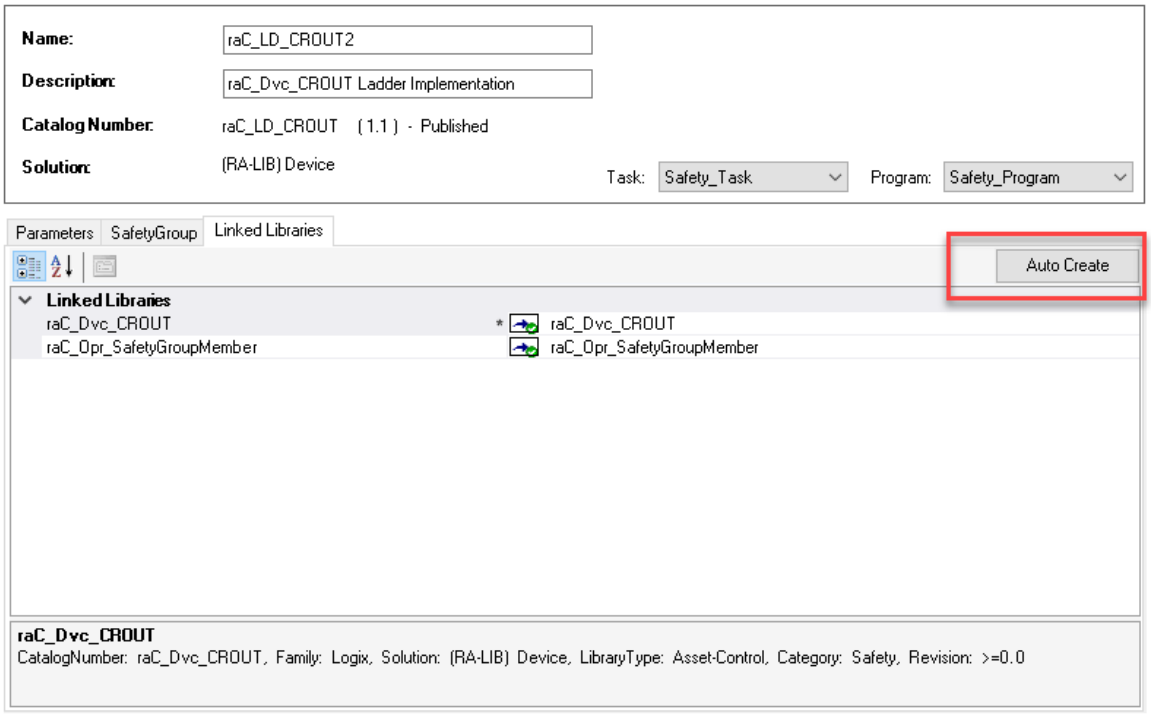
- Configure the instruction. Each safety device instruction has unique configuration which may include settings related to feedback, reaction times, command tags, and IO tags. IO tags by default will be set with placeholders (e.g. Out1, Fdbk2, etc). These tags may be left as-is and updated to physical IO tags later, or modified at this time. Note that when using the plug-in you are unable to browse for IO tags directly from the wizard.
- The HMI Configuration options are not used in the Plug-In Wizard and can be ignored.

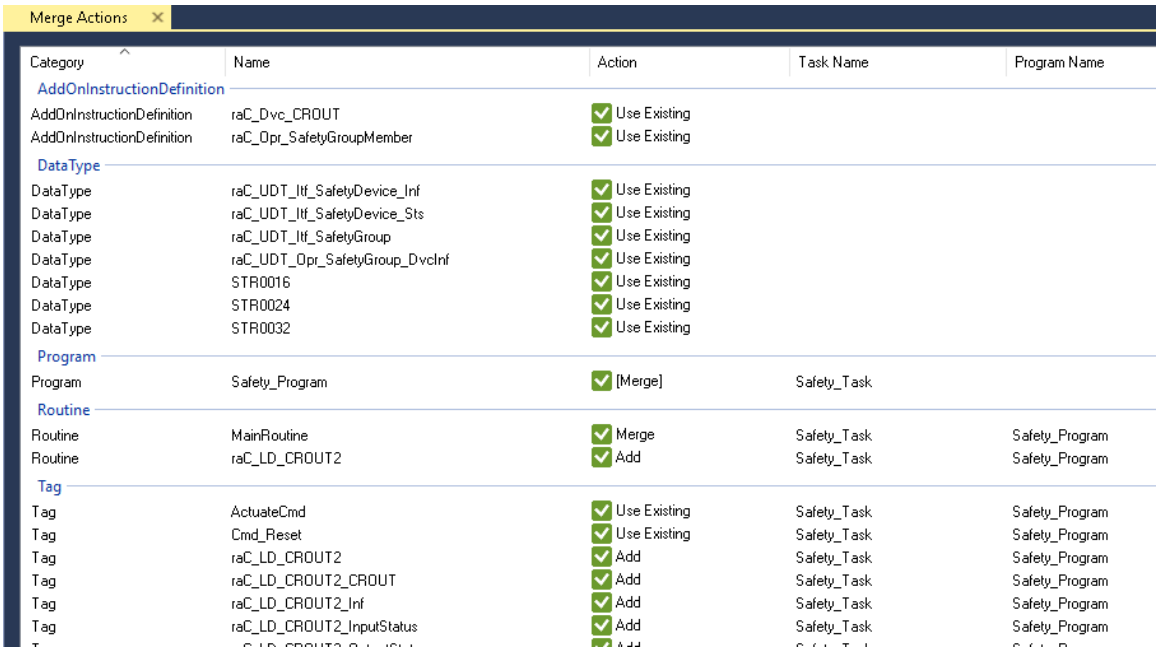


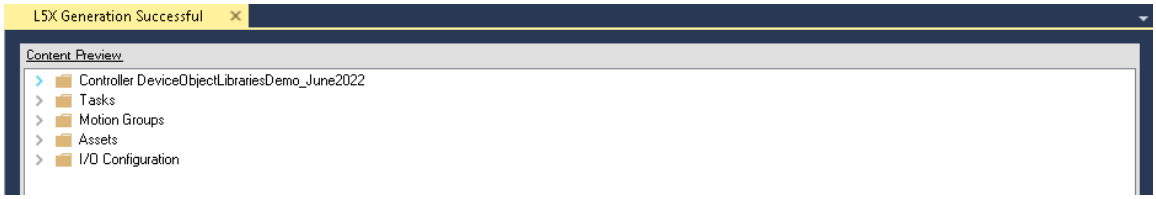
- The SafetyGroup tab is not applicable when using the plug-in wizard and can be skipped.
- Click next or click on the *Linked Libraries* tab. Click the *Auto Create* button to automatically create all of the required linked libraries.

You can manually create new linked libraries or point to existing linked libraries if necessary. You may need to do this if you would like to use an older version of library objects when multiple versions are installed in Application Code Manager.

- On the following screen you can select the desired Merge Actions. Generally these can be left with the default actions.
  - Add: used when AOIs don't previously exist in application
  - Overwrite: usually preferred. Used when AOIs previously exist but may or may not be the same revision.
  - Use Existing: used when AOIs previously exist in the application and you do not wish to overwrite the existing items.
- Click next and you can now see any new logic and modules that will be created.
- Click Finish to complete the import.


Name:
raC_LD_CROUT2
Description:
raC_Dvc_CROUT Ladder Implementation
Catalog Number:
raC_LD_CROUT (1.1) - Published
Solution:
(RA-LIB) Device
Task:
Safety_Task
Program:
Safety_Program
Parameters
SafetyGroup
Linked Libraries
Auto Create
Linked Libraries
raC_Dvc_CROUT
raC_Dvc_CROUT
raC_Opr_SafetyGroupMember
raC_Opr_SafetyGroupMember
raC_Dvc_CROUT
CatalogNumber: raC_Dvc_CROUT, Family: Logix, Solution: (RA-LIB) Device, LibraryType: Asset-Control, Category: Safety, Revision: >=0.0


Merge Actions
Category
Name
Action
Task Name
Program Name
AddOnInstructionDefinition
AddOnInstructionDefinition raC_Dvc_CROUT Use Existing
AddOnInstructionDefinition raC_Opr_SafetyGroupMember Use Existing
DataType
DataType raC_UDT_Itf_SafetyDevice_Inf Use Existing
DataType raC_UDT_Itf_SafetyDevice_Sts Use Existing
DataType raC_UDT_Itf_SafetyGroup Use Existing
DataType raC_UDT_Opr_SafetyGroup_DvcInf Use Existing
DataType STR0016 Use Existing
DataType STR0024 Use Existing
DataType STR0032 Use Existing
Program
Program Safety_Program [Merge] Safety_Task
Routine
Routine MainRoutine Merge Safety_Task Safety_Program
Routine raC_LD_CROUT2 Add Safety_Task Safety_Program
Tag
Tag ActuateCmd Use Existing Safety_Task Safety_Program
Tag Cmd_Reset Use Existing Safety_Task Safety_Program
Tag raC_LD_CROUT2 Add Safety_Task Safety_Program
Tag raC_LD_CROUT2_CROUT Add Safety_Task Safety_Program
Tag raC_LD_CROUT2_Inf Add Safety_Task Safety_Program
Tag raC_LD_CROUT2_InputStatus Add Safety_Task Safety_Program


L5X Generation Successful
Content Preview
Controller DeviceObjectLibrariesDemo_June2022
Tasks
Motion Groups
Assets
I/O Configuration

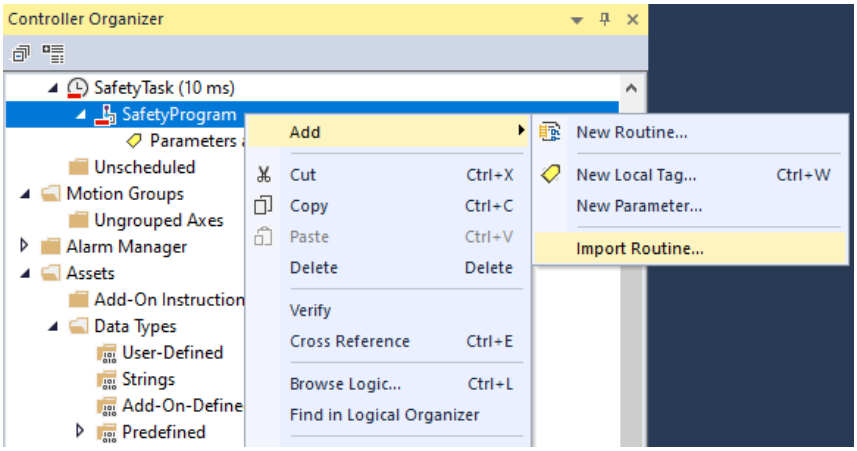
Import Routine Logic

An Add-On Instruction is defined once in each controller project, and can be instantiated multiple times in your application code. To use pre-engineered logic, import each desired ROUTINE.L5X file into a controller project.

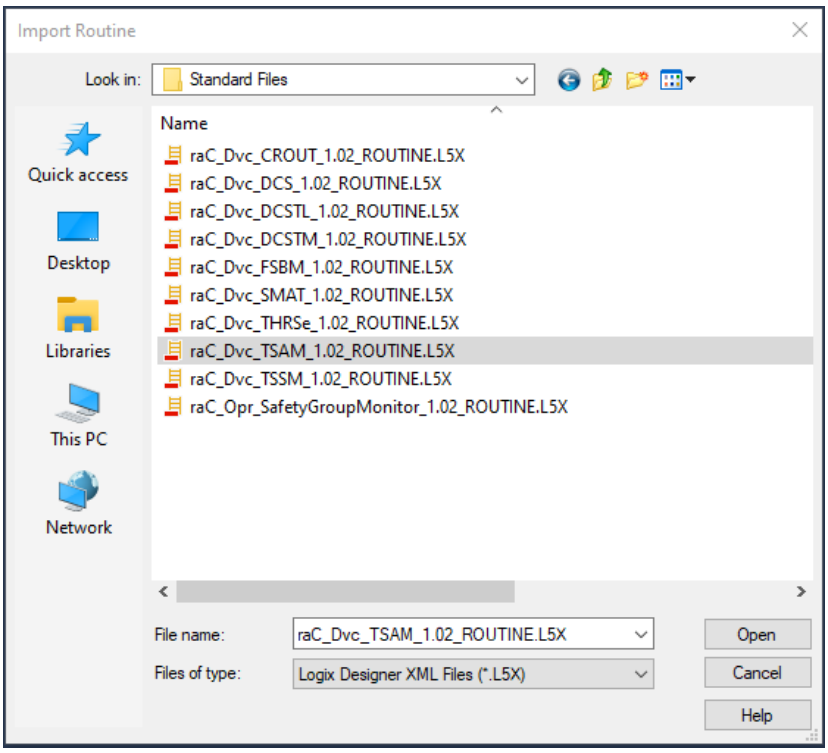
1. In the Studio 5000 Logix Designer® application, open a new or existing project.

IMPORTANT Add-On Instruction definitions can be imported, but not updated, online.

2. Right click on the Program in the Controller Organizer and choose Add then Import Routine...



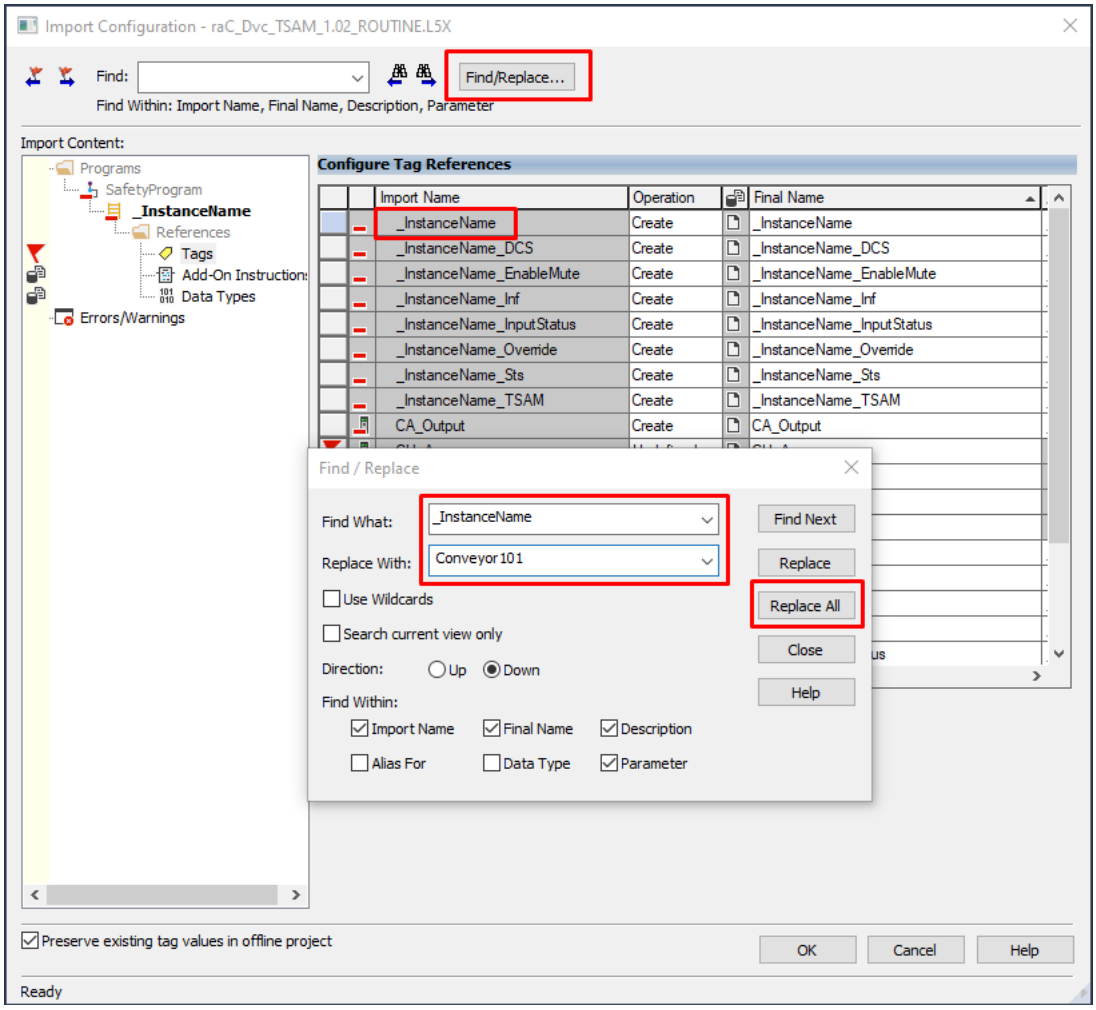
3. Select the desired ROUTINE and Select Open. The file will have a name like *raC_Dvc_TSAM_1.02_ROUTINE.L5X*



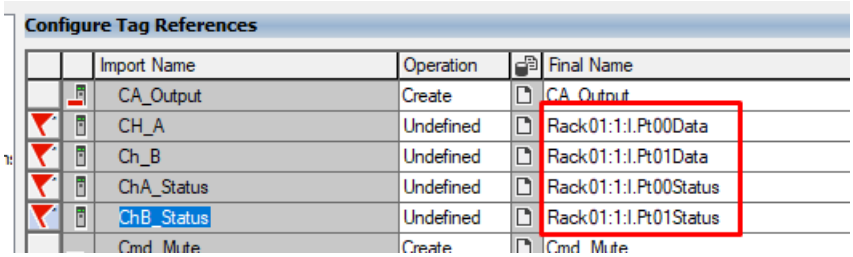
Some Add-On Instructions are provided in ROUTINE import files.

Both "ROUTINE" and "AOI".L5X files are provided. Import the ROUTINE into a Program to get all required additional tags, data types, and message configurations.

4. An *Import Configuration* dialogue window will open and display generic import names which include "_InstanceName". Click the *Find Replace...*button and replace all instances of "_InstanceName" with your desired device name (e.g. "Conveyor101").

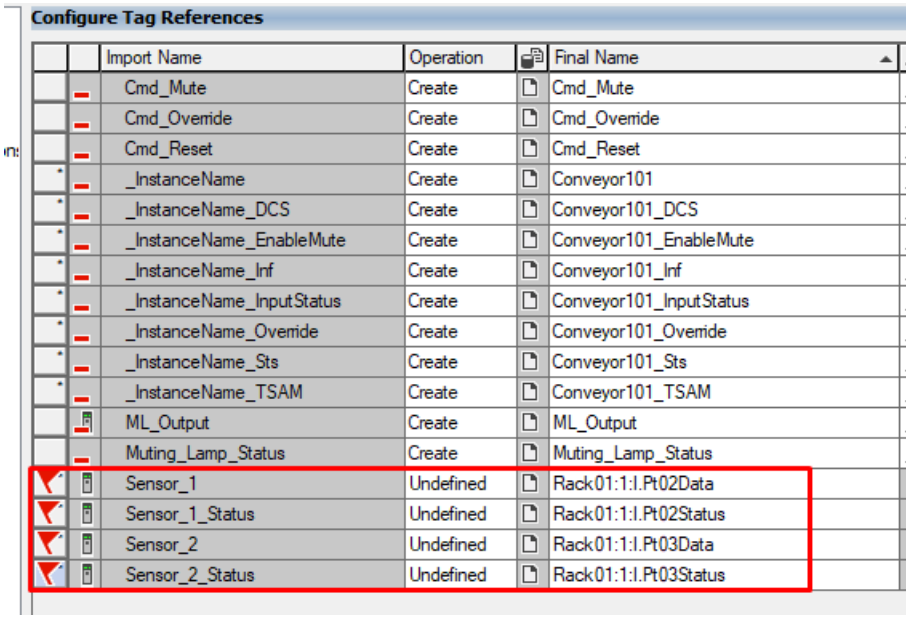


5. You will need to assign the Safety IO Card tag addresses for CH_A, CH_B, CHA_Status & CHB_Status. You can type in or browse for the correct input (:I). tags in your project. In this example our input card is called Rack01_01 (1734-IB8S)



6. Assign the Safety IO Card tag addresses for Sensor_1, Sensor_2, Sensor_1_Status & Sensor_2_Status. You can type in or browse for the

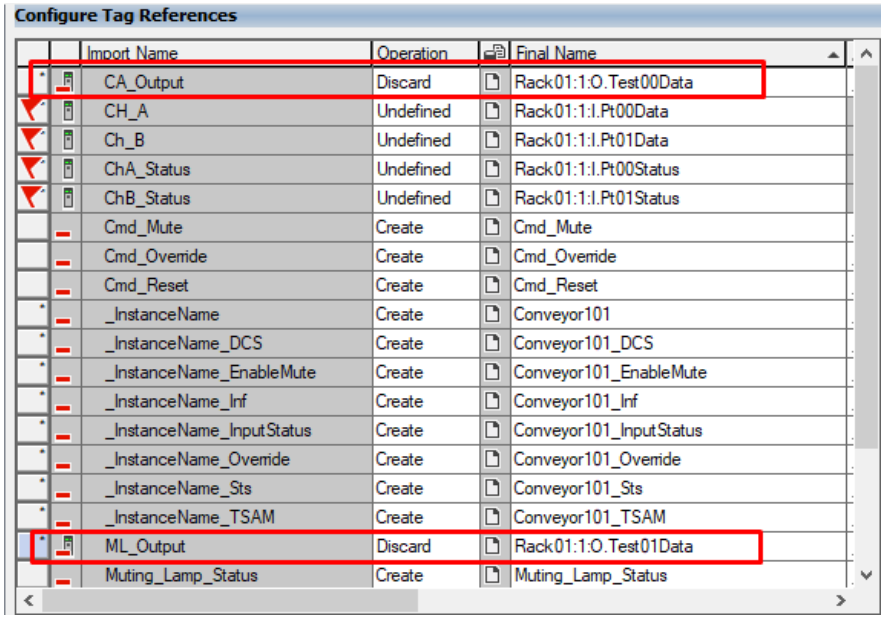
correct input (:I). tags in your project. In this example our input card is called Rack01_01 (1734-IB8S).



Configure Tag References

Import Name	Operation	Final Name
Cmd_Mute	Create	Cmd_Mute
Cmd_Override	Create	Cmd_Override
Cmd_Reset	Create	Cmd_Reset
_InstanceName	Create	Conveyor101
_InstanceName_DCS	Create	Conveyor101_DCS
_InstanceName_EnableMute	Create	Conveyor101_EnableMute
_InstanceName_Inf	Create	Conveyor101_Inf
_InstanceName_InputStatus	Create	Conveyor101_InputStatus
_InstanceName_Override	Create	Conveyor101_Override
_InstanceName_Sts	Create	Conveyor101_Sts
_InstanceName_TSAM	Create	Conveyor101_TSAM
ML_Output	Create	ML_Output
Muting_Lamp_Status	Create	Muting_Lamp_Status
Sensor_1	Undefined	Rack01:1:I.Pt02Data
Sensor_1_Status	Undefined	Rack01:1:I.Pt02Status
Sensor_2	Undefined	Rack01:1:I.Pt03Data
Sensor_2_Status	Undefined	Rack01:1:I.Pt03Status

7. Assign the Safety IO Card tag addresses for ML_Output & CA_Output. You can type in or browse for the correct Test Output (:O). tags in your project. In this example our input card is called Rack01_01 (1734-IB8S).



Configure Tag References

Import Name	Operation	Final Name
CA_Output	Discard	Rack01:1:O.Test00Data
CH_A	Undefined	Rack01:1:I.Pt00Data
Ch_B	Undefined	Rack01:1:I.Pt01Data
ChA_Status	Undefined	Rack01:1:I.Pt00Status
ChB_Status	Undefined	Rack01:1:I.Pt01Status
Cmd_Mute	Create	Cmd_Mute
Cmd_Override	Create	Cmd_Override
Cmd_Reset	Create	Cmd_Reset
_InstanceName	Create	Conveyor101
_InstanceName_DCS	Create	Conveyor101_DCS
_InstanceName_EnableMute	Create	Conveyor101_EnableMute
_InstanceName_Inf	Create	Conveyor101_Inf
_InstanceName_InputStatus	Create	Conveyor101_InputStatus
_InstanceName_Override	Create	Conveyor101_Override
_InstanceName_Sts	Create	Conveyor101_Sts
_InstanceName_TSAM	Create	Conveyor101_TSAM
ML_Output	Discard	Rack01:1:O.Test01Data
Muting_Lamp_Status	Create	Muting_Lamp_Status

8. The ROUTINE will now be imported into your Program

Using Studio 5000 View Designer®

Using View Designer Project Files

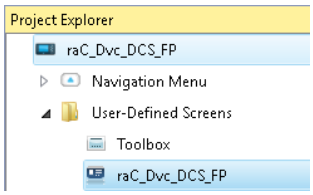
Studio 5000 View Designer® may be used for HMI development for PanelView™ 5000 applications. Open up your Studio 5000 View Designer®

project alongside a second application instance running the required VPD file in the library folder *HMI - ViewDesigner - vpd*.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following video for this section: "How_To_Configure_Safety_Device_Objects_in_ViewDesigner"

You will notice there are two screens available under the *User-Defined Screens* folder:

- Toolbox: This has the graphic symbol launch buttons for the faceplate.
- raC_Dvc_xxxxx_FP: This is a faceplate pop-up screen.



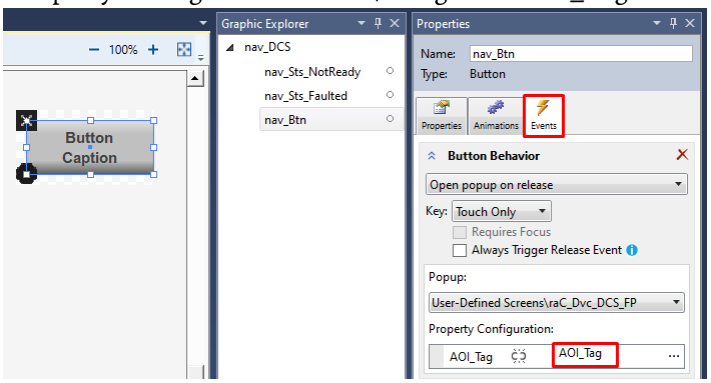
To include these files in your project, perform the following steps:

- Copy the entire faceplate _FP screen from the supplied VPD project to your project application.
- Open the Toolbox screen and copy the desired graphic symbol and paste it into a screen in your project application.

Configuring View Designer Objects

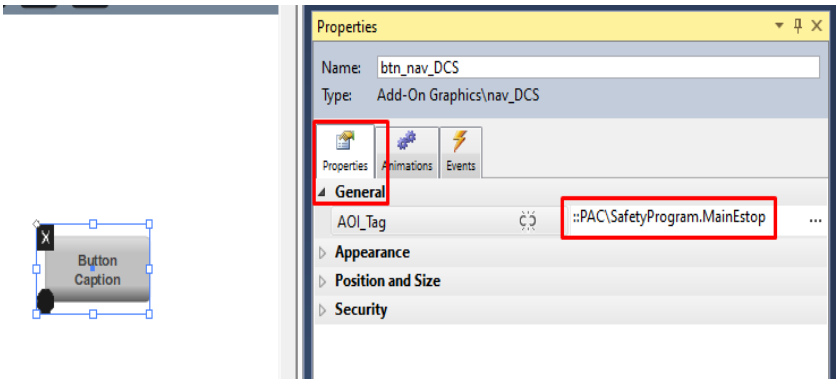
To link the launch button to the faceplate, first right click on the button and select Open Graphic Definition. Click on nav_Btn. In the *Events* tab within the *Properties* window. Assign the following settings:

- Popup: Browse the identical Device object screen under the User-Defined Screens.
- Property Configuration: Write/Assign text AOI_Tag



Next, Come back to the Screen where you pasted the launch button. In the *Properties* tab within the *Properties* window. Assign the following settings:

AOI_Tag: Browse the AOI_Tag from controller and assign to it.



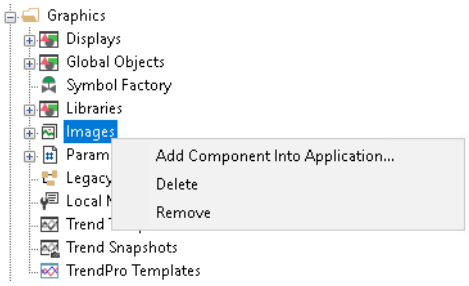
Using FactoryTalk® View Studio

Import FactoryTalk® View Visualization Files

There are several components to import for the visualization files. You import files from the downloaded Rockwell Automation® library files via FactoryTalk® View SE/ME. The workflow is the same for both FactorTalk View ME and SE.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following video for this section: "How_To_Import_and_Configure_Safety_Device_Objects_in_FTViewME"

All image and display items can be imported either by right-clicking in FactoryTalk® View on the Graphic sub-folder (e.g. Displays, Global Objects, Images) or simply dragging and dropping the files into the application.



Import files in this order:

1. Import HMI Images files.

   Select all the images in the *\HMI FactoryTalk View Images - png* folder and Open.

2. Import Global Object files

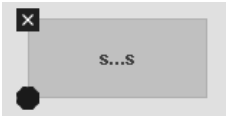
   Select the global object (.ggfx) files from the *\HMI - FactoryTalk View ME\Global Objects - ggfx* or *\HMI - FactoryTalk View SE\Global Objects - ggfx* folder

3. Import HMI Faceplates

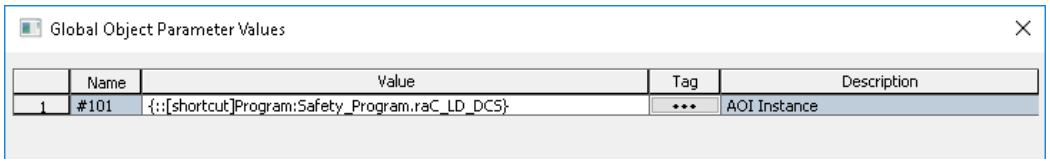
Select the faceplate (.gfx) files from the *\HMI - FactoryTalk View ME\Displays - gfx* or *\HMI - FactoryTalk View SE\Displays - gfx* folder

Configuring FactoryTalk® View Objects

Once the files have been imported into the FactoryTalk® View Studio project, you can begin using them in your application. Open the *Global Display (raC-3-ME) Graphic Symbols - Safety Device* or *(raC-3-SE) Graphic Symbols - Safety Device*. Copy the desired launch button style and paste it into a display in your application where you would like to open the faceplate. For more information on graphic symbols, refer to the Graphic Symbols section of the specific device type chapter in this manual.



To configure the graphic symbol launch button, right-click and select *Global Object Parameter Values*. The Global Object Parameter values for the AOI Instance (#101) is mandatory. You can browse for the tag in your controller project by clicking '...' or manually type them in. These parameters may vary depending on the graphic symbol used, please refer to the Graphic Symbols section of the device type for detailed information.



These Global Object Parameter Values are automatically configured when you use Studio 5000® Application Code Manager to design and configure your project. Refer to Using Studio 5000® Application Code Manager for more information.

Library Upgrades

Add-On Instruction Upgrades

There are two methods to upgrading existing device object add-on instructions in a project. You can do this either by using the Studio 5000 Plug-

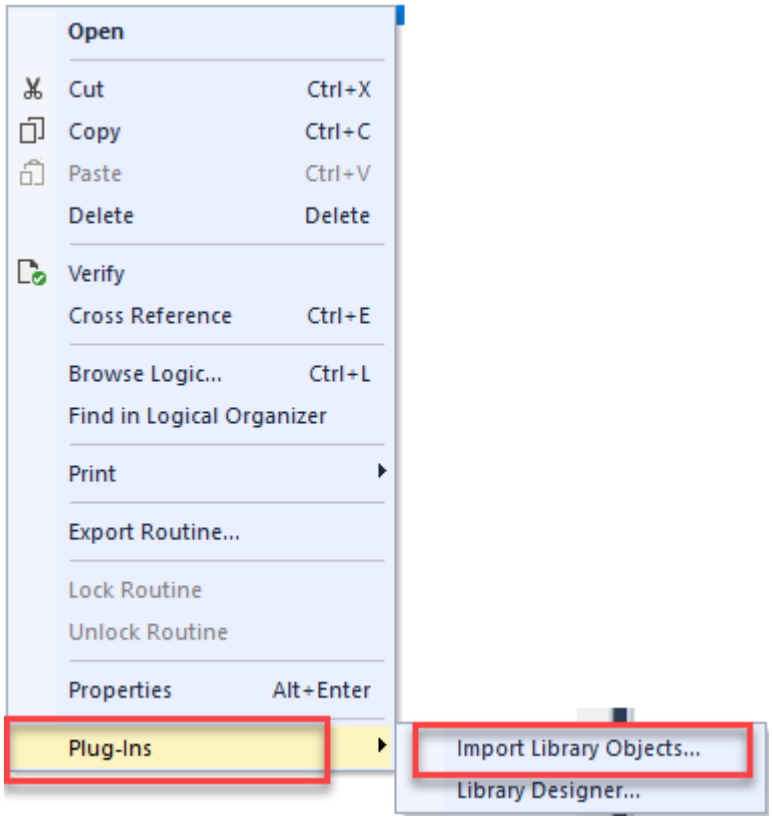
In *Import Library Objects* Wizard or by importing individual add-on instruction AOI.L5X files. Both methods are described in the following sections.

Note that all updates to Add-On Instructions must be done with Studio 5000 Logix Designer® in OFFLINE mode and a download to the controller is required.

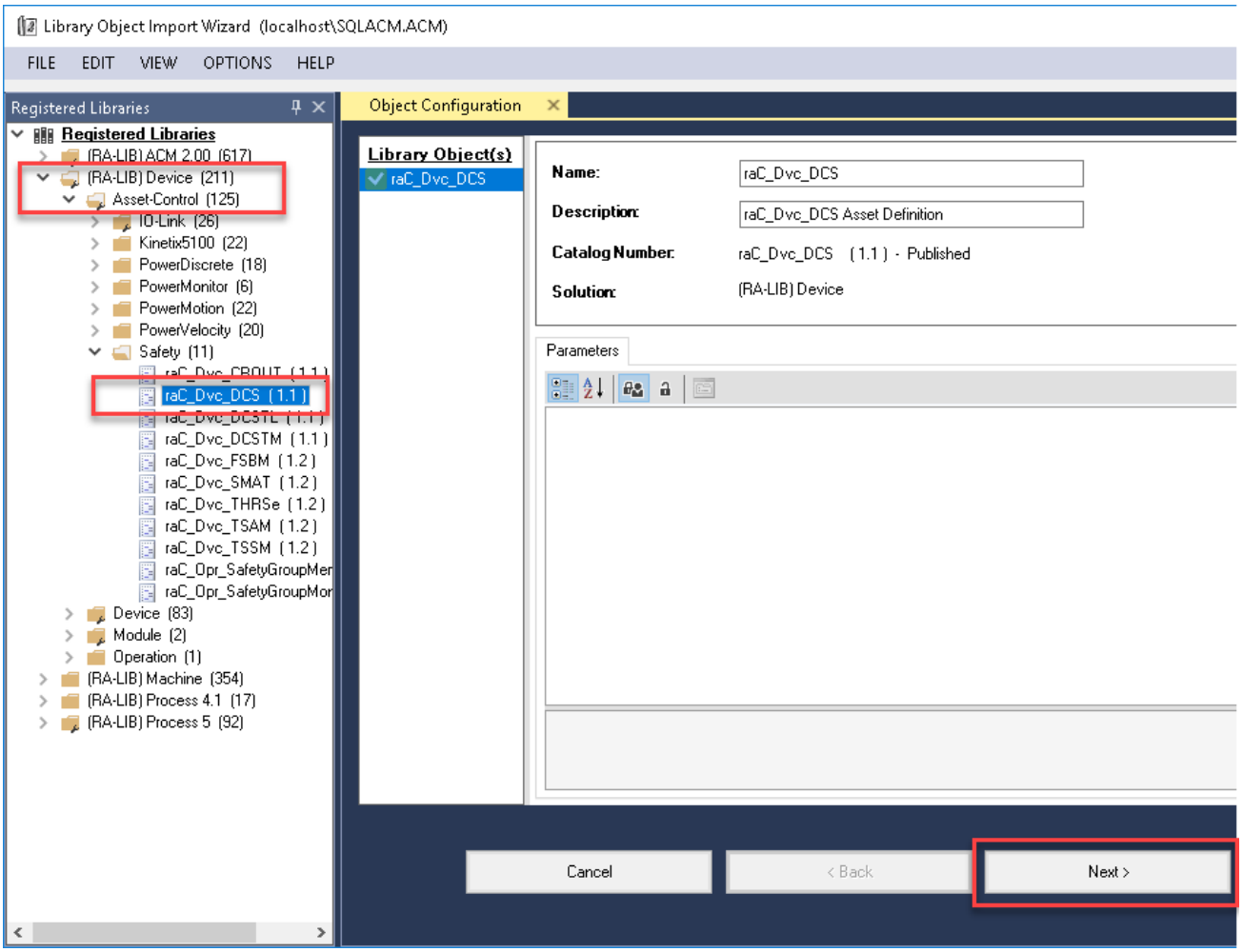
Upgrades Using Studio 5000 Plug-In to Import Library Objects

If Studio 5000® Application Code Manager is installed, you can use the Studio 5000 Plug-In *Import Library Objects* Wizard to update existing Add-On Instructions. For complete information on Studio 5000® Application Code Manager, refer to the section Using Application Code Manager.

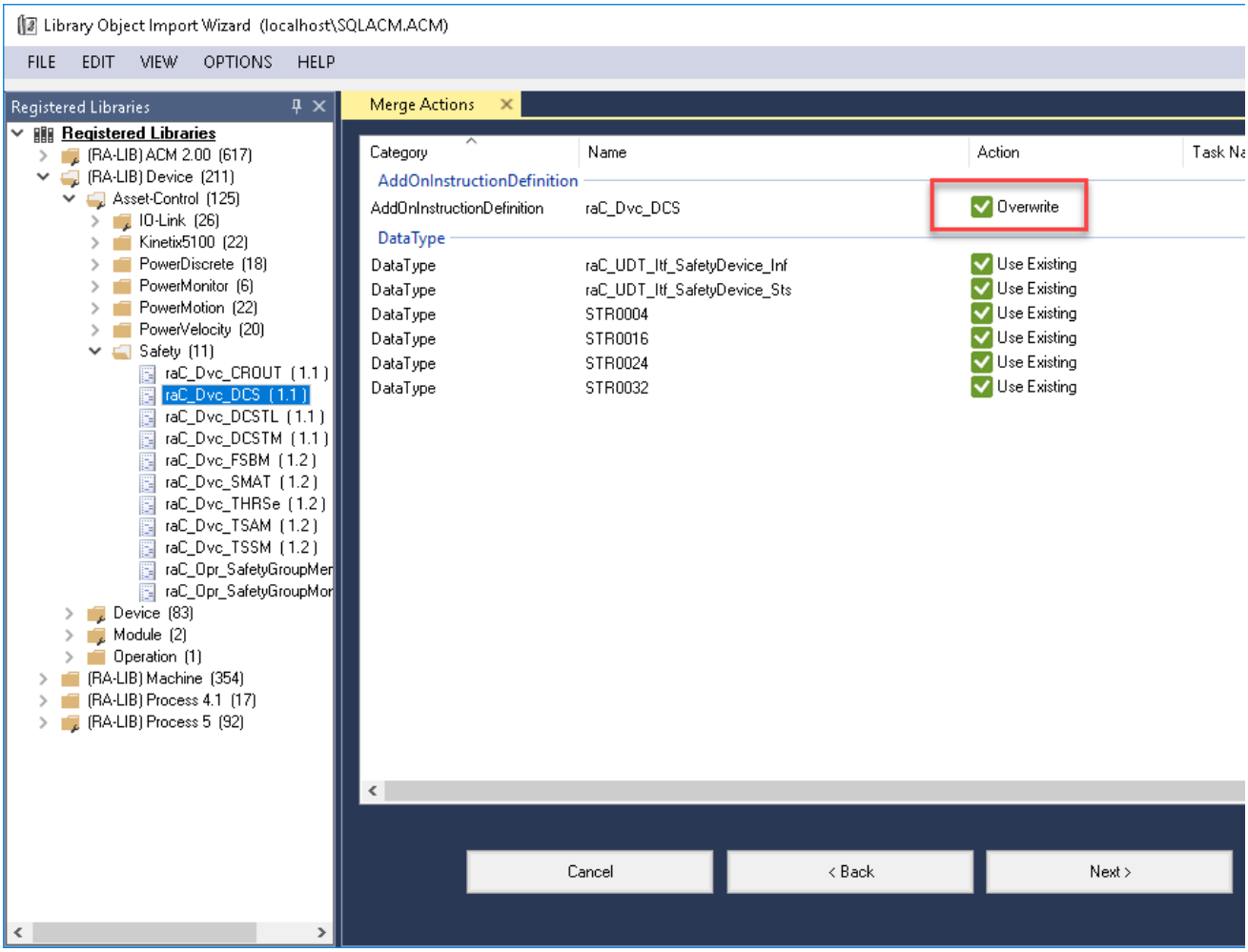
Right-click in your controller organizer or within a routine to access *Plug-Ins > Import Library Objects...*



The *Library Object Import Wizard* dialogue window will open. Under *Registered Libraries* expend *(RA-LIB) Device > Asset-Control* and find the desired object and version. Drag the object into the *Object Configuration* window on the right. Click Next to continue.



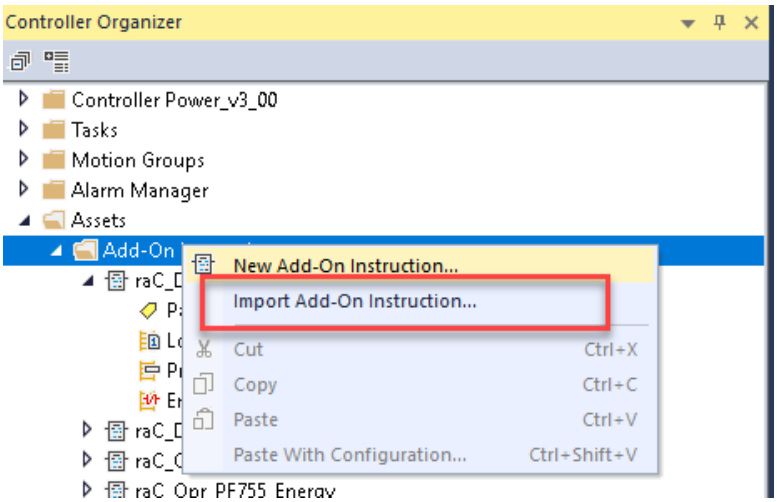
In the *Merge Actions* window, select the *Action* for the *AddOnIntructionDefinition* to *Overwrite*. This will update any existing instance of the object to the newer version. You may also choose to overwrite any other DataTypes or Tags. Review the release notes of the latest library release to understand what may be impacted. Click next and finish to complete the process.



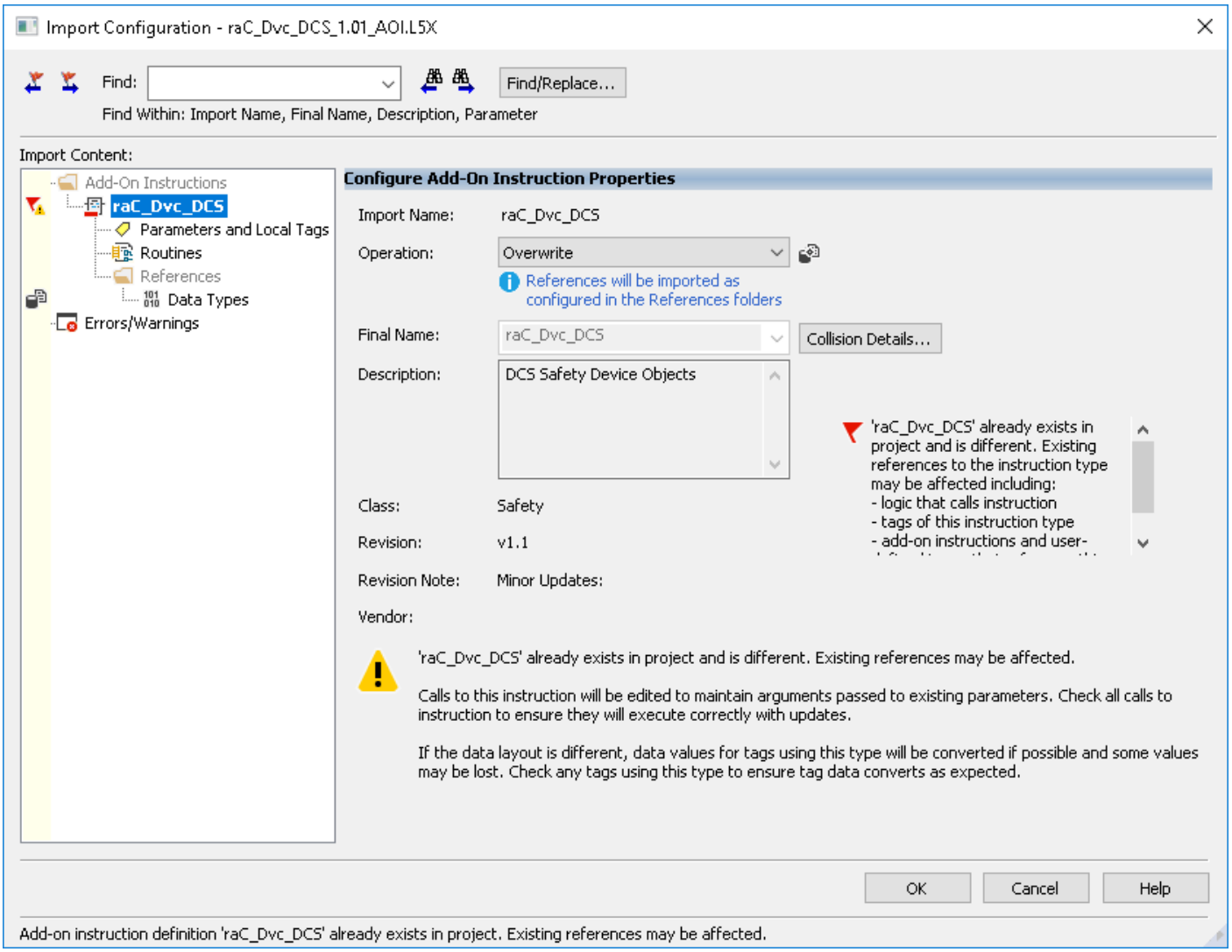
Upgrades by Importing AOI.L5X Files

To upgrade or migrate a project that uses a previous library version to a newer one, the add-on instruction L5X files are supplied. To perform an upgrade to an object perform the following steps:

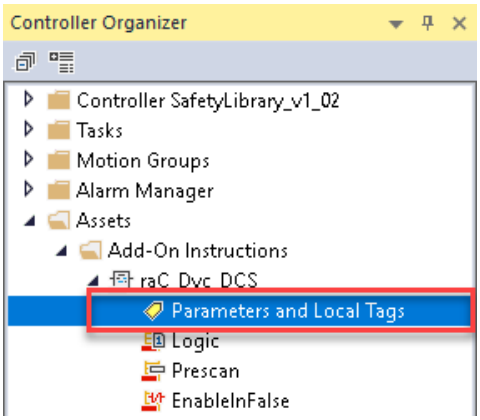
- Open the controller file. Note changes must be done offline.
- In the *Controller Organizer* pane right-click on *Assets > Add-On Instructions* and select *Import Add-On Instruction*. Navigate to the AOI.L5X file in the *Studio 5000 Logix Designer® Files - L5X* and Open.



- You will be prompted that there is an existing version of the instruction that is different. Choose *Overwrite* as the operation and select OK once you have read and understood the warnings. Your existing logic will be updated with the new add-on instruction. Verify that your code compiles and test adequately.

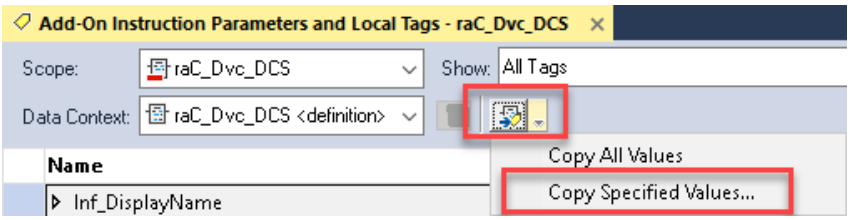


- In order to ensure the HMI faceplate still works properly you will need to update the object's library information stored in the Inf_Lib tag. In the *Controller Organizer* pane under *Assets > Add-On Instructions* expend the device object that was updated. Double-click on *Parameters and Local Tags* to open up the instructions tags.

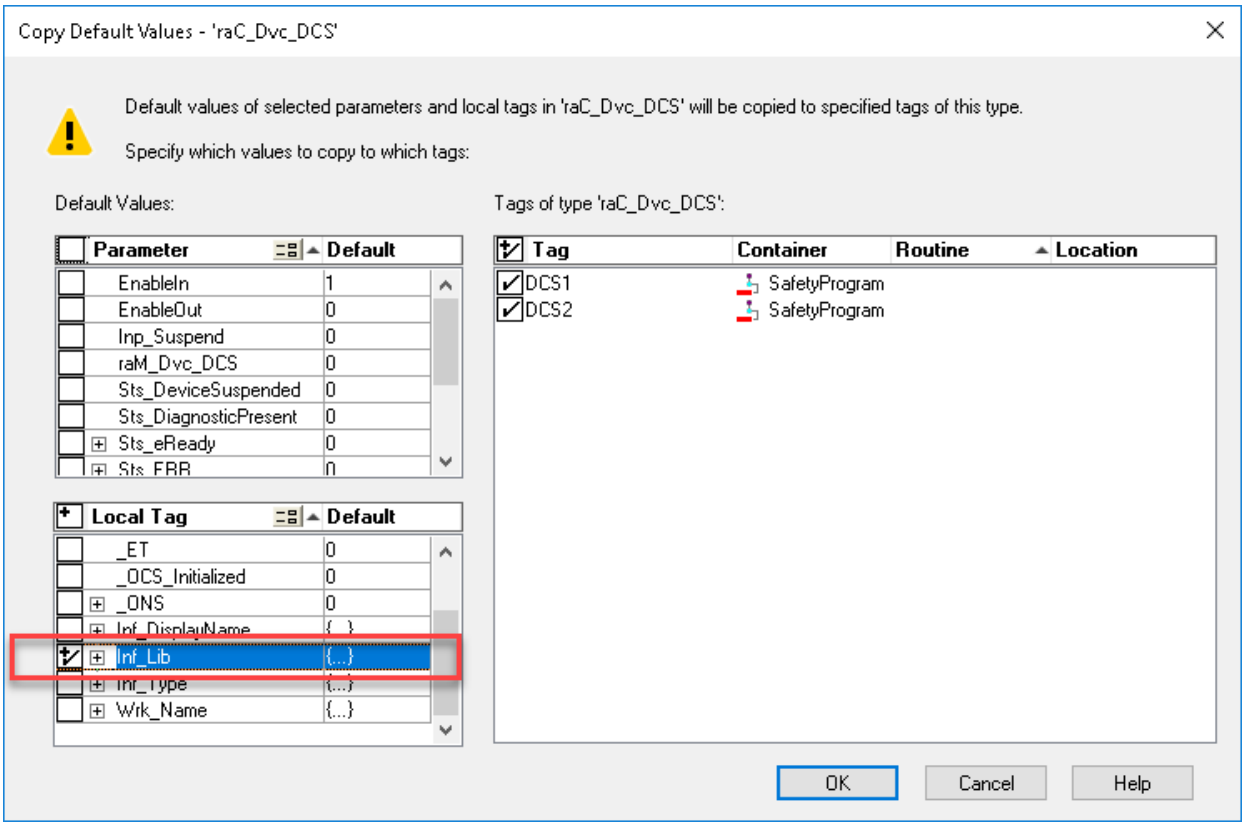


- In the *Add-On Instruction Parameters and Local Tags* window, you may notice that the Inf_Lib tag in the add-on instruction definition matches

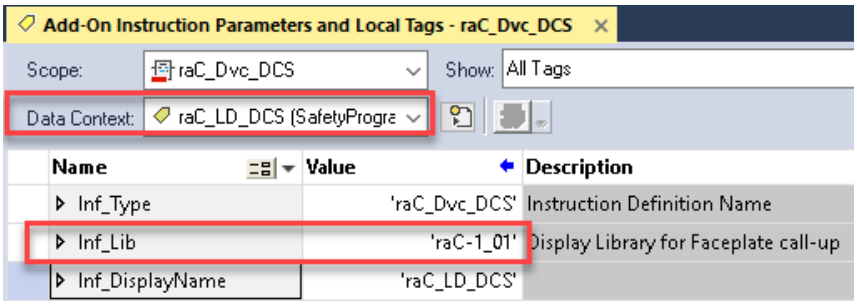
the new library revision number. Click on the down-arrow to the right of the copy button and select *Copy Specified Values...*



•



- In the *Copy Default Values* window, be sure to **first uncheck all Parameters and Local Tags** by clicking the +/- box in the top right. Failure to do so may result in overwriting settings in the existing objects.
- Check only *Inf_Lib* in the *Local Tag* area. On the right, all affected objects should be selected. Click OK.
- You can now confirm that the *Inf_Lib* tag has been updated to the current library (e.g. 'raC-3_01') by changing the *Data Context* drop-down to a specific device object.



FactoryTalk® View Upgrades

To upgrade a device object in a FactoryTalk® View ME/SE application, simply import the new faceplate .gfx display file into the application. If any global objects or images have been added or modified, you may need to import these as well. Any unused displays from previous versions may be removed or deleted from the application.

Note that the reference to the faceplate version is set in the Add-On Instruction Local Tag *Inf_Lib* so there does not need to be other modifications to the HMI application.

Studio 5000 View Designer® Upgrades

To upgrade a device object in a Studio 5000 View Designer® application, simply open the new View Designer .vpd file and copy the raC_Dvc_xxxxx_FP pop-up screen into the existing application. Find any graphic symbol launch buttons in the application that open the faceplate, and update the Action to open the new pop-up screen. Any unused pop-up screens from previous versions may be removed or deleted from the application.

Using Application Code Manager

Overview of Application Code Manager

Studio 5000® Application Code Manager is a tool that enables more efficient project development with libraries of reusable code. Application Code Manager creates modular objects with customizable configuration parameters using the reusable content. Application Code Manager can also create the associated visualization, historical and alarming elements for a project.

Studio 5000® Application Code Manager can be easily used along with Rockwell Automation® application code libraries such as the PlantPAx® Process Objects Library, Machine Builder Library, and Device Object Libraries. For more information on Studio 5000® Application Code Manager, refer to the Application Code Manager User Manual.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following video for this section: "How_To_Import_and_Add_Safety_Device_and_Operation_Objects_in_ACM"

Creating a New Project

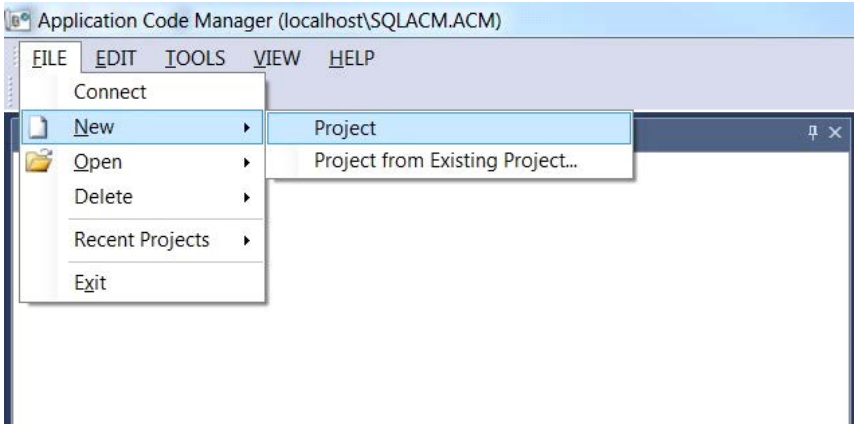
Begin by opening Application Code Manager.

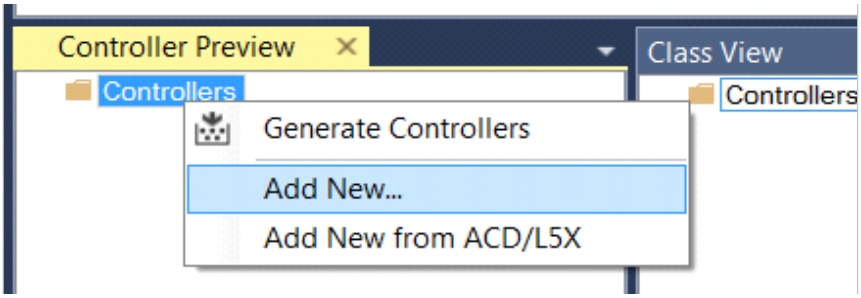
Note: the last project (if any) is opened by default; otherwise a blank screen is displayed.

Create a New Project or open an existing project. Navigate to *File > New > Project*.

Select the desired project type (e.g. (*RA-LIB) ACM 2.00 Project - Basic_Project*) and fill in the *Name* and *Description*.

To add a new controller to a project, in the *Controller Preview* window, right-click on *Controllers* and select *Add New...*

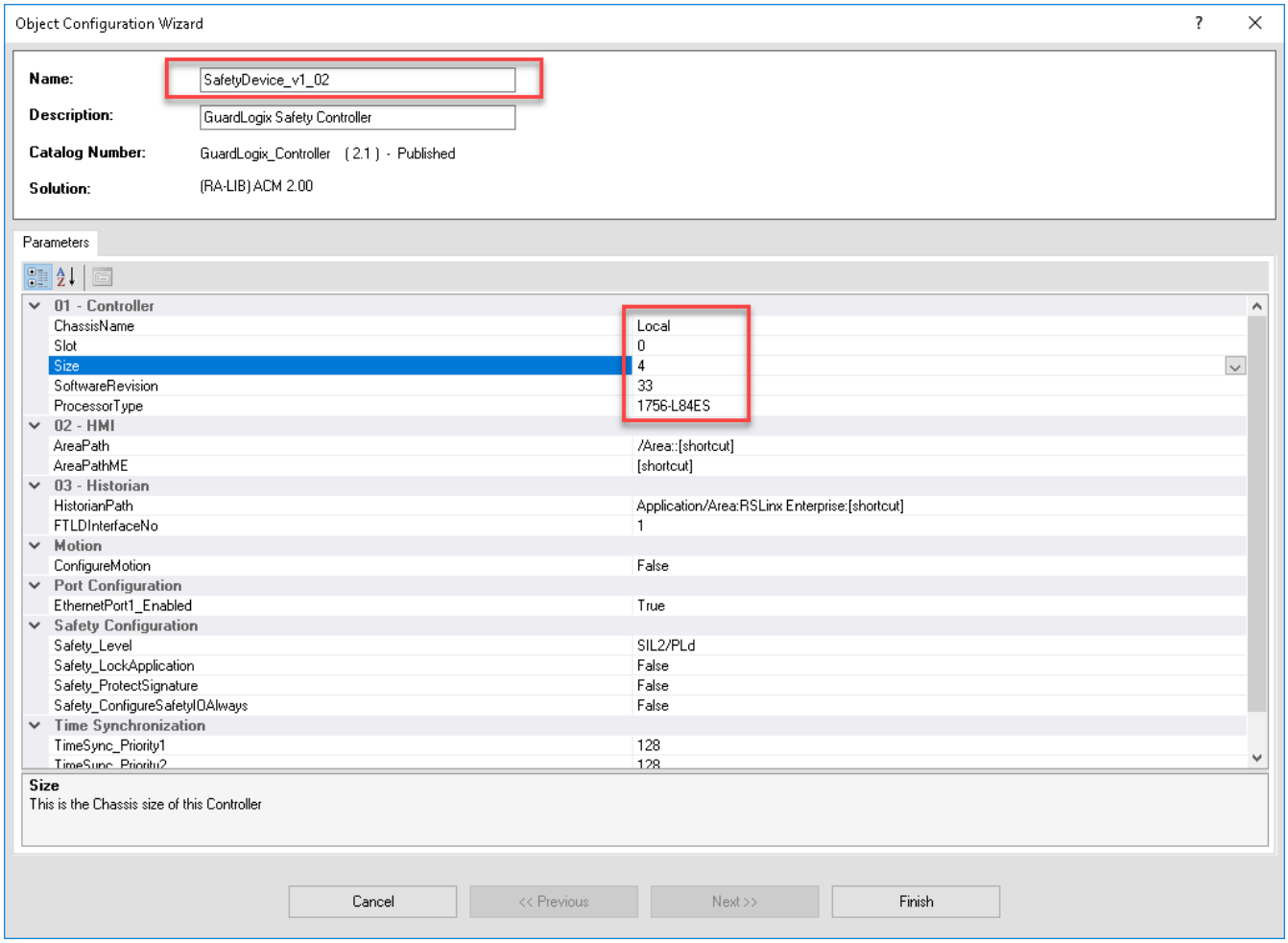




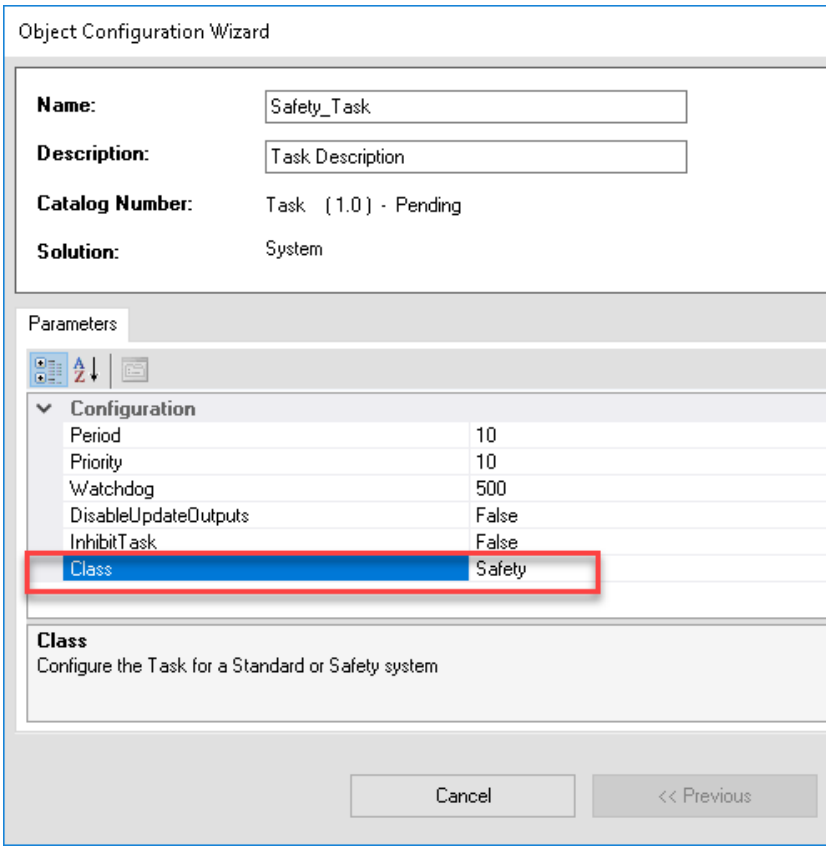
Select the desired controller type (e.g. *GuardLogix_Controller* or *Compact_GuardLogix_Controller*). Enter a *Name* and *Description* for the controller. Select the appropriate Chassis and Processor configurations.

Safety Controllers and Safety Tasks must be selected when using the Safety Device Library objects.

You can also configure the HMI *AreaPath* and.or *AreaPathME* parameters which will be referenced if you use Application Code Manager to generate FactoryTalk® View ME/SE displays with graphic symbol launch buttons.



You can now add in any desired tasks and programs to your controller. A Safety Class Task must be used with Safety Device Library Objects. Right-Click on the *Tasks* folder underneath your controller in the *Controller Preview* and *Add New Task.* Set the *Class* parameter to *Safety*. Similarly, right-click on any Task and select *Add New Program*. All programs under a Safety Class Task will be Safety Class Programs. Complete the desired parameters for Tasks and Programs such as name, type, period, etc.



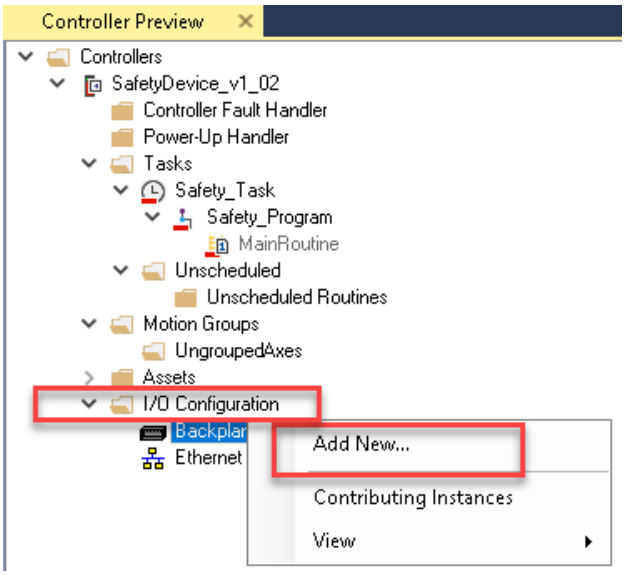
Adding & Configuring Device Objects

Prior to adding in any Device Objects, ensure you have registered the library in Application Code Manager. Refer to Registering Libraries in Studio 5000® Application Code Manager for details.

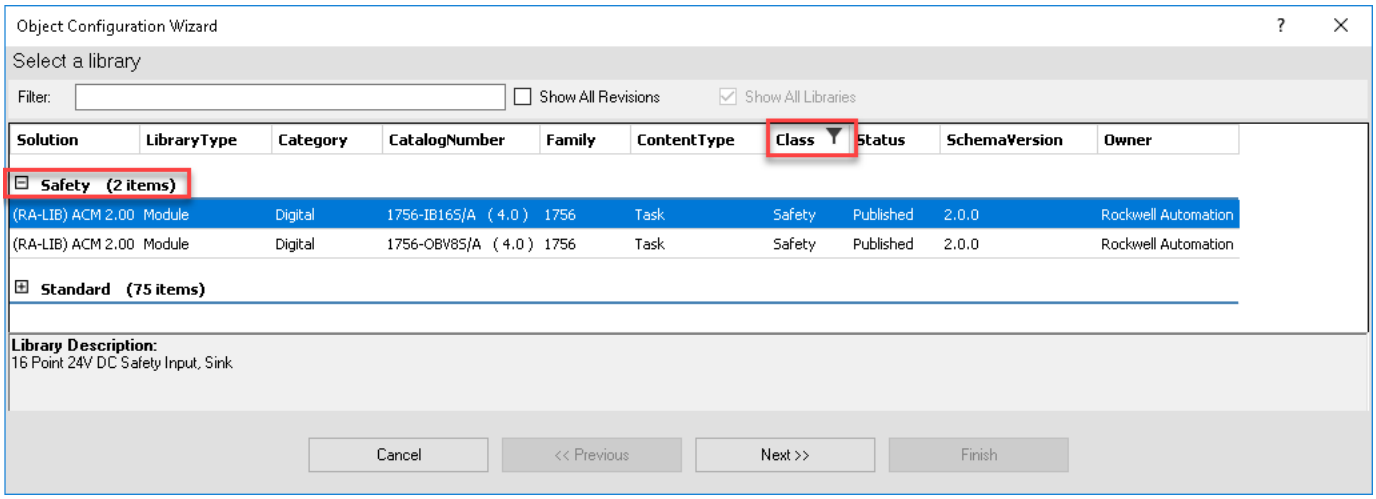
Adding Safety I/O

If not already done you may need to add Safety I/O modules to you Controller I/O Configuration. This will allow you to link tags in the Safety Device Library objects to I/O tags. Alternatively, you can replace or link these at a later time.

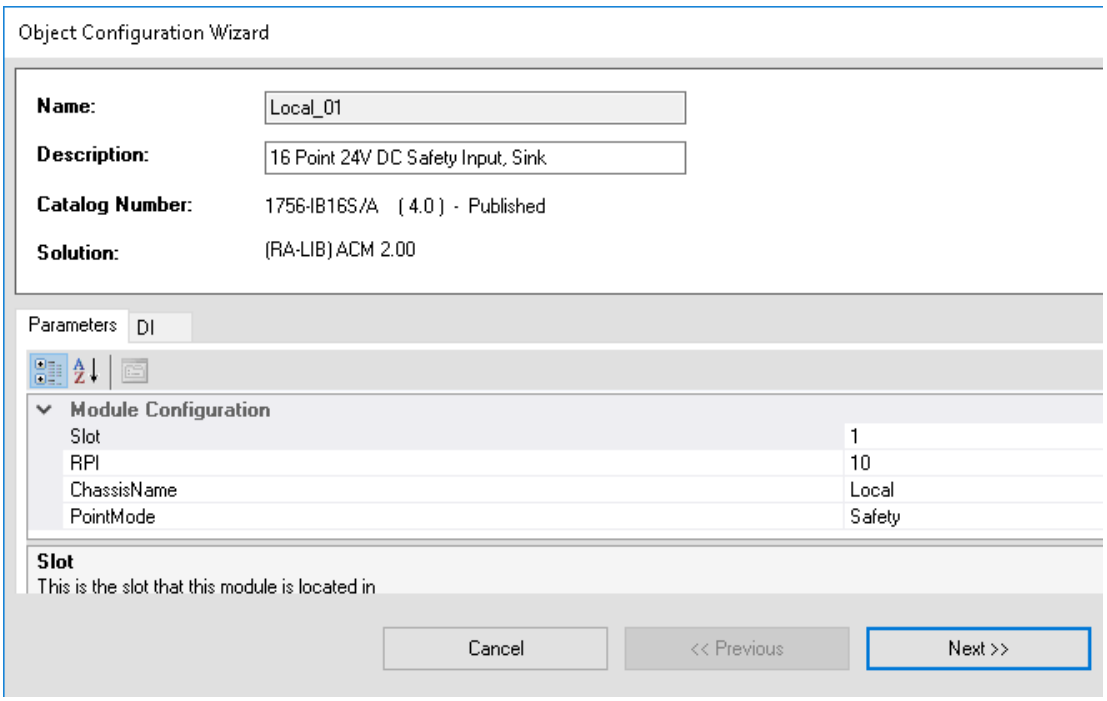
In the *Controller Preview* pane you find the *I/O Configuration* folder underneath your controller. You may add modules directly to the backplane or add remote I/O racks on the Ethernet network. Right-click the desired location and select *Add New*.



You may choose to click on the *Class* filter to easily sort and find *Safety* I/O. Selected the desired Safety I/O module and click *Next*.

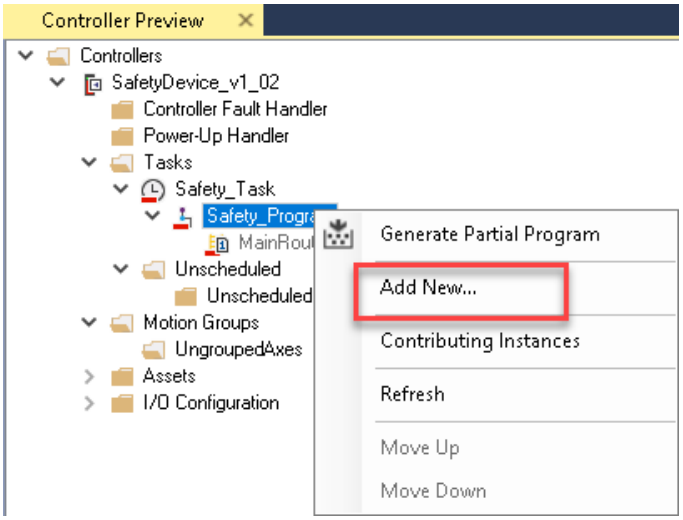


Assign the desired *Slot* and *RPI* parameters for the module and click *Finish*.

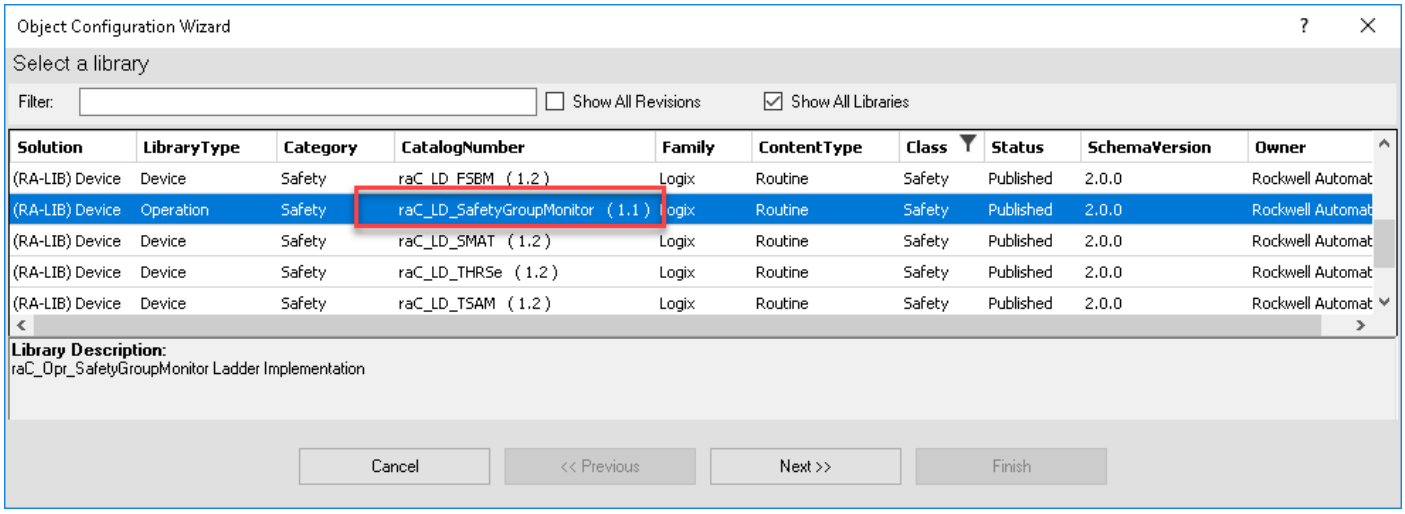


Adding Safety Group Monitor

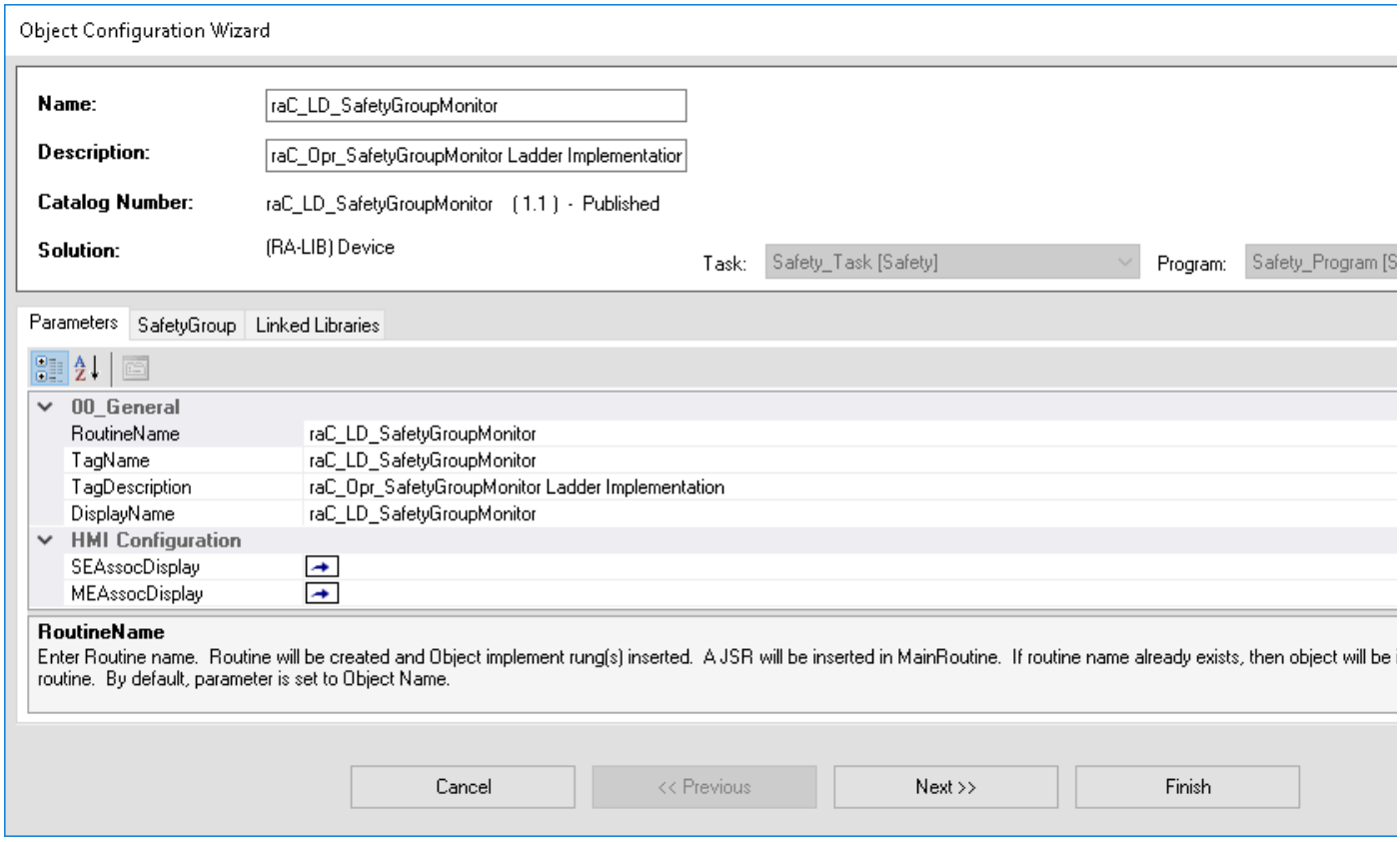
You can optionally add a Safety Group Monitor object prior to adding Safety Device Library instructions to group them into logical systems for monitoring and diagnostic purposes. To add a Safety Group Monitor Object into a project, right-click on a Program and *Add New...*



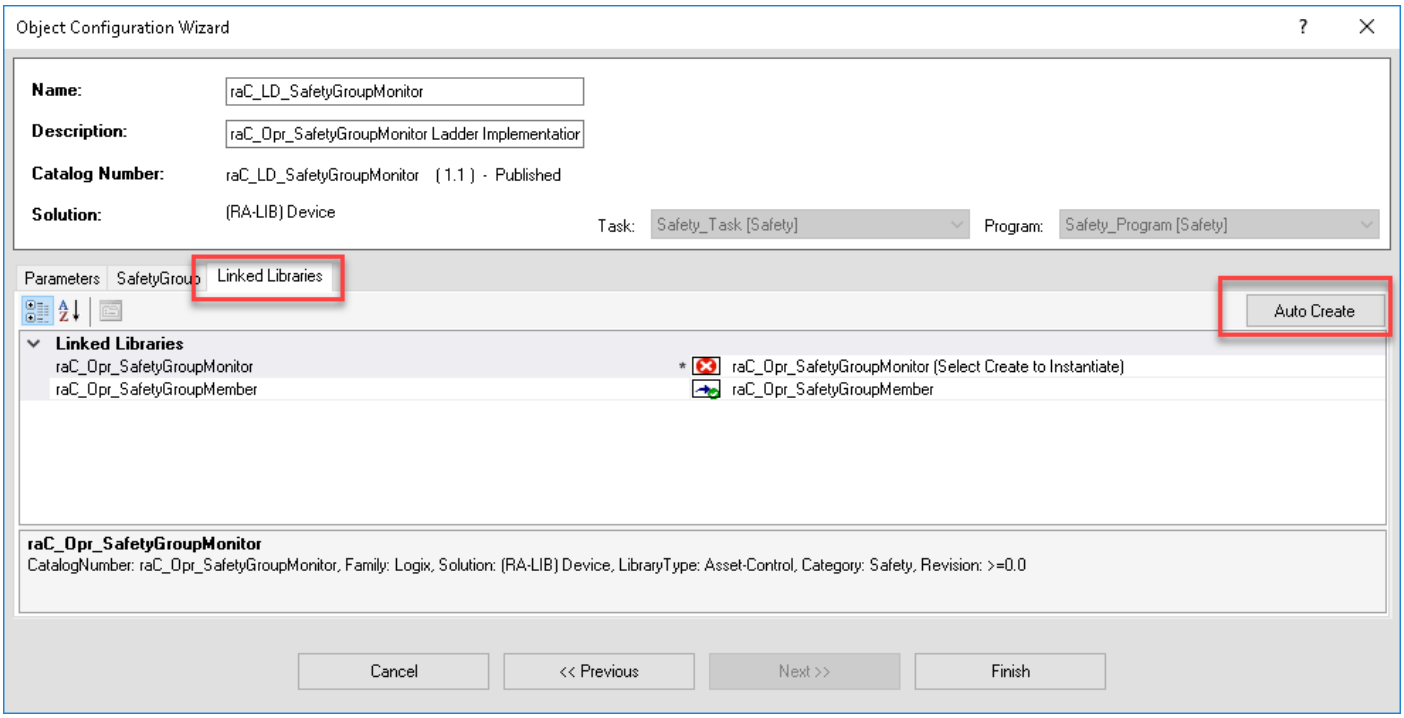
Select *raC_LD_SafetyGroupMonitor* and click *Next*.



- Assign your desired *Name* and *Description* fields.
- For HMI Configuration refer to Configuring Displays.

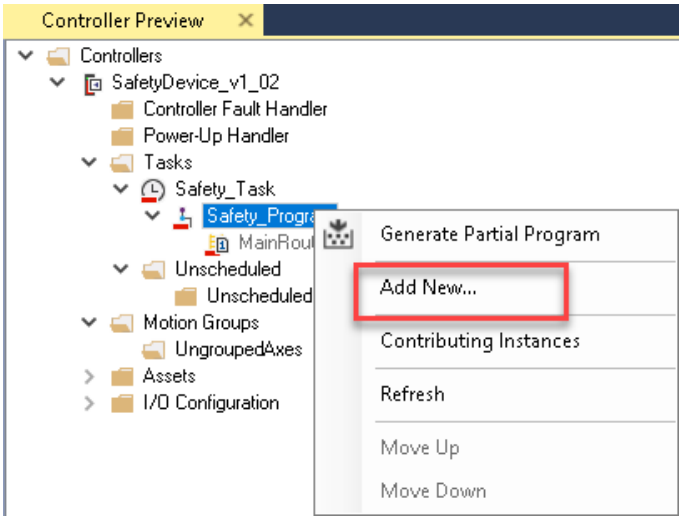


- The *SafetyGroup* tab is not required unless you would like to cascade multiple Safety Group Monitors into another Safety Group Monitor.
- Click on the *Linked Libraries* tab and click *Auto Create*. Complete by clicking *Finish*.

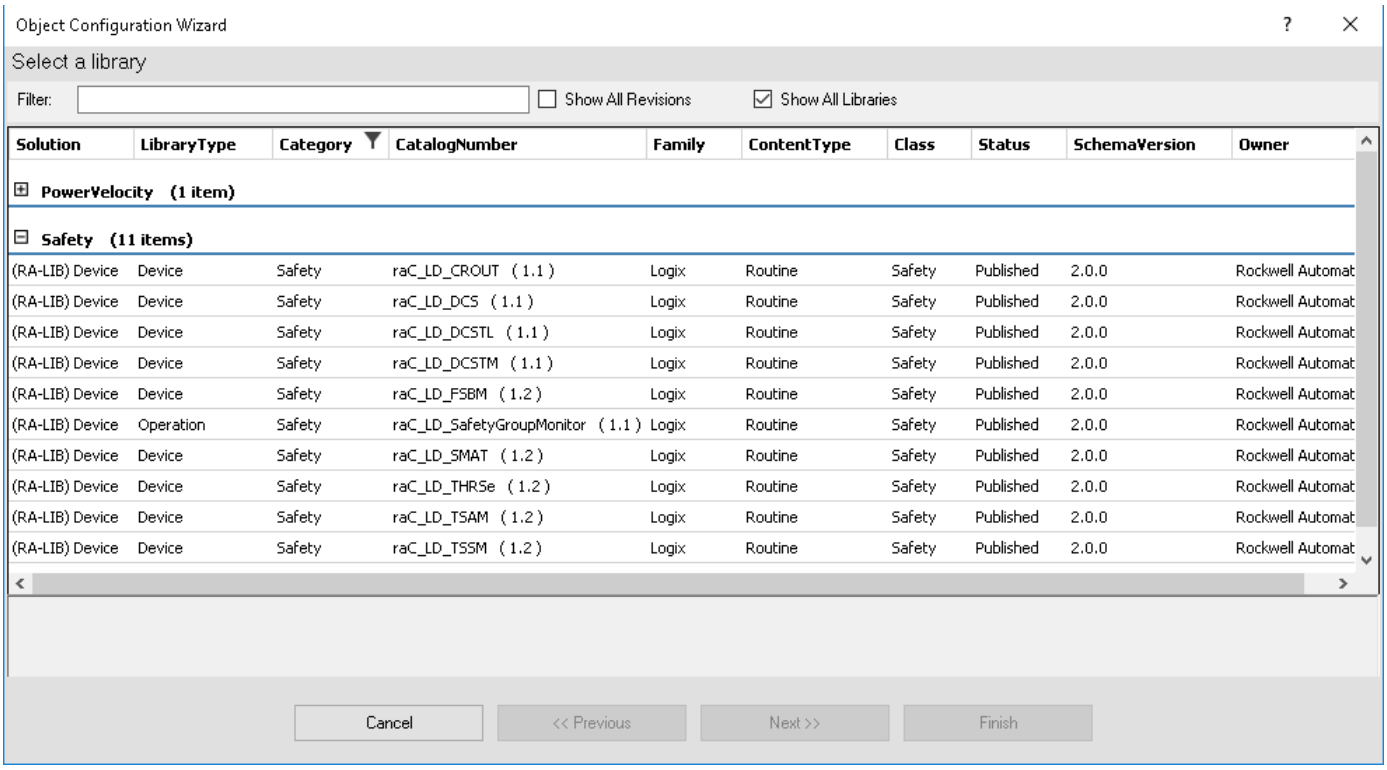


Adding Safety Device Instructions

To add a Device Object into a project, right-click on a Program and *Add New...*



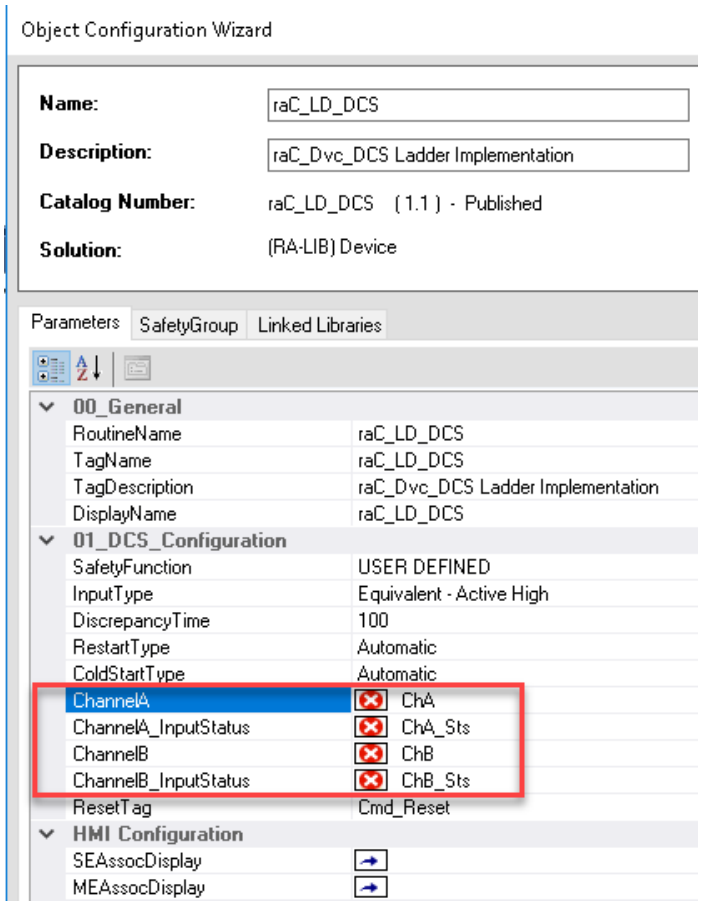
Select the Device Object that you wish to import. Note that you should select the *Device* library type rather than the *Asset - Control* library type (e.g. raC_LD_DCS) where LD stands for Ladder Logic.



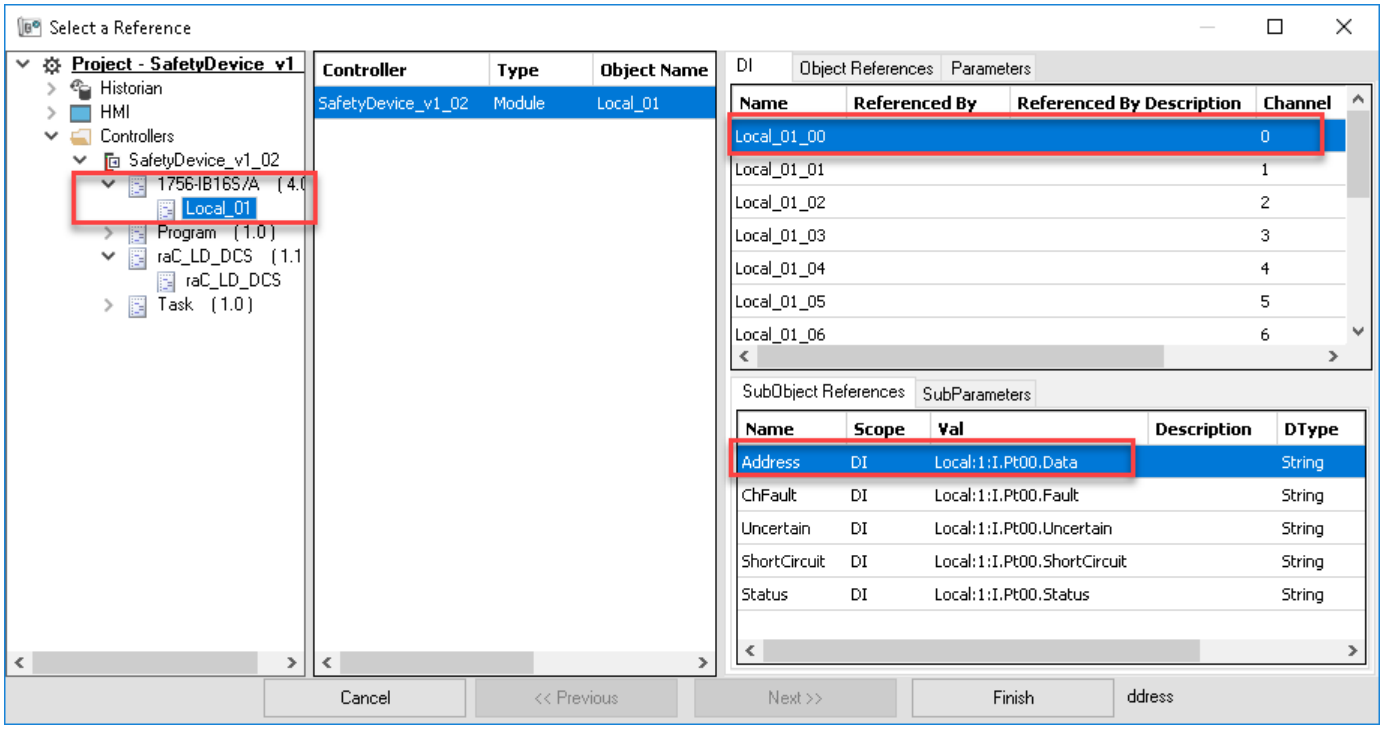
Fill in all of the required configuration parameters for the device object. The following example shows a configuration of the raC_LD_DCS object.

Perform the following configuration:

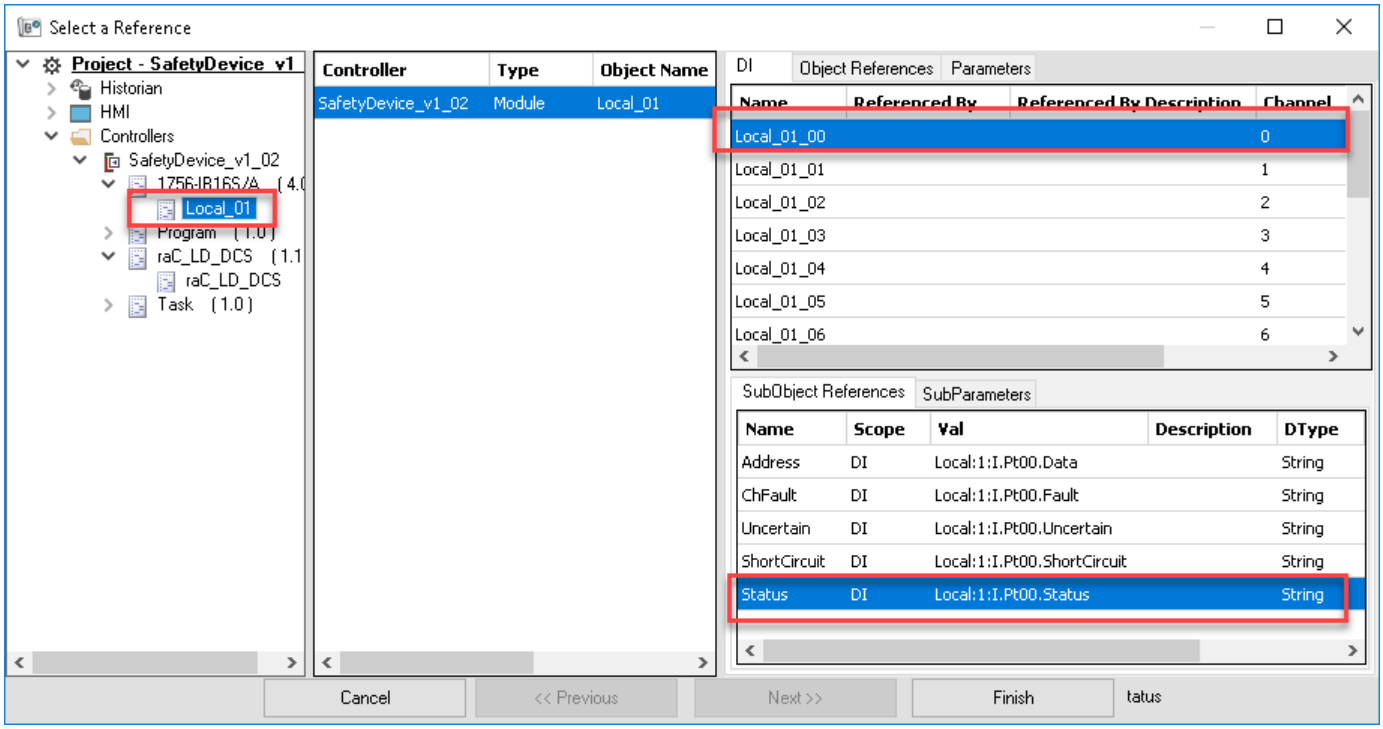
- Enter a **name** and **description**. Maximum name length can be 22 characters. Note that other parameters such as the RoutineName, TagName, etc will auto-complete based on these fields.
- Assign the **Task** and **Program**.
- Assign a **DisplayName** which is used as the faceplate title and object tag description.
- Configure the instruction. Each safety device instruction has unique configuration which may include settings related to feedback, reaction times, command tags, and IO tags. IO tags by default will be set with placeholders (e.g. ChA, Out1, Fdbk2, etc). These tags may be left as-is and updated to physical IO tags later, or modified at this time. Red X's will be shown if the default values are left - note that this is acceptable.



- To assign Safety I/O module tags to these instruction configuration parameters click the ... browse button. In the left pane, navigate to and highlight the instance of the Safety I/O module. In the top-right pane in the *DI* tab select the desired I/O channel number. In the bottom-right pane in the *SubObject References* tab select the *Address* reference for I/O tags.



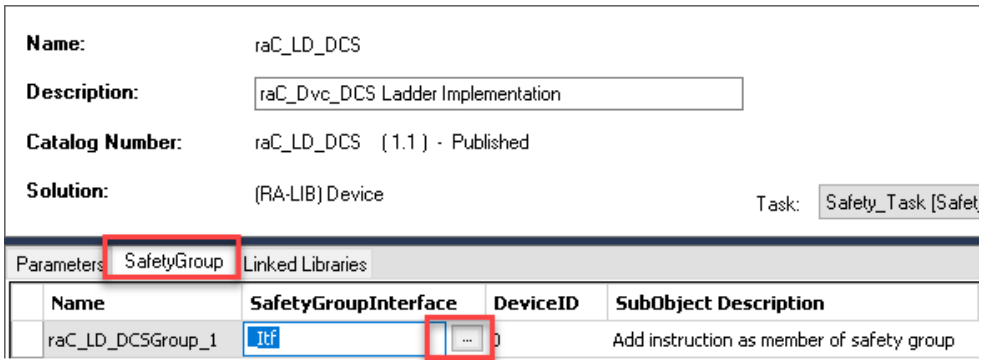
- Similarly for status tags (e.g. ChAStatus) browse to the I/O module and channel number and instead select the *Status* SubObject Reference.



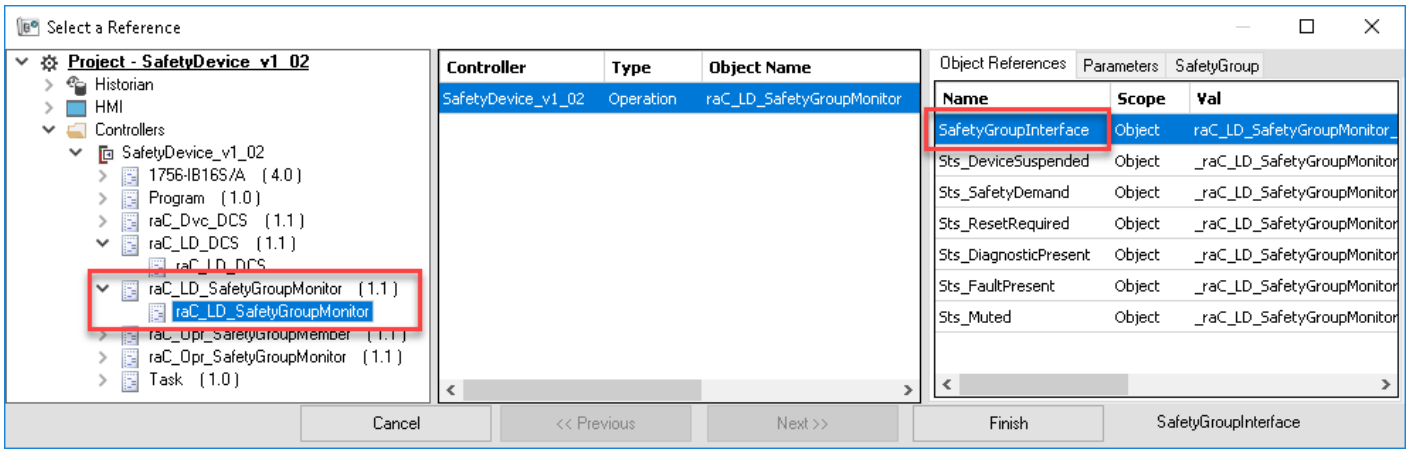
- Once complete, all of the default tags will be replaced with I/O tags and any red X's will be removed.

ChannelA	Local_01.DI.Local_01_00#Address
ChannelA_InputStatus	Local_01.DI.Local_01_00#Status
ChannelB	Local_01.DI.Local_01_01#Address
ChannelB_InputStatus	Local_01.DI.Local_01_01#Status

- For HMI Configuration refer to Configuring Displays.
- If you would previously created a Safety Group Monitor object and would like to link this instruction to that group click on the *SafetyGroup* tab.
- Click on the ... to browse fore a *SafetyGroupInterface*.

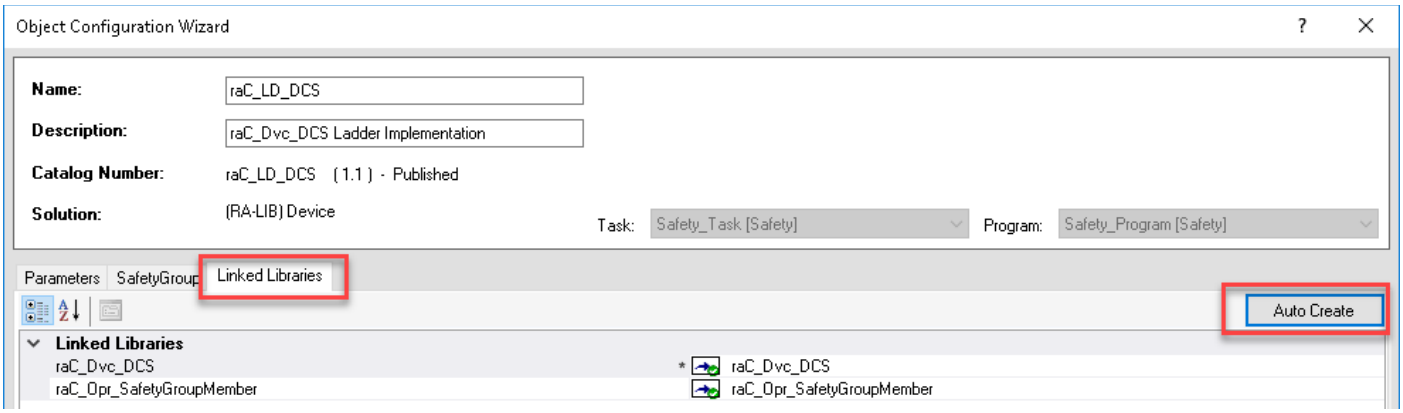


- In the *Select a Reference* dialogue window, find and highlight the instance of the *raC_LD_SafetyGroupMonitor* in the left pane. In the right pane under the *Object References* tab highlight the *SafetyGroupInterface* reference and click *Finish*.



- Back in the *SafetyGroup* tab of the safety instruction you can assign a *DeviceID* to this particular instruction in the Safety Group. This must be a unique number that has not already been assigned to another instruction and you must keep track of which numbers have been used. Each Safety Group Monitor supports up to 32 instructions (0-31).

- Click on the *Linked Libraries* tab. Click the *Auto Create* button to automatically create all of the required linked libraries.



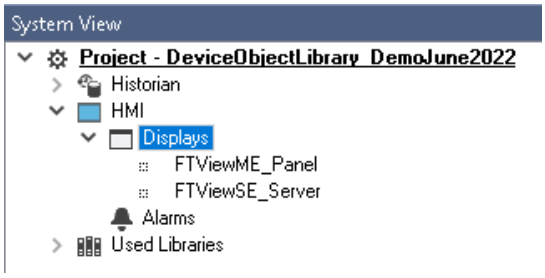
- Click Finish to complete the import.

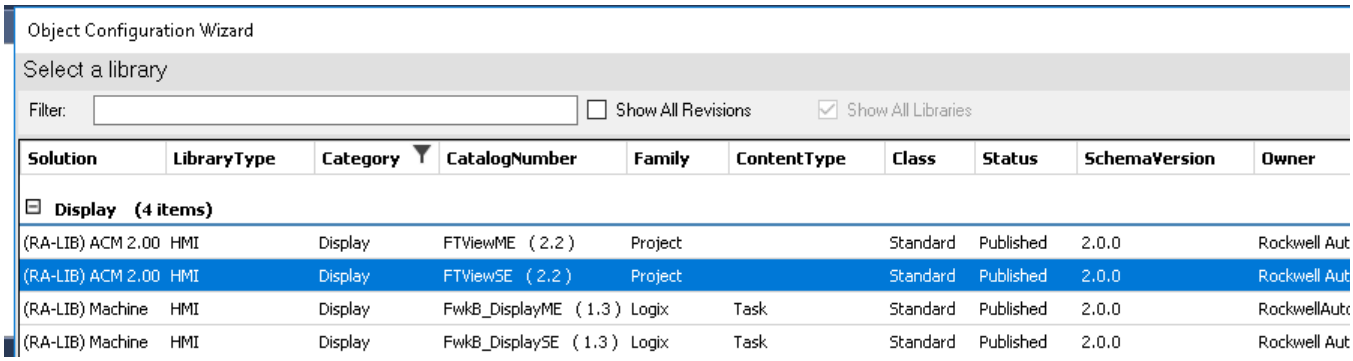
For specific devices details, refer to the appropriate chapter in this manual.

Configuring Displays

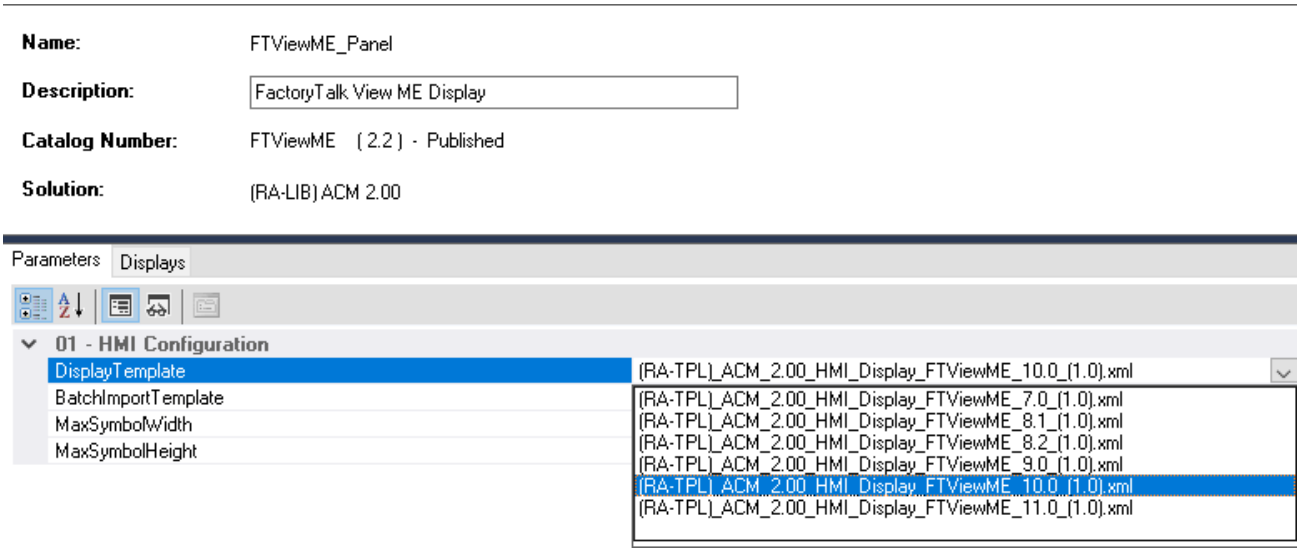
Application Code Manager can be used to automatically configure graphic symbol launch buttons for device objects In FactoryTalk® View ME or SE. Note that Application Code Manager is not compatible with Studio 5000 View Designer® applications.

First you must add Displays to your project. Under the *System View* panel expand *HMI* and right-click on *Displays* to select *Add*. Choose the type of display (e.g. *FTViewME* or *FTViewSE* depending on your project requirements.

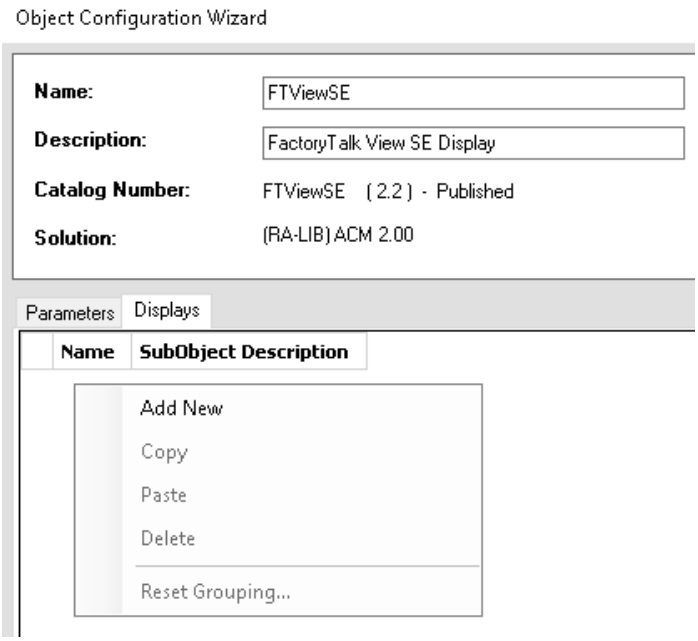




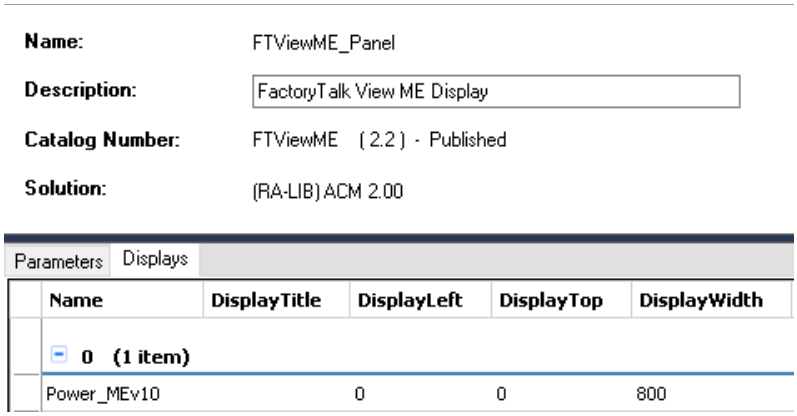
In the display object parameter configuration, you must select the *DisplayTemplate* type to match the version of FactoryTalk® View application that you are using.



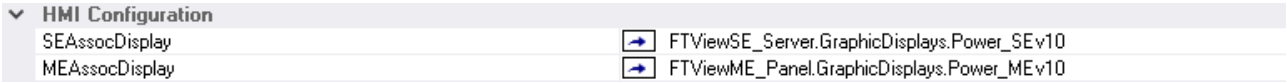
Navigate to the *Displays* tab where you can right-click and *Add New* display.

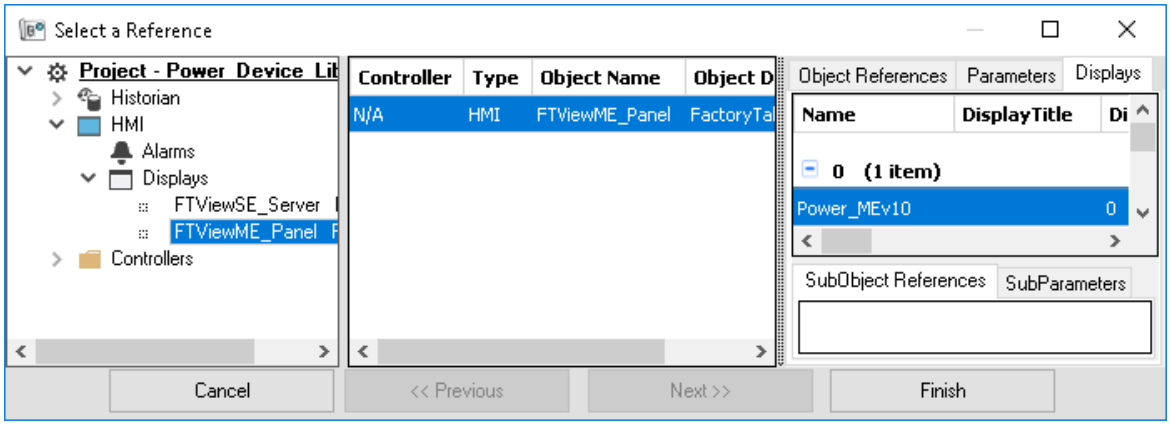


Set the desired name and display parameters. Generally all display parameters aside from *Name* can be left as default since this will often be used as a temporary display where object launch buttons are copied from.



Return to your device object configuration and view the *HMI Configuration* section of the parameters. You can browse or type in the HMI_Server_Name.HMI_Display_Name.



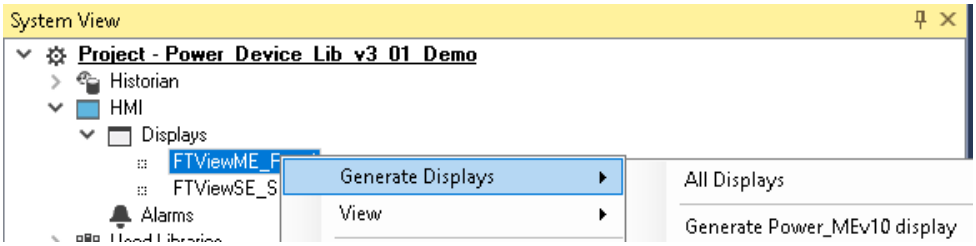


If you browse for the display, select the desired display server in the left panel, then click on the *Display* tab in the right panel and select the specific display. Click finish.

This workflow can be followed for either FactoryTalk® View ME or SE depending on the project requirements.

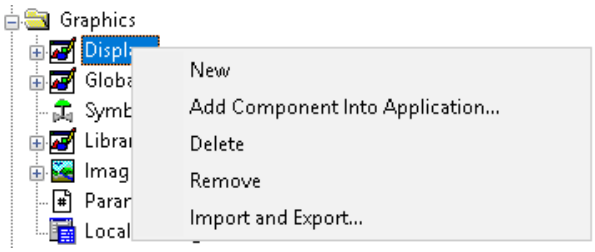
Generating Displays

Once you have assigned displays to all of the device objects, you can generate the displays. In the *System View* highlight the desired display server under *HMI > Display* and right-click to select *Generate Displays > All Displays* or select individual displays. Choose a place to save the generated files and take note of it.



Importing Displays into FactoryTalk® View Studio

To import the configured displays, open your FactoryTalk® View ME/SE project in FactoryTalk® View Studio. Right-click on *Graphic > Displays* and select *Import and Export...*

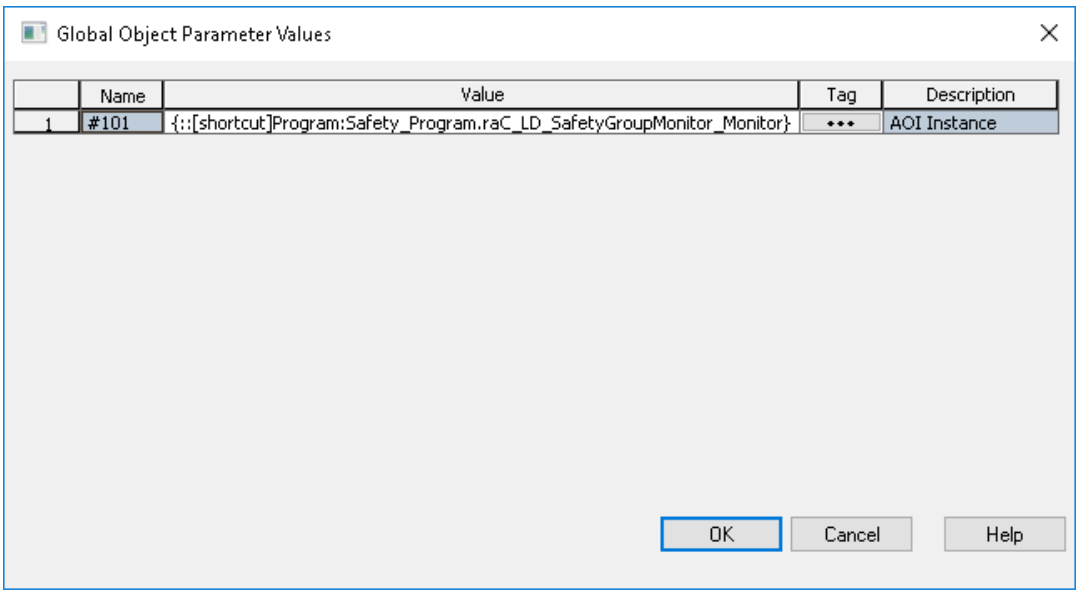


Follow the required prompts:

- Import graphic information into displays
- Choose whether or not to backup displays
- Choose either a *Single display import file* (must have an existing or blank display to import into) or *Multiple displays batch import file* if *All Displays* was used to Generate Displays.
- If this is the first time it is recommended to import *Multiple displays batch import file* and then *Create new objects on the display*.
- If you have done this before and are updating the imported display after modifying your Application Code Manager project, you can choose *Update existing objects on the display*.
- Browse for the BatchImport.xml file or individual display.xml file.

Open up the newly imported display. Notice that there are graphic symbol launch buttons labeled and configured for each item that was configured in Application Code Manager.

Right-click on the object and select *Global Object Parameters* to view that all of the parameters have been pre-configured for you.

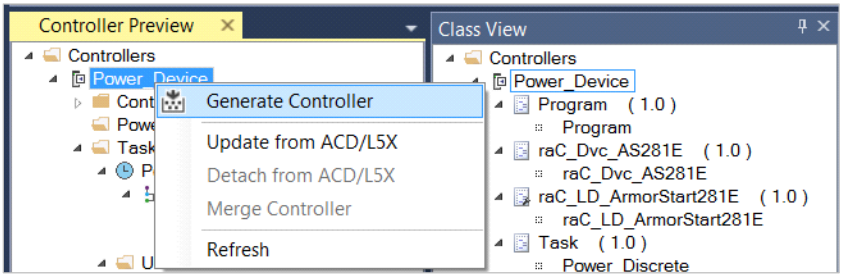


You may not copy and paste this graphic symbol onto any other display in your application.

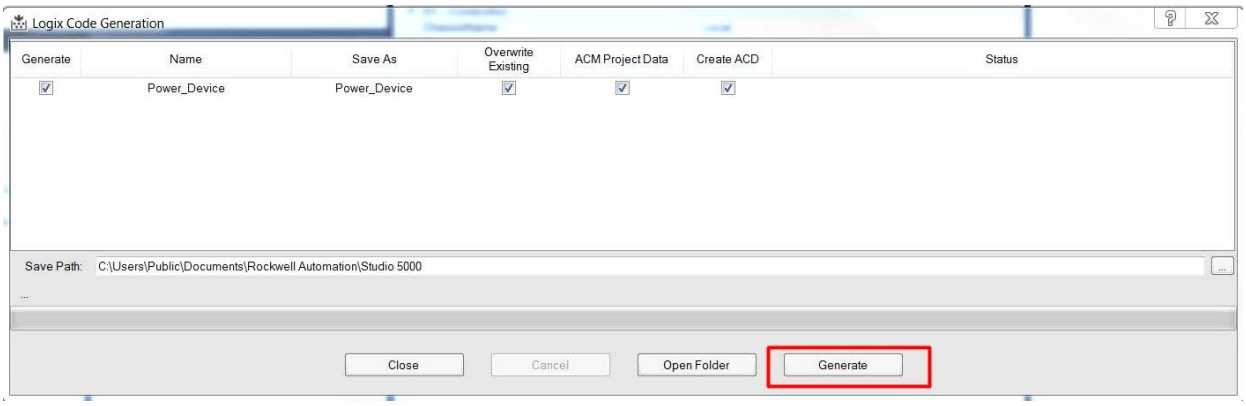
Generating Controller Files

Once you have completed configuring your project in Studio 5000® Application Code Manager, you can generate the controller file for use in Studio 5000 Logix Designer®.

In the *Controller Preview* pane right-click on the controller name within the *Controllers* folder and select *Generate Controller*.



In the *Logix Code Generation* dialogue window you will need to check *Create ACD*. You ma also need to check *Overwrite Existing* if this is not the first time generating the controller code.



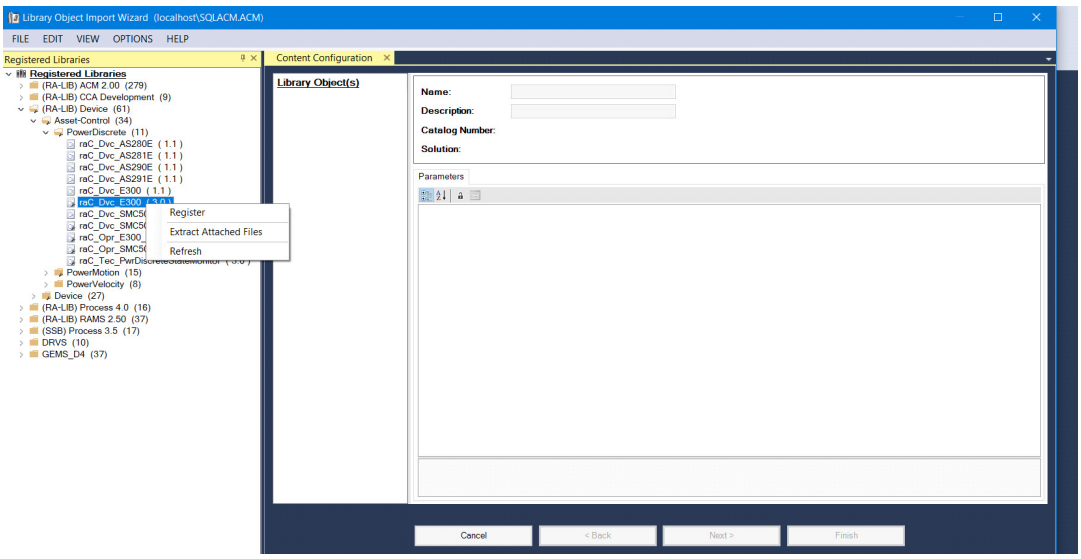
Once the controller file is generated, you can navigate to the location set in *Save Path* and open your file. Note that all of the configuration that was done in Application Code Manager is now shown in your Logix Designer ACD file.

Exporting Attachments

Application Code Libraries not only contain Logix code, but also contain Visualization collateral and associated documentation. Every Asset library contains at least a reference manual (RM). Those libraries which have associated Visualization content also have all required global objects (GO), images, static displays and View Designer applications added as attachments. In this manner the user can generate only the necessary visualization and documentation for the objects included in the project.

In Application Code Manager, all of the attachments are associated with the device objects in the *(RA-LIB) Device > Asset-Control* folder. These can be accessed both through the full Application Code Manager software, or via the Studio 5000 Plug-In “Import Library Objects”.

To access the attached files, right click on the objects (e.g. raC_Dvc_DCS) and select *Extract Attached Files*.



Select the destination folder on your computer, and select OK. An Extract Attachments dialog will show the extraction status.

The extracted folder will contain the following:

- Reference Manual
- Required Images
- Studio 5000® View Designer Faceplate Files
- FactoryTalk® View Machine Edition Display
- FactoryTalk® View Machine Edition Global Objects
- FactoryTalk® View Site Edition Display
- FactoryTalk® View Site Edition Global Objects.

Chapter 5

Using the Safety Device Library with Other Application Code Libraries

Application Code Libraries

The Safety Device Library is can be used alongside other Application Code Libraries.

The IO Device Library is recommended to be used along with the Safety Device Library if Studio 5000® Application Code Manager is used for project development. This will allow you to link Safety Device Object instructions to Safety IO module tags.

The Machine Builder Library and PlantPAx® Process Object Library application-level library objects may be used in the same applications with the device-level objects in the Safety Device Library. At this time there are no direct dependencies or interaction points between these libraries. They may be used independently from one another but within the same application.

Other libraries utilize the common device interface UDTs to interact with device level objects. In the case of the Safety Device Library you may programmatically reference the Status (Sts) interface of the instructions. This is covered in detail in Interfaces section of this document.

DCS - Dual Channel Input Stop (raC_Dvc_DCS)

Overview

The Dual Channel Input Stop device object (raC_Dvc_DCS) includes a faceplate which displays status and configuration information of dual-input safety devices whose main function is to stop a machine safely. Common examples include E-stops, light curtains, area scanners, safety mats, cable (rope) pull switches or safety gates.

The instruction is ready when both safety inputs Channel A and Channel B are in the active state as determined by the Input type operand, and the correct reset actions are carried out.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following videos for this section:
"Operational_Overview_of_Safety_Device_Object_Faceplates"
"Operational_Overview_of_Safety_Application_Example_in_LogixDesigner"

Functional Description

The Dual Channel Input Stop pre-configured Device Objects:

- Collect, Process and Deliver Data between Smart Devices and Application Logic
- Provide Device Status & Diagnostics Faceplates for Machine Startup, Commissioning, Operations, and Maintenance
- Include Common HMI Faceplate Components, Device States, and Control Interfaces providing Application Development and Operation Efficiency

All these features provide quick feedback, shorten recovery time, and simplify implementation.

Required Files

Device Objects include Add-On Instructions (AOIs) and HMI faceplates. The revision number (e.g. 1.01) used in filenames can change as new revisions are created.

Controller Files

Add-On Instructions are reusable code objects that contain encapsulated logic that can streamline implementing your system. This lets you create your own instruction set for programming logic as a supplement to the instruction set provided natively in the ControlLogix® firmware. An Add-On Instruction is defined once in each controller project, and can be instantiated multiple times in your application code as needed.

The Add-On Instruction must be imported into the controller project to be used in the controller configuration. These can be imported as Add-On Instruction files, or as part of the ROUTINE Import or Import Library Objects wizard.

All Add-On Instruction and ROUTINE Import files can be found in the */Studio 5000 Logix Designer® Files - L5X/* folder in the library.

Device/Item	Add-On Instruction	ROUTINE Import
DCS	raC_Dvc_DCS_1.01_AOI.L5X	raC_Dvc_DCS_1.01_ROUTINE.L5X

FactoryTalk® View HMI Files

FactoryTalk® View ME or SE applications require importing the desired device faceplates in addition to all Global Object (ggfx) files and all images located in the */HMI FactoryTalk View Images - png/* folder of the library. FactoryTalk® View ME files are stored in the */HMI - FactoryTalk View ME/* library folder and FactoryTalk® View SE files are stored in the */HMI - FactoryTalk View SE/* library folder.

Device/Item	Type	FactoryTalk® View ME Faceplate	FactoryTalk® View SE Faceplate
DCS	Display	(raC-1_01-ME) raC_Dvc_DCS-Faceplate.gfx	(raC-1_01-SE) raC_Dvc_DCS-Faceplate.gfx
Graphic Symbols	Global Object	(raC-1-ME) Graphic Symbols - Safety Device	(raC-1-SE) Graphic Symbols - Safety Device.ggfx
Toolbox	Global Object	(raC-1-ME) Toolbox - Safety Device.ggfx	(raC-1-SE) Toolbox - Safety Device.ggfx

Studio 5000 View Designer® HMI Files

All Studio 5000 View Designer® Files can be found in the */HMI - ViewDesigner - vpd/* folder of the library.

Device/Item	Studio 5000 View Designer® Faceplate
DCS	(raC-1_02-VD) raC_Dvc_Safety.vpd

Studio 5000® Application Code Manager Files

Studio 5000® Application Code Manager (ACM) can be optionally used if it is installed. All devices can be easily registered in the ACM repositories by running the *setup.cmd* file located in the root folder of the library.

Individual HSL4 files are provided as an alternative to running the setup.cmd to allow users to manually register specific implementation objects. Each object has two files - an Asset Control file and a Device file. The Asset Control files include attachments of all required files for that object. The Device files

are used to actually add that device into a Studio 5000 project and these reference the Asset Control files.

All Studio 5000® Application Code Manager files can be found in the /*ApplicationCodeManagerLibraries*/ folder of the library. The files included are as follows:

Implementation Object	Asset Control File (.HSL4)	Device File (.HSL4)
DCS	(RA-LIB)_Device_Asset-Control_Safety_raC_Dvc_DCS_(1.2)	(RA-LIB)_Device_Device_Safety_raC_LD_DCS_(1.2)

Operations

The Safety Discrete objects provides only physical operation mode. There is no virtual device mode offered.

Execution

The following table explains the handling of instruction execution conditions.

Condition	Description
EnableIn False (false rung)	Processing for EnableIn False (false rung) is handled the same as if the device were taken out of service by Command. The device outputs are de-energized and the device is shown as Program Out of Service on the HMI. All alarms are cleared.
Powerup (prescan, first scan)	On prescan, any commands that are received before first scan are discarded. The device is de-energized. On first scan, the device is treated as if it were returning from Hand command source: the instruction state is set based on the position feedback that is received from the device. If the feedback is valid for one position, the device is set to that position. If the device does not have position feedback or the position feedback is invalid, the device is set to the 'unknown/powerup' state. The command source is set to its default, either Operator or Program (unlocked).
Postscan	No SFC Postscan logic is provided.

Add-On Instruction I/O Data

InOut Data

InOut	Function / Description	DataType
Inp_DCS	Input Interface - DCS Instruction	DCI_STOP
Out_Ctrl_Inf	Information Interface	raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	Status Interface	raC_UDT_Itf_SafetyDevice_Sts

Input Data

Input	Function/Description	DataType
EnableIn	Enable Input - System Defined Parameter	BOOL
Inp_Suspended	Suspend device operation	BOOL

Output Data

Output	Function/Descritpion	DataType
Sts_DeviceSuspended	Device is suspended	BOOL
Sts_SafetyDemand	Safety demand is present	BOOL
Sts_ResetRequired	Reset is required	BOOL
Sts_DiagnosticPresent	Diagnostic is present	BOOL
Sts_FaultPresent	Fault is present	BOOL
Sts_Ready	Device is ready	BOOL
Sts_ERR	Instruction Error Code	DINT
Sts_eReady	0: Ready 1: Safety Demand 2: Reset Required 3: Suspended	SINT
raC_Dvc_DCS	Unique Parameter Name for auto - discovery	BOOL
EnableOut	Enable Output - System Defined Parameter	BOOL

Data Types

The following Safety Common Control Interface tags are the primary device program tags to read and write to when interfacing to safety devices. The value of using these tags in your specific application code is that you may use a number of different safety devices such as DCS, DCSTL, etc without having to update your application device interface tags. The primary function of these interfaces within the Safety Device Library is to pass individual instruction information to the Safety Group Monitor instruction.

Refer to the Interfaces section for detailed information on interfaces.

raC_UDT_Itf_SafetyDevice_Inf

Member	Description	Data Type
Type	Provides device object type information e.g. raC_Dvc_SafetyGroupMonitor, raC_Dvc_DCS, raC_Dvc_DCSTL etc.	STR0032
DisplayName	Provides object display name	STR0024
eClass	Provide instruction type information eClass ID: Instruction Type 0: DCS 1: DCSTL 2: DCSTM 3: CROUT 5: SafetyGroupMonitor 6: TSAM 7: TSSM 8: FSBM 9: THRSe 10: SMAT 11:SS2 12: SS1 13:SFX 14:STO	DINT

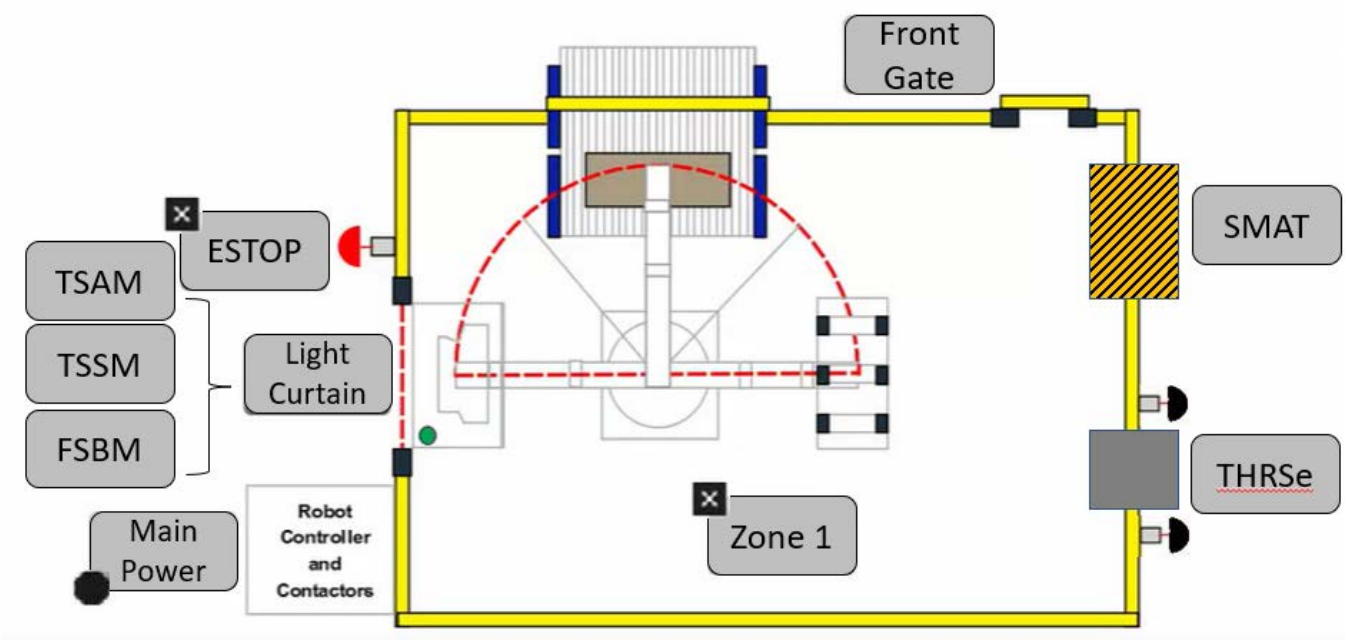
raC_UDT_Itf_SafetyDevice_Sts

This is the Safety Device Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data. The table below shows member names, descriptions, and tag data types.

Input	Description	Data Type
bSts	Bit overlay (Visible) covering all subsequent Boolean members. (Unused)	DINT
DeviceSuspended	Device suspended status (Inp_Suspend = 1)	BOOL
SafetyDemand	Safety demand status ON (1): Safety demand is present on the device object OFF (0): No safety demand	BOOL
ResetRequired	Reset Required status ON (1): Device requires reset signal OFF (0): No reset required	BOOL
DiagnosticPresent	Diagnostic status ON (1): A diagnostic is present in the device object OFF (0): Device is operating normally	BOOL
FaultPresent	Fault Status ON (1): A fault is present in the device object OFF (0): Device is operating normally	BOOL
Muted	Device Muted status (Available for DCSTM) ON (1): Muted status is present on one or more safety instructions within a specific safety group OFF (0): None of the instructions within group is muted	BOOL

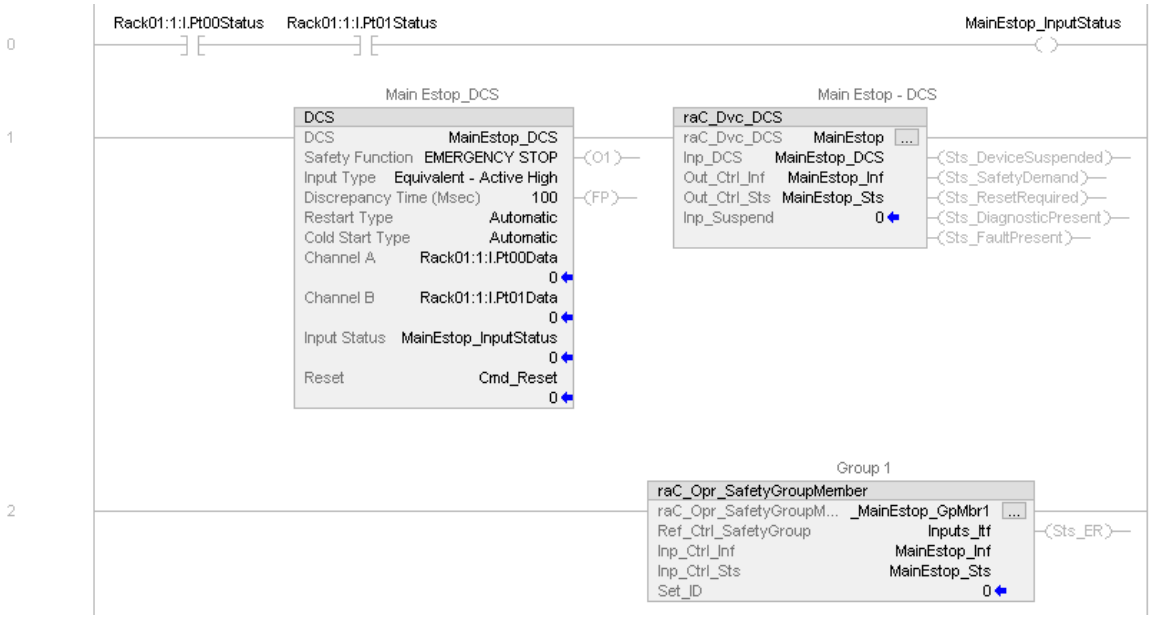
Programming Example

Fully configured device routine is provided below for reference. This example is taken from the *Safety Program - Inputs* routine in the *SafetyApplication.ACD* example application provided in the library.



This example uses the DCS instruction for an e-stop function shown as *ESTOP* in the safe machine diagram. Channel A and Channel B are tied to safety class discrete inputs. Rung descriptions are as follows:

- Rung 0: Check input statuses
- Rung 1: Configure DCS and raC_Dvc_DCS instructions
- Rung 2: raC_Opr_SafetyGroupMember (optional) assigns MainEstop DCS to Inputs Safety Group at ID #0.



Note that this programming example is the same code that is imported when either importing the supplied ROUTINE .L5X files or when using Application Code Manager or the Studio 5000 Import Library Objects wizard plug-in.

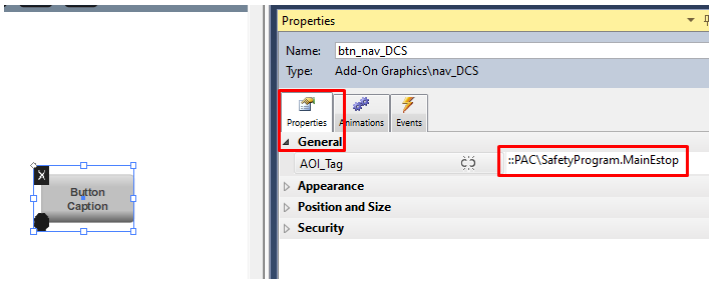
Graphic Symbols

Graphic Symbols are used as launch buttons within HMI applications to open up faceplate displays. See Basic Launch Button Attributes section for details on configuration and indicators.

FactoryTalk® View ME/SE Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Global Object Parameter Values
btn_nav_safety		Faceplate navigation button with string tag label. This launch button graphic object allows the user to navigate to the device object faceplate. The text on the button face is set to the tag's .Inf_DisplayName.	#101: AOI Instance (e.g. {::[PAC]Program::SafetyProgram._InstanceName})

Studio 5000 View Designer® Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Property Configuration
Launch		The supplied launch button in View Designer is used to navigate to the faceplate in a user application.	

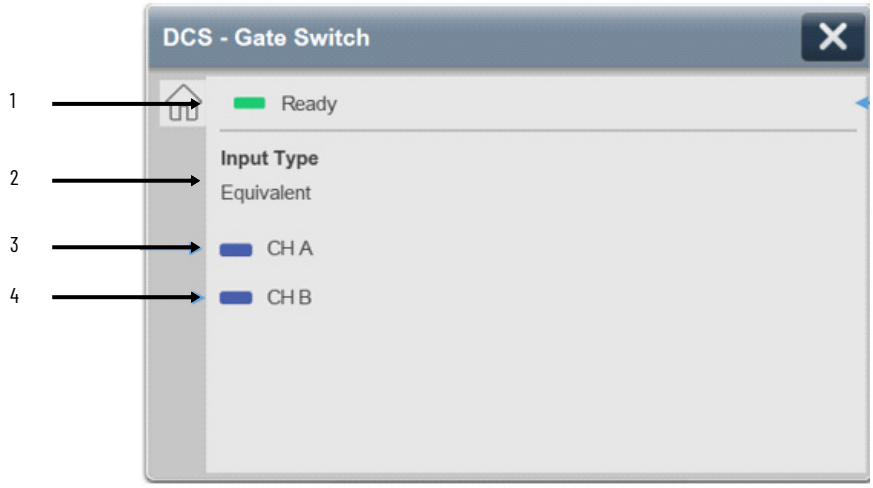
Faceplates

There are basic faceplate attributes that are common across all instructions. See Basic Faceplate Attributes on page 27.

Home

The Home tab is the main tab of the faceplate. It provides the status of the safety input device along with configuration information.

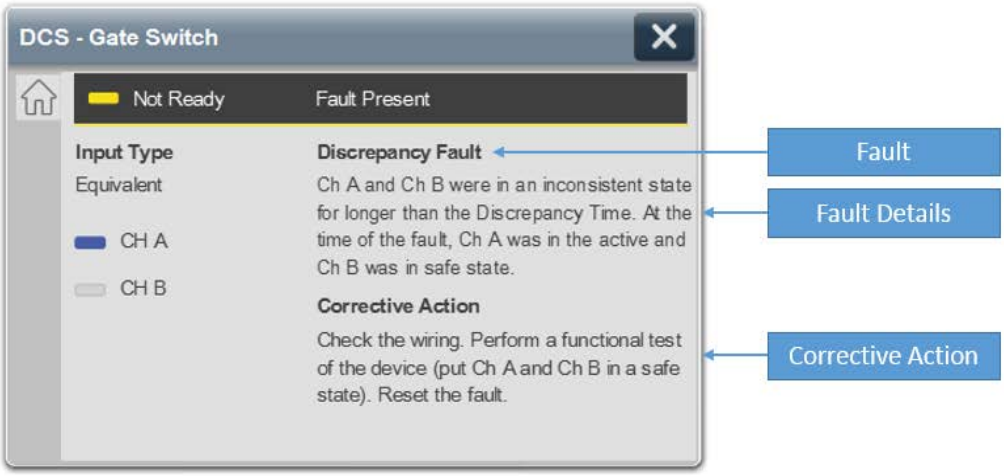
The Home tab shows the input type and input channel A/B statuses. Additionally there is a Ready banner at the top showing the status of the device.



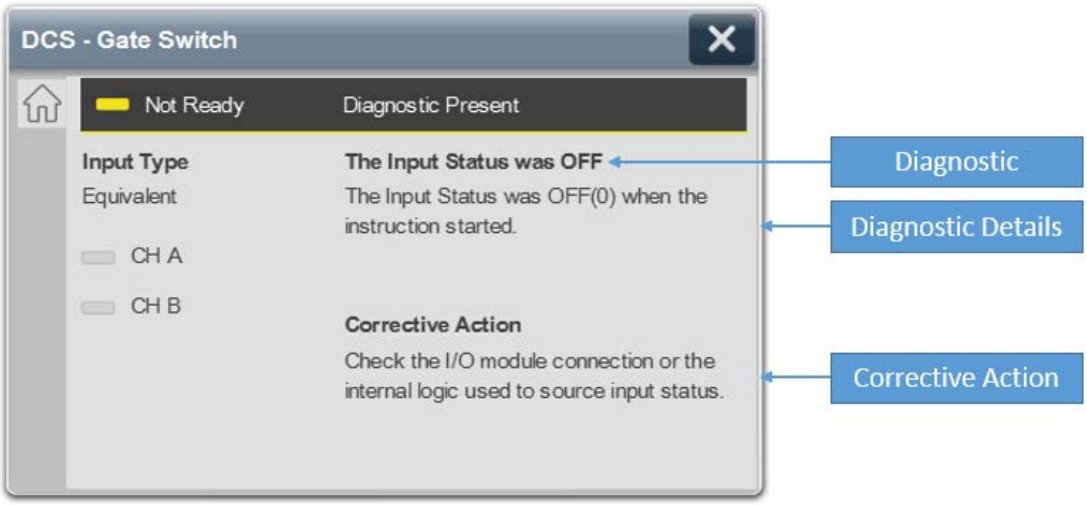
Item	Description
1	Banner- Ready Status
2	Input Type Equivalent = Active High; Inputs are in active state when Channel A and Channel B inputs are 1 Complimentary = Inputs are in active state when Channel A is 1 and Channel B is 0
3	Input Channel A Status OFF (0) = Gray LED ON (1) = Blue LED
4	Input Channel B Status OFF (0) = Gray LED ON (1) = Blue LED

Home tab screen also provides fault and diagnostic information with required corrective actions on the right plane of the faceplate

When there is fault present, banner with black background is displayed on top of home screen with **Not Ready** and **Fault Present** text.



When there is diagnostic present, banner with black background is displayed on top of home screen with **Not Ready** and **Diagnostic Present** text.



Application Code Manager

Safety Device Library objects can be set-up and configured using Studio 5000® Application Code Manager.

Refer to the section Using Application Code Manager for complete details.

Definition Object: raC_Dvc_DCS

This object contains the AOI definition and used as linked library to implement object. This gives flexibility to choose to instantiate only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Implementation Object: raC_LD_Dvc_DCS

Parameter Name	Default Value	Instance Name	Definition	Description
RoutineName	{ObjectName}	{RoutineName}	Routine	Enter Routine name. Routine will be created and Object implement rung(s) inserted. A JSR will be inserted in MainRoutine. If routine name already exists, then object will be inserted into existing routine. By default, parameter is set to Object Name.
TagName	{ObjectName}	{TagName}	Backing Tag	Enter the backing tag of the main AOI. This will serve as the base tag name for other tags in this object that are derived from the base.
TagDescription	{ObjectDescription}	{TagDescription}		Tag Description of the main AOI backing tag

Parameter Name	Default Value	Instance Name	Definition	Description
DisplayName	[ObjectName]	[DisplayName]		Faceplate title/group event list description. Please avoid underscore ("_") at the end.
SafetyFunction	USER DEFINED			This parameter provides a text name for how this instruction is being used. It is for information/documentation purposes only. Options: - USER DEFINED - EMERGENCY STOP - SAFETY GATE - LIGHT CURTAIN - AREA SCANNER - SAFETY MAT - CABLE (ROPE) PULL SWITCH
InputType	Equivalent - Active High			This parameter selects input channel behavior. Equivalent - Active High or Complimentary.
DiscrepencyTime	100			The amount of time that the inputs can be in an inconsistent state before an instruction fault is generated. Range 5-3000ms.
RestartType	Automatic			This input configures Output 1 for either Manual or Automatic Restart.
ColdStartType	Automatic			This parameter specifies the Output 1 behavior when applying controller power or mode change to Run. Automatic or Manual.
ChannelA	ChA		Input Parameter	Enter the input address (data) tag for Channel A from safety digital input module.
ChannelA_InputStatus	ChA_Sts		Input Parameter	Enter the input address (Status) tag for Channel A from safety digital input module.
ChannelB	ChB		Input Parameter	Enter the input address (data) tag for Channel B from safety digital input module.
ChannelB_InputStatus	ChB_Sts		Input Parameter	Enter the input address (Status) tag for Channel B from safety digital input module.
ResetTag	Cmd_Reset		Controller Tag	Enter Tagname for Reset signal. Please avoid underscore ("_") at the end.

Note that if the tag names are manually entered or not linked to input channel tags a red 'X' will be shown beside the parameter. This is acceptable and the program can still be generated.

Linked Libraries

Link Name	Catalog Number	Revision	Solution	Category
raC_Dvc_DCS	raC_Dvc_DCS	1	(RA-LIB) Device	Safety
raC_Opr_SafetyGroupMember	raC_Opr_SafetyGroupMember	1	(RA-LIB) Device	Safety

Configured HMI Content

HMI Content	Instance Name	Description
Launch Button	{ObjectName}_GO_LaunchFP	Global Object configured callout instance

Attachments

Name	Description	File Name	Extraction Path
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols SE	(raC-1-SE) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewSE\Global Objects
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols ME	(raC-1-ME) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewME\Global Objects
V1_raC_Dvc_DCS	Faceplate SE	(raC-1_xx-SE) raC_Dvc_DCS-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
V1_raC_Dvc_DCS	Faceplate ME	(raC-1_xx-ME) raC_Dvc_DCS-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
V1_raC_Dvc_DCS	View Designer	(raC-1_xx-VD) raC_Dvc_Safety.vpd	{ProjectName}\Visualization\ViewDesigner
V1_RM_raC_DvcSafety	Reference Manual	DEVICE-RM500C-EN-P.pdf	{ProjectName}\Documentation
V1_Safety_Images	HMI Image Set	Safety_Images.zip	{ProjectName}\Visualization\Images

DCSTL - Dual Channel Input Stop with Test and Lock (raC_Dvc_DCSTL)

Overview

The Dual Channel Input Stop with Test and Lock device object (raC_Dvc_DCSTL) includes a faceplate which displays status and configuration information of dual-input safety devices whose main function is to stop a machine safely. Common examples include safety gates and slide locks.

The instruction is ready when both safety inputs Channel A and Channel B are in the active state as determined by the Input type operand, and the correct reset actions are carried out. In addition, the faceplate also shows a Locked Feedback and Hazard Stopped signal from a safety device.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following videos for this section:
"Operational_Overview_of_Safety_Device_Object_Faceplates"
"Operational_Overview_of_Safety_Application_Example_in_LogixDesigner"

Functional Description

The Dual Channel Input Stop with Test and Lock pre-configured Device Objects:

- Collect, Process and Deliver Data between Smart Devices and Application Logic
- Provide Device Status & Diagnostics Faceplates for Machine Startup, Commissioning, Operations, and Maintenance
- Include Common HMI Faceplate Components, Device States, and Control Interfaces providing Application Development and Operation Efficiency

All these features provide quick feedback, shorten recovery time, and simplify implementation.

Required Files

Device Objects include Add-On Instructions (AOIs) and HMI faceplates. The revision number (e.g. 1.01) used in filenames can change as new revisions are created.

Controller Files

Add-On Instructions are reusable code objects that contain encapsulated logic that can streamline implementing your system. This lets you create your own instruction set for programming logic as a supplement to the instruction set

provided natively in the ControlLogix® firmware. An Add-On Instruction is defined once in each controller project, and can be instantiated multiple times in your application code as needed.

The Add-On Instruction must be imported into the controller project to be used in the controller configuration. These can be imported as Add-On Instruction files, or as part of the ROUTINE Import or Import Library Objects wizard.

All Add-On Instruction and ROUTINE Import files can be found in the */Studio 5000 Logix Designer® Files - L5X/* folder in the library.

Device/Item	Add-On Instruction	ROUTINE Import
DCSTL	raC_Dvc_DCSTL_1.01_AOI.L5X	raC_Dvc_DCSTL_1.01_ROUTINE.L5X

FactoryTalk® View HMI Files

FactoryTalk® View ME or SE applications require importing the desired device faceplates in addition to all Global Object (ggfx) files and all images located in the */HMI FactoryTalk View Images - png/* folder of the library. FactoryTalk® View ME files are stored in the */HMI - FactoryTalk View ME/* library folder and FactoryTalk® View SE files are stored in the */HMI - FactoryTalk View SE/* library folder.

Device/Item	Type	FactoryTalk® View ME Faceplate	FactoryTalk® View SE Faceplate
DCSTL	Display	(raC-1_01-ME) raC_Dvc_DCSTL-Faceplate.gfx	(raC-1_01-SE) raC_Dvc_DCSTL-Faceplate.gfx
Graphic Symbols	Global Object	(raC-1-ME) Graphic Symbols - Safety Device	(raC-1-SE) Graphic Symbols - Safety Device.ggfx
Toolbox	Global Object	(raC-1-ME) Toolbox - Safety Device.ggfx	(raC-1-SE) Toolbox - Safety Device.ggfx

Studio 5000 View Designer® HMI Files

All Studio 5000 View Designer® Files can be found in the */HMI - ViewDesigner - vpd/* folder of the library.

Device/Item	Studio 5000 View Designer® Faceplate
DCSTL	(raC-1_02-VD) raC_Dvc_Safety.vpd

Studio 5000® Application Code Manager Files

Studio 5000® Application Code Manager (ACM) can be optionally used if it is installed. All devices can be easily registered in the ACM repositories by running the *setup.cmd* file located in the root folder of the library.

Individual HSL4 files are provided as an alternative to running the setup.cmd to allow users to manually register specific implementation objects. Each object has two files - an Asset Control file and a Device file. The Asset Control files include attachments of all required files for that object. The Device files are used to actually add that device into a Studio 5000 project and these reference the Asset Control files.

All Studio 5000® Application Code Manager files can be found in the /*ApplicationCodeManagerLibraries*/ folder of the library. The files included are as follows:

Implementation Object	Asset Control File (.HSL4)	Device File (.HSL4)
DCSTL	(RA-LIB)_Device_Asset-Control_Safety_raC_Dvc_DCSTL_(1.2)	(RA-LIB)_Device_Device_Safety_raC_LD_DCSTL_(1.2)

Operations

The Safety Discrete objects provides only physical operation mode. There is no virtual device mode offered.

Execution

The following table explains the handling of instruction execution conditions.

Condition	Description
EnableIn False (false rung)	Processing for EnableIn False (false rung) is handled the same as if the device were taken out of service by Command. The device outputs are de-energized and the device is shown as Program Out of Service on the HMI. All alarms are cleared.
Powerup (prescan, first scan)	On prescan, any commands that are received before first scan are discarded. The device is de-energized. On first scan, the device is treated as if it were returning from Hand command source: the instruction state is set based on the position feedback that is received from the device. If the feedback is valid for one position, the device is set to that position. If the device does not have position feedback or the position feedback is invalid, the device is set to the 'unknown/powerup' state. The command source is set to its default, either Operator or Program (unlocked).
Postscan	No SFC Postscan logic is provided.

Add-On Instruction I/O Data

InOut Data

InOut	Function / Description	DataType
Inp_DCSTL	Input Interface - DCSTL Instruction	DCI_STOP_TEST_LOCK
Out_Ctrl_Inf	Information Interface	raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	Status Interface	raC_UDT_Itf_SafetyDevice_Sts

Input Data

Input	Function/Description	DataType
EnableIn	Enable Input - System Defined Parameter	BOOL
Inp_Suspended	Suspend device operation	BOOL

Output Data

Output	Function/Descritpion	DataType
Sts_DeviceSuspended	Device is suspended	BOOL
Sts_SafetyDemand	Safety demand is present	BOOL
Sts_ResetRequired	Reset is required	BOOL
Sts_DiagnosticPresent	Diagnostic is present	BOOL
Sts_FaultPresent	Fault is present	BOOL
Sts_Ready	Device is ready	BOOL
Sts_ERR	Instruction Error Code	DINT
Sts_eReady	0: Ready 1: Safety Demand 2: Reset Required 3: Suspended	SINT
raC_Dvc_DCSTL	Unique Parameter Name for auto - discovery	BOOL
EnableOut	Enable Output - System Defined Parameter	BOOL

Data Types

The following Safety Common Control Interface tags are the primary device program tags to read and write to when interfacing to safety devices. The value of using these tags in your specific application code is that you may use a number of different safety devices such as DCS, DCSTL, etc without having to update your application device interface tags. The primary function of these interfaces within the Safety Device Library is to pass individual instruction information to the Safety Group Monitor instruction.

Refer to the Interfaces section for detailed information on interfaces.

raC_UDT_Itf_SafetyDevice_Inf

Member	Description	Data Type
Type	Provides device object type information e.g. raC_Dvc_SafetyGroupMonitor, raC_Dvc_DCS, raC_Dvc_DCSTL, etc.	STR0032
DisplayName	Provides object display name	STR0024
eClass	Provide instruction type information eClass ID: Instruction Type 0: DCS 1: DCSTL 2: DCSTM 3: CROUT 5: SafetyGroupMonitor 6: TSAM 7: TSSM 8: FSBM 9: THRSe 10: SMAT 11:SS2 12: SS1 13:SFX 14:STO	DINT

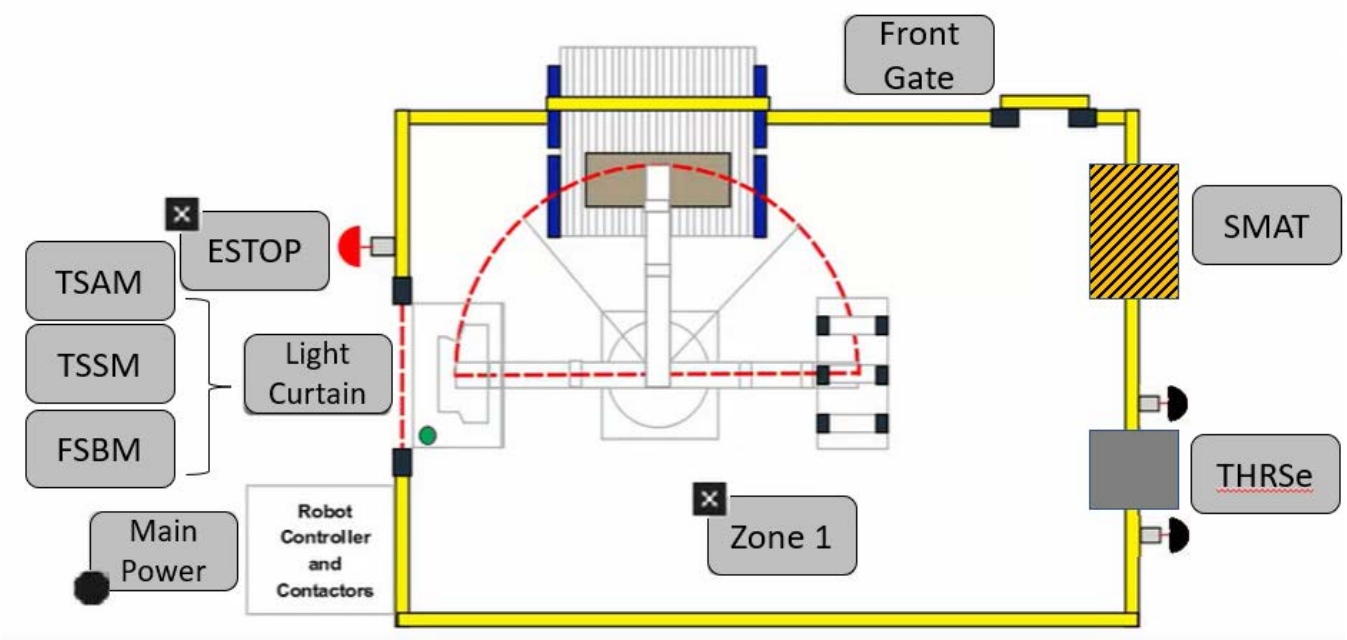
raC_UDT_Itf_SafetyDevice_Sts

This is the Safety Device Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data. The table below shows member names, descriptions, and tag data types.

Input	Description	Data Type
bSts	Bit overlay (Visible) covering all subsequent Boolean members. (Unused)	DINT
DeviceSuspended	Device suspended status (Inp_Suspend = 1)	BOOL
SafetyDemand	Safety demand status ON (1): Safety demand is present on the device object OFF (0): No safety demand	BOOL
ResetRequired	Reset Required status ON (1): Device requires reset signal OFF (0): No reset required	BOOL
DiagnosticPresent	Diagnostic status ON (1): A diagnostic is present in the device object OFF (0): Device is operating normally	BOOL
FaultPresent	Fault Status ON (1): A fault is present in the device object OFF (0): Device is operating normally	BOOL
Muted	Device Muted status (Available for DCSTM) ON (1): Muted status is present on one or more safety instructions within a specific safety group OFF (0): None of the instructions within group is muted	BOOL

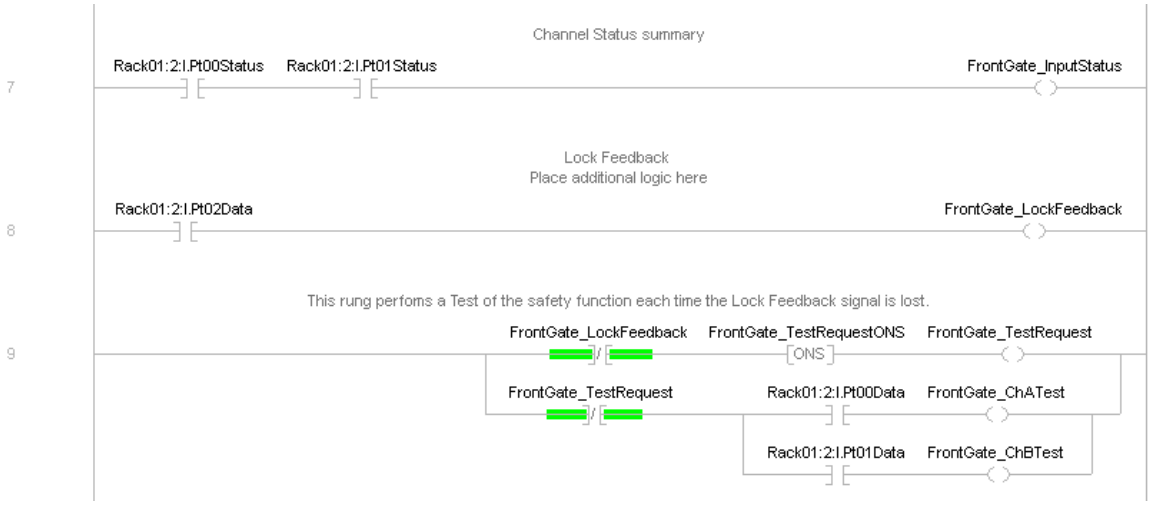
Programming Example

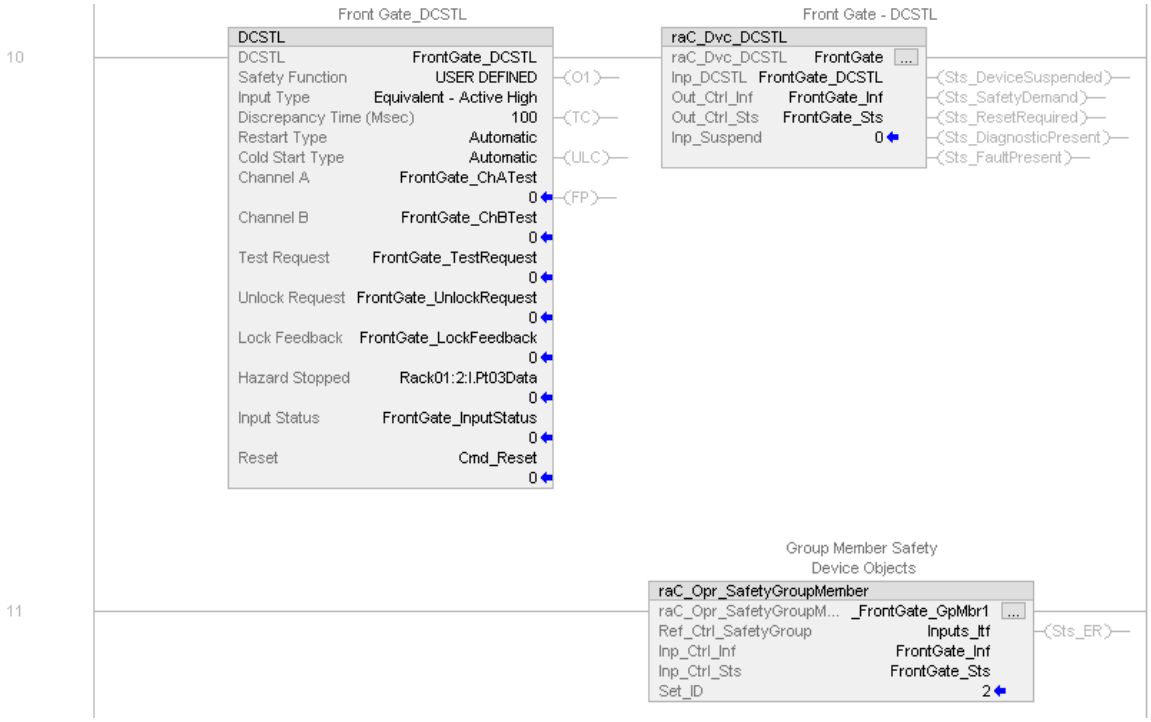
Fully configured device routine is provided below for reference. This example is taken from the *Safety Program - Inputs* routine in the *SafetytApplication.ACD* example application provided in the library.



This example uses the DCSTL instruction for a gate function shown as *Front Gate* in the safe machine diagram. Channel A and Channel B are tied to safety class discrete inputs. Rung descriptions are as follows:

- Rung 7: Check input statuses
- Rung 8: Check lock feedback status
- Rung 9: Test function performed on lock feedback status
- Rung 10: Configure DCSTL and raC_Dvc_DCSTL instructions
- Rung 11: raC_Opr_SafetyGroupMember (optional) assigns FrontGate DCSTL to Inputs Safety Group at ID #2.





Note that this programming example is the same code that is imported when either importing the supplied ROUTINE .L5X files or when using Application Code Manager or the Studio 5000 Import Library Objects wizard plug-in.

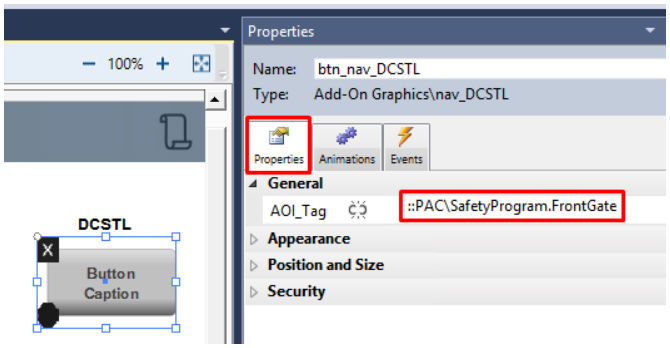
Graphic Symbols

Graphic Symbols are used as launch buttons within HMI applications to open up faceplate displays. See Basic Launch Button Attributes section for details on configuration and indicators

FactoryTalk® View ME/SE Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Global Object Parameter Values
btn_nav_safety		Faceplate navigation button with string tag label. This launch button graphic object allows the user to navigate to the device object faceplate. The text on the button face is set to the tag's .Inf_DisplayName.	#101: AOI Instance (e.g. {::[PAC]Program::SafetyProgram._InstanceName})

Studio 5000 View Designer® Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Property Configuration
Launch		The supplied launch button in View Designer is used to navigate to the faceplate in a user application.	

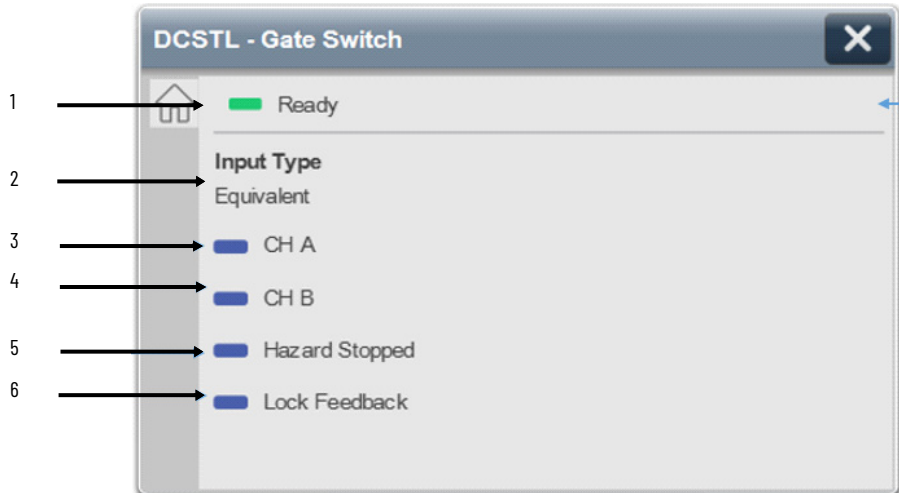
Faceplates

There are basic faceplate attributes that are common across all instructions. See Basic Faceplate Attributes on page 27.

Home

The Home tab is the main tab of the faceplate. It provides the status of the safety input device along with configuration information.

The Home tab shows the input type and input channel A/B statuses. Additionally there is a Ready banner at the top showing the status of the device.

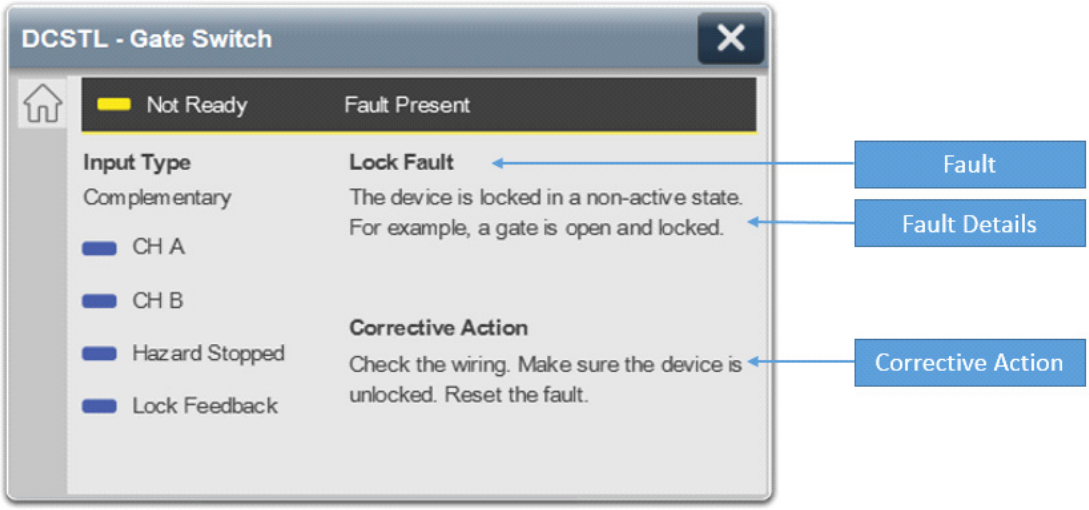


Item	Description
1	Banner- Ready Status
2	Input Type Equivalent = Active High; Inputs are in active state when Channel A and Channel B inputs are 1 Complimentary = Inputs are in active state when Channel A is 1 and Channel B is 0
3	Input Channel A Status OFF (0) = Gray LED ON (1) = Blue LED

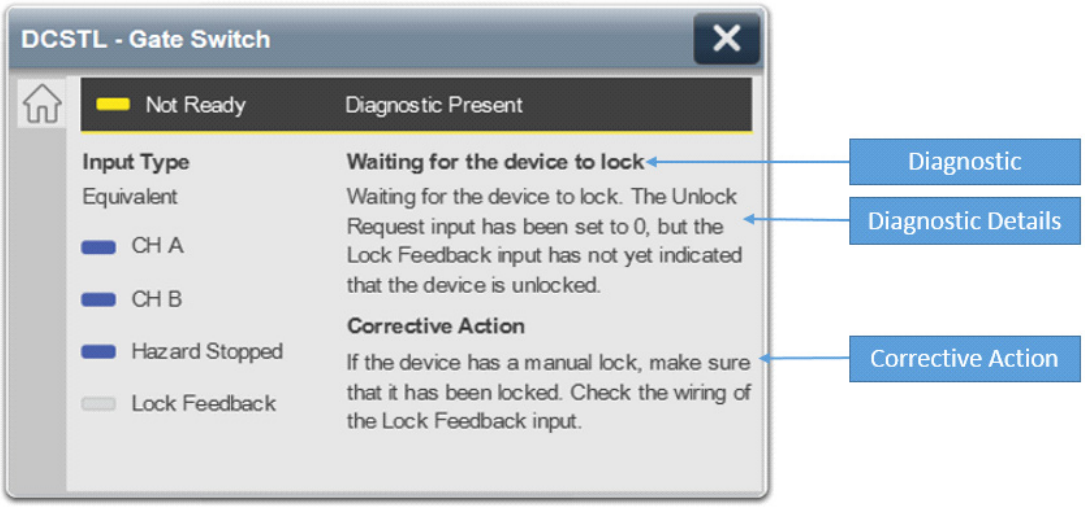
Item	Description
4	Input Channel B Status OFF (0) = Gray LED ON (1) = Blue LED
5	Hazard Stopped Status. This input must be ON (1) in order to issue an unlock command (energize the Unlock Command output). OFF (0) = Gray LED. The unlock command output cannot be energized. ON (1) = Blue LED. The unlock command output can be energized.
6	Lock Feedback Status. This input must be ON (1) in order to energize Output 1. OFF (0) = Gray LED. The safety monitoring device is currently not locked. ON (1) = Blue LED. The safety monitoring device is currently locked.

Home tab screen also provides fault and diagnostic information with required corrective actions on the right plane of the faceplate

When there is fault present, banner with black background is displayed on top of home screen with **Not Ready** and **Fault Present** text.



When there is diagnostic present, banner with black background is displayed on top of home screen with **Not Ready** and **Diagnostic Present** text.



Application Code Manager

Safety Device Library objects can be set-up and configured using Studio 5000® Application Code Manager.

Refer to the section Using Application Code Manager for complete details.

Definition Object: raC_Dvc_DCSTL

This object contains the AOI definition and used as linked library to implement object. This gives flexibility to choose to instantiate only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Implementation Object: raC_LD_Dvc_DCSTL

Parameter Name	Default Value	Instance Name	Definition	Description
RoutineName	{ObjectName}	{RoutineName}	Routine	Enter Routine name. Routine will be created and Object implement rung(s) inserted. A JSR will be inserted in MainRoutine. If routine name already exists, then object will be inserted into existing routine. By default, parameter is set to Object Name.
TagName	{ObjectName}	{TagName}	Backing Tag	Enter the backing tag of the main AOI. This will serve as the base tag name for other tags in this object that are derived from the base.
TagDescription	{ObjectDescription}	{TagDescription}		Tag Description of the main AOI backing tag
DisplayName	[ObjectName]	[DisplayName]		Faceplate title/group event list description. Please avoid underscore ("_") at the end.
ChanIsLockFB	False			DCSTL Channel Inputs provide lock feedback.
SafetyFunction	USER DEFINED			This parameter provides a text name for how this instruction is being used. It is for information/documentation purposes only. Options: - USER DEFINED - SAFETY GATE - SLIDE LOCK
InputType	Equivalent - Active High			This parameter selects input channel behavior. Equivalent - Active High or Complimentary.
DiscrepencyTime	100			The amount of time that the inputs can be in an inconsistent state before an instruction fault is generated. Range 5-3000ms.
RestartType	Automatic			This input configures Output 1 for either Manual or Automatic Restart.

Parameter Name	Default Value	Instance Name	Definition	Description
ColdStartType	Automatic			This parameter specifies the Output 1 behavior when applying controller power or mode change to Run. Automatic or Manual.
ChannelA	ChA		Input Parameter	Enter the input address (data) tag for Channel A from safety digital input module.
ChannelA_InputStatus	ChA_Sts		Input Parameter	Enter the input address (Status) tag for Channel A from safety digital input module.
ChannelB	ChB		Input Parameter	Enter the input address (data) tag for Channel B from safety digital input module.
ChannelB_InputStatus	ChB_Sts		Input Parameter	Enter the input address (Status) tag for Channel B from safety digital input module.
TestRequest	TestRequest		Controller Tag	This parameter forces a functional test to occur. Please avoid underscore ("_") at the end.
UnlockRequest	UnlockRequest		Controller Tag	This input is used to request a lock and unlock of electromechanical locking devices. Please avoid underscore ("_") at the end.
LockFeedback	LockFeedback		Input Parameter	Enter the input address tag for Lock Feedback from safety digital input module.
HazardStopped	HazardStopped		Input Parameter	Enter the input address tag of Hazard Stop from safety digital input module.
ResetTag	Cmd_Reset		Controller Tag	Enter Tagname for Reset signal. Please avoid underscore ("_") at the end.

Note that if the tag names are manually entered or not linked to input channel tags a red 'X' will be shown beside the parameter. This is acceptable and the program can still be generated.

Linked Libraries

Link Name	Catalog Number	Revision	Solution	Category
raC_Dvc_DCSTL	raC_Dvc_DCSTL	1	(RA-LIB) Device	Safety
raC_Opr_SafetyGroupMember	raC_Opr_SafetyGroupMember	1	(RA-LIB) Device	Safety

Configured HMI Content

HMI Content	Instance Name	Description
Launch Button	{ObjectName}_GO_LaunchFP	Global Object configured callout instance

Attachments

Name	Description	File Name	Extraction Path
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols SE	(raC-1-SE) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewSE\Global Objects
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols ME	(raC-1-ME) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewME\Global Objects
V1_raC_Dvc_DCSTL	Faceplate SE	(raC-1_xx-SE) raC_Dvc_DCSTL-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
V1_raC_Dvc_DCSTL	Faceplate ME	(raC-1_xx-ME) raC_Dvc_DCSTL-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
V1_raC_Dvc_DCSTL	View Designer	(raC-1_xx-VD) raC_Dvc_Safety.vpd	{ProjectName}\Visualization\ViewDesigner
V1_RM_raC_DvcSafety	Reference Manual	DEVICE-RM500C-EN-P.pdf	{ProjectName}\Documentation
V1_Safety_Images	HMI Image Set	Safety_Images.zip	{ProjectName}\Visualization\Images

DCSTM - Dual Channel Input Stop with Test and Mute (raC_Dvc_DCSTM)

Overview

The Dual Channel Input Stop with Test and Mute device object (raC_Dvc_DCSTM) includes a faceplate which displays status and configuration information of dual-input safety devices whose main function is to stop a machine safely. Common examples include light curtains, area scanners and safety mats.

The instruction is ready when both safety inputs Channel A and Channel B are in the active state as determined by the Input type operand, and the correct reset actions are carried out. In addition, the faceplate also shows a mute function used to mute the safety device.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following videos for this section:
"Operational_Overview_of_Safety_Device_Object_Faceplates"
"Operational_Overview_of_Safety_Application_Example_in_LogixDesigner"

Functional Description

The Dual Channel Input Stop with Test and Mute pre-configured Device Objects:

- Collect, Process and Deliver Data between Smart Devices and Application Logic
- Provide Device Status & Diagnostics Faceplates for Machine Startup, Commissioning, Operations, and Maintenance
- Include Common HMI Faceplate Components, Device States, and Control Interfaces providing Application Development and Operation Efficiency

All these features provide quick feedback, shorten recovery time, and simplify implementation.

Required Files

Device Objects include Add-On Instructions (AOIs) and HMI faceplates. The revision number (e.g. 1.01) used in filenames can change as new revisions are created.

Controller Files

Add-On Instructions are reusable code objects that contain encapsulated logic that can streamline implementing your system. This lets you create your own instruction set for programming logic as a supplement to the instruction set

provided natively in the ControlLogix® firmware. An Add-On Instruction is defined once in each controller project, and can be instantiated multiple times in your application code as needed.

The Add-On Instruction must be imported into the controller project to be used in the controller configuration. These can be imported as Add-On Instruction files, or as part of the ROUTINE Import or Import Library Objects wizard.

All Add-On Instruction and ROUTINE Import files can be found in the */Studio 5000 Logix Designer® Files - L5X/* folder in the library.

Device/Item	Add-On Instruction	ROUTINE Import
DCSTM	raC_Dvc_DCSTM_1.01_AOI.L5X	raC_Dvc_DCSTM_1.01_ROUTINE.L5X

FactoryTalk® View HMI Files

FactoryTalk® View ME or SE applications require importing the desired device faceplates in addition to all Global Object (ggfx) files and all images located in the */HMI FactoryTalk View Images - png/* folder of the library. FactoryTalk® View ME files are stored in the */HMI - FactoryTalk View ME/* library folder and FactoryTalk® View SE files are stored in the */HMI - FactoryTalk View SE/* library folder.

Device/Item	Type	FactoryTalk® View ME Faceplate	FactoryTalk® View SE Faceplate
DCSTM	Display	(raC-1_01-ME) raC_Dvc_DCSTM-Faceplate.gfx	(raC-1_01-SE) raC_Dvc_DCSTM-Faceplate.gfx
Graphic Symbols	Global Object	(raC-1-ME) Graphic Symbols - Safety Device	(raC-1-SE) Graphic Symbols - Safety Device.ggfx
Toolbox	Global Object	(raC-1-ME) Toolbox - Safety Device.ggfx	(raC-1-SE) Toolbox - Safety Device.ggfx

Studio 5000 View Designer® HMI Files

All Studio 5000 View Designer® Files can be found in the */HMI - ViewDesigner - vpd/* folder of the library.

Device/Item	Studio 5000 View Designer® Faceplate
DCSTM	(raC-1_02-VD) raC_Dvc_Safety.vpd

Studio 5000® Application Code Manager Files

Studio 5000® Application Code Manager (ACM) can be optionally used if it is installed. All devices can be easily registered in the ACM repositories by running the *setup.cmd* file located in the root folder of the library.

Individual HSL4 files are provided as an alternative to running the setup.cmd to allow users to manually register specific implementation objects. Each object has two files - an Asset Control file and a Device file. The Asset Control files include attachments of all required files for that object. The Device files are used to actually add that device into a Studio 5000 project and these reference the Asset Control files.

All Studio 5000® Application Code Manager files can be found in the /*ApplicationCodeManagerLibraries*/ folder of the library. The files included are as follows:

Implementation Object	Asset Control File (.HSL4)	Device File (.HSL4)
DCSTM	(RA-LIB)_Device_Asset-Control_Safety_raC_Dvc_DCSTM_(1.2)	(RA-LIB)_Device_Device_Safety_raC_LD_DCSTM_(1.2)

Operations

The Safety Discrete objects provides only physical operation mode. There is no virtual device mode offered.

Execution

The following table explains the handling of instruction execution conditions.

Condition	Description
EnableIn False (false rung)	Processing for EnableIn False (false rung) is handled the same as if the device were taken out of service by Command. The device outputs are de-energized and the device is shown as Program Out of Service on the HMI. All alarms are cleared.
Powerup (prescan, first scan)	On prescan, any commands that are received before first scan are discarded. The device is de-energized. On first scan, the device is treated as if it were returning from Hand command source: the instruction state is set based on the position feedback that is received from the device. If the feedback is valid for one position, the device is set to that position. If the device does not have position feedback or the position feedback is invalid, the device is set to the 'unknown/powerup' state. The command source is set to its default, either Operator or Program (unlocked).
Postscan	No SFC Postscan logic is provided.

Add-On Instruction I/O Data

InOut Data

InOut	Function / Description	DataType
Inp_DCSTM	Input Interface - DCSTM Instruction	DCI_STOP_TEST_MUTE
Out_Ctrl_Inf	Information Interface	raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	Status Interface	raC_UDT_Itf_SafetyDevice_Sts

Input Data

Input	Function/Description	DataType
EnableIn	Enable Input - System Defined Parameter	BOOL
Inp_Suspended	Suspend device operation	BOOL

Output Data

Output	Function/Descritpion	DataType
Sts_DeviceSuspended	Device is suspended	BOOL
Sts_SafetyDemand	Safety demand is present	BOOL
Sts_ResetRequired	Reset is required	BOOL
Sts_DiagnosticPresent	Diagnostic is present	BOOL
Sts_FaultPresent	Fault is present	BOOL
Sts_Ready	Device is ready	BOOL
Sts_Muted	Safety Device Is Muted	BOOL
Sts_ERR	Instruction Error Code	DINT
Sts_eReady	0: Ready 1: Safety Demand 2: Reset Required 3: Suspended	SINT
raC_Dvc_DCSTM	Unique Parameter Name for auto - discovery	BOOL
EnableOut	Enable Output - System Defined Parameter	BOOL

Data Types

The following Safety Common Control Interface tags are the primary device program tags to read and write to when interfacing to safety devices. The value of using these tags in your specific application code is that you may use a number of different safety devices such as DCS, DCSTM, etc without having to update your application device interface tags. The primary function of these interfaces within the Safety Device Library is to pass individual instruction information to the Safety Group Monitor instruction.

Refer to the Interfaces section for detailed information on interfaces.

raC_UDT_Itf_SafetyDevice_Inf

Member	Description	Data Type
Type	Provides device object type information e.g. raC_Dvc_SafetyGroupMonitor, raC_Dvc_DCS, raC_Dvc_DCSTM, etc.	STR0032
DisplayName	Provides object display name	STR0024
eClass	Provide instruction type information eClass ID: Instruction Type 0: DCS 1: DCSTL 2: DCSTM 3: CROUT 5: SafetyGroupMonitor 6: TSAM 7: TSSM 8: FSBM 9: THRSe 10: SMAT 11:SS2 12: SS1 13:SFX 14:STO	DINT

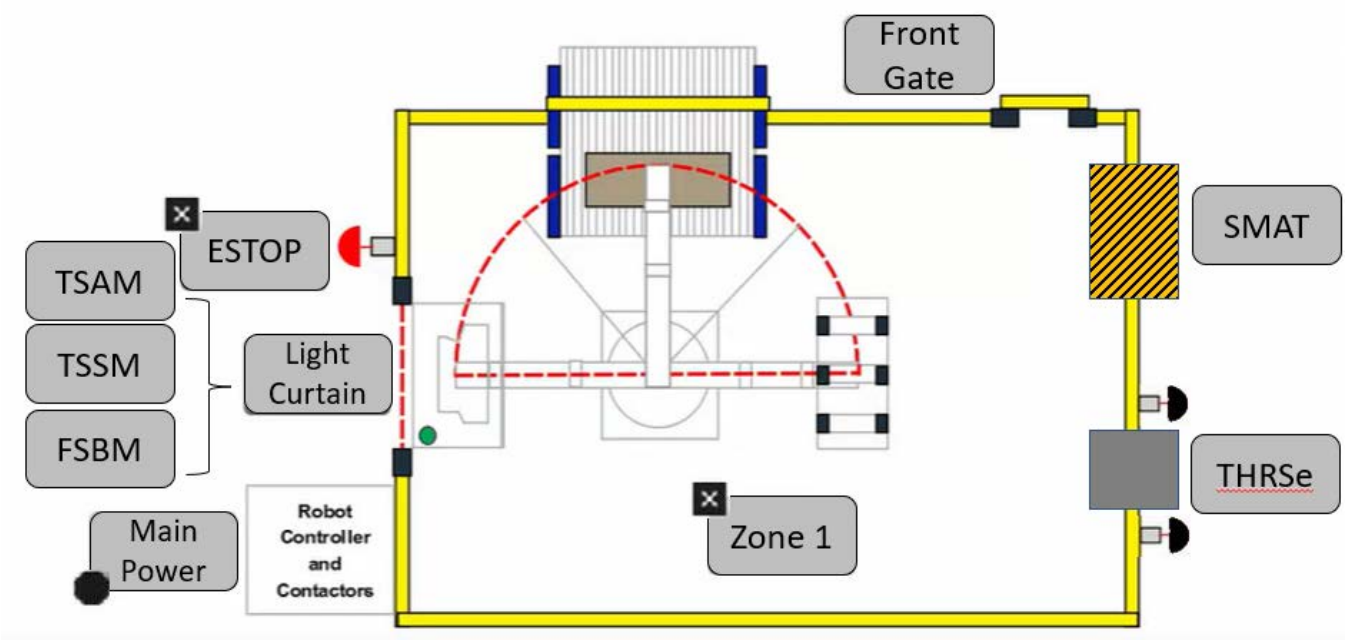
raC_UDT_Itf_SafetyDevice_Sts

This is the Safety Device Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data. The table below shows member names, descriptions, and tag data types.

Input	Description	Data Type
bSts	Bit overlay (Visible) covering all subsequent Boolean members. (Unused)	DINT
DeviceSuspended	Device suspended status (Inp_Suspend = 1)	BOOL
SafetyDemand	Safety demand status ON (1): Safety demand is present on the device object OFF (0): No safety demand	BOOL
ResetRequired	Reset Required status ON (1): Device requires reset signal OFF (0): No reset required	BOOL
DiagnosticPresent	Diagnostic status ON (1): A diagnostic is present in the device object OFF (0): Device is operating normally	BOOL
FaultPresent	Fault Status ON (1): A fault is present in the device object OFF (0): Device is operating normally	BOOL
Muted	Device Muted status (Available for DCSTM) ON (1): Muted status is present on one or more safety instructions within a specific safety group OFF (0): None of the instructions within group is muted	BOOL

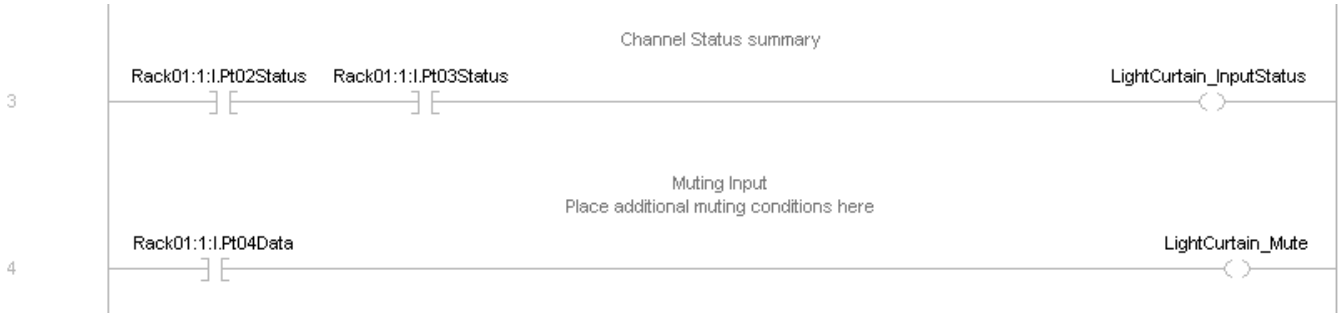
Programming Example

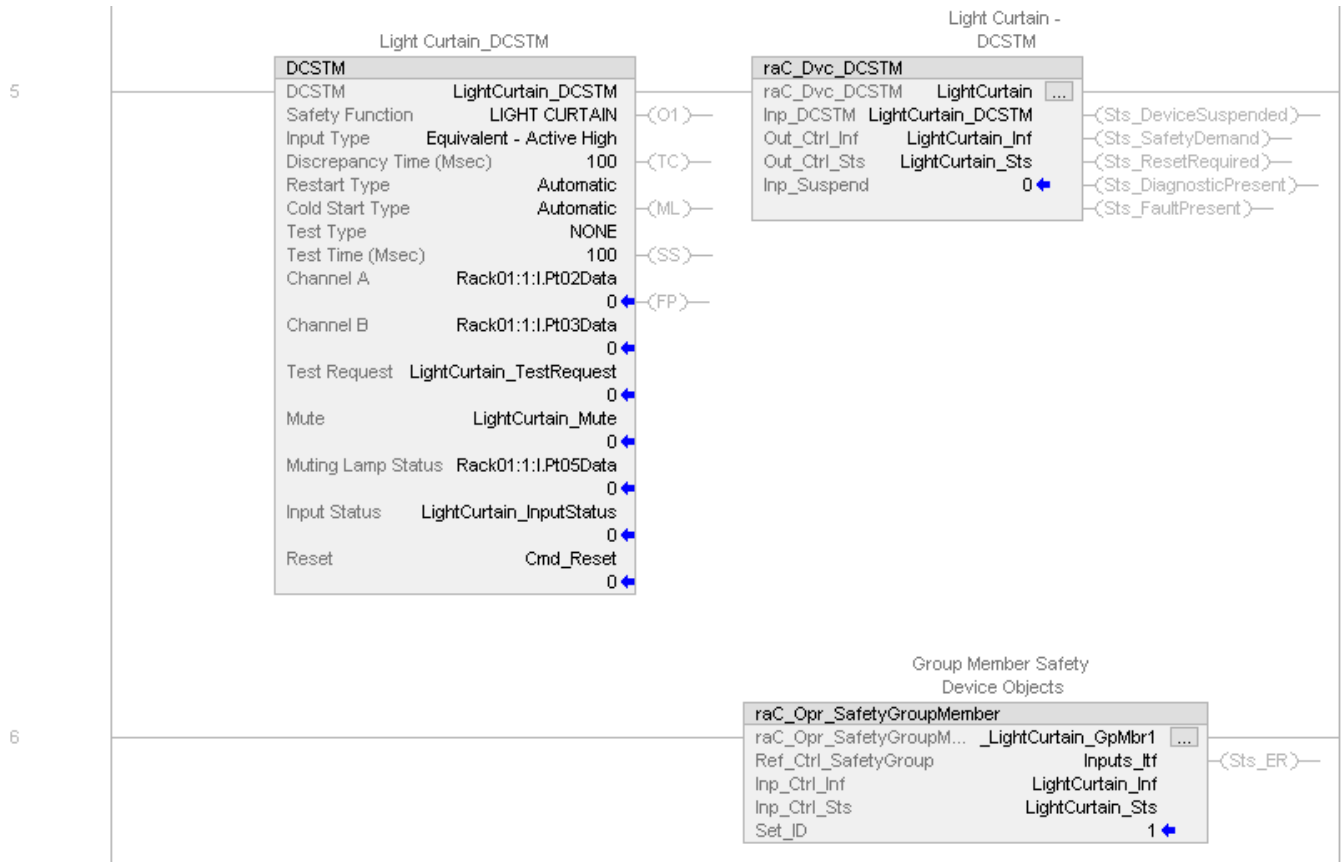
Fully configured device routine is provided below for reference. This example is taken from the *Safety Program - Inputs* routine in the *SafetytApplication.ACD* example application provided in the library.



This example uses the DCSTM instruction for a light curtain function shown as *Light Curtain* in the safe machine diagram. Channel A and Channel B are tied to safety class discrete inputs. Rung descriptions are as follows:

- Rung 3: Check input statuses
- Rung 4: Check light curtain muting input status
- Rung 5: Configure DCSTM and raC_Dvc_DCSTM instructions
- Rung 6: raC_Opr_SafetyGroupMember (optional) assigns LightCurtain DCSTM to Inputs Safety Group at ID #1.





Note that this programming example is the same code that is imported when either importing the supplied ROUTINE .L5X files or when using Application Code Manager or the Studio 5000 Import Library Objects wizard plug-in.

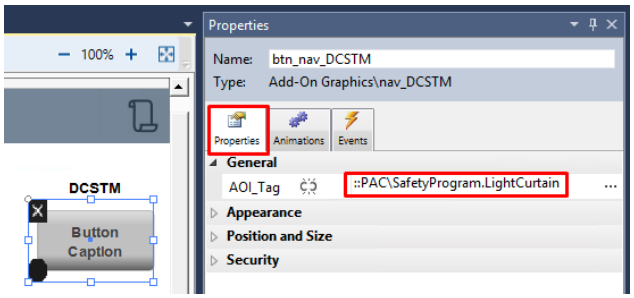
Graphic Symbols

Graphic Symbols are used as launch buttons within HMI applications to open up faceplate displays. See Basic Launch Button Attributes section for details on configuration and indicators

FactoryTalk® View ME/SE Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Global Object Parameter Values
btn_nav_safety		Faceplate navigation button with string tag label. This launch button graphic object allows the user to navigate to the device object faceplate. The text on the button face is set to the tag's .Inf_DisplayName.	#101: AOI Instance (e.g. {::[PAC]Program::SafetyProgram._InstanceName})

Studio 5000 View Designer® Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Property Configuration
Launch		The supplied launch button in View Designer is used to navigate to the faceplate in a user application.	

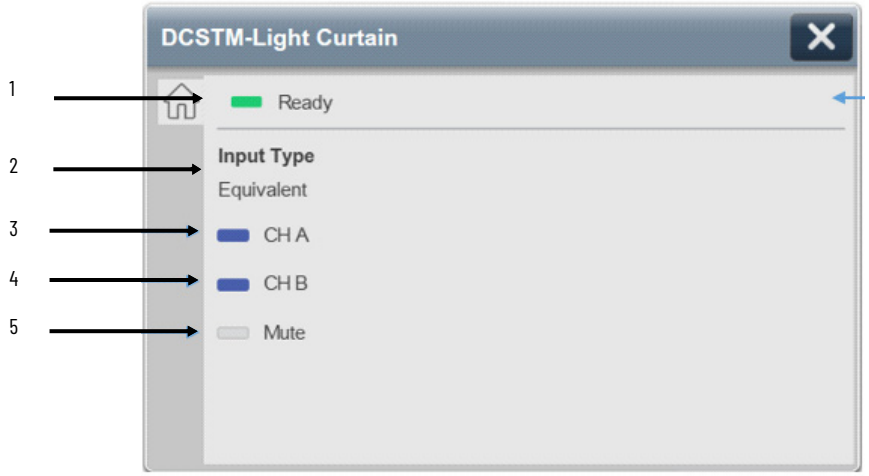
Faceplates

There are basic faceplate attributes that are common across all instructions. See Basic Faceplate Attributes on page 27.

Home

The Home tab is the main tab of the faceplate. It provides the status of the safety input device along with configuration information.

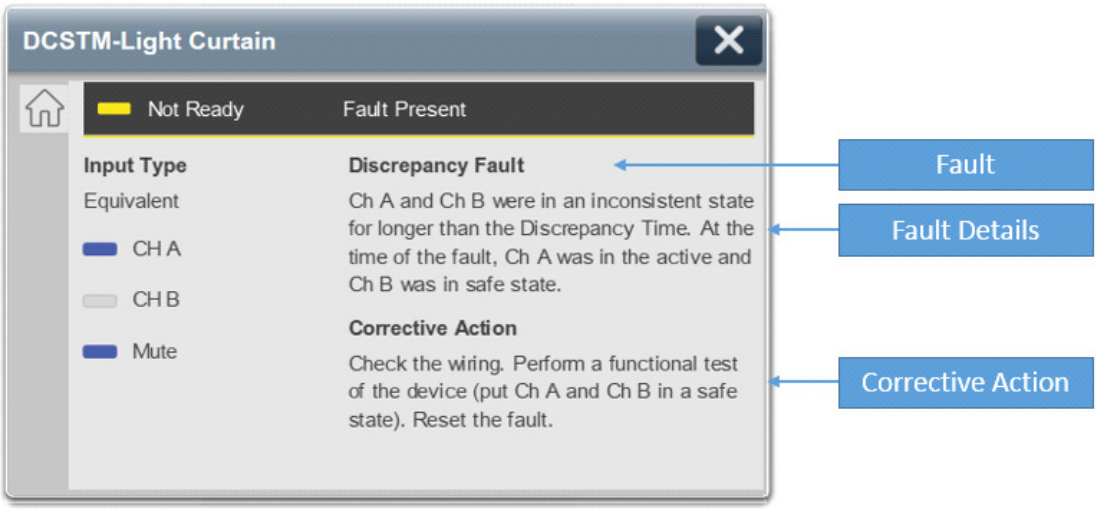
The Home tab shows the input type and input channel A/B statuses and Mute status. Additionally there is a Ready banner at the top showing the status of the device.



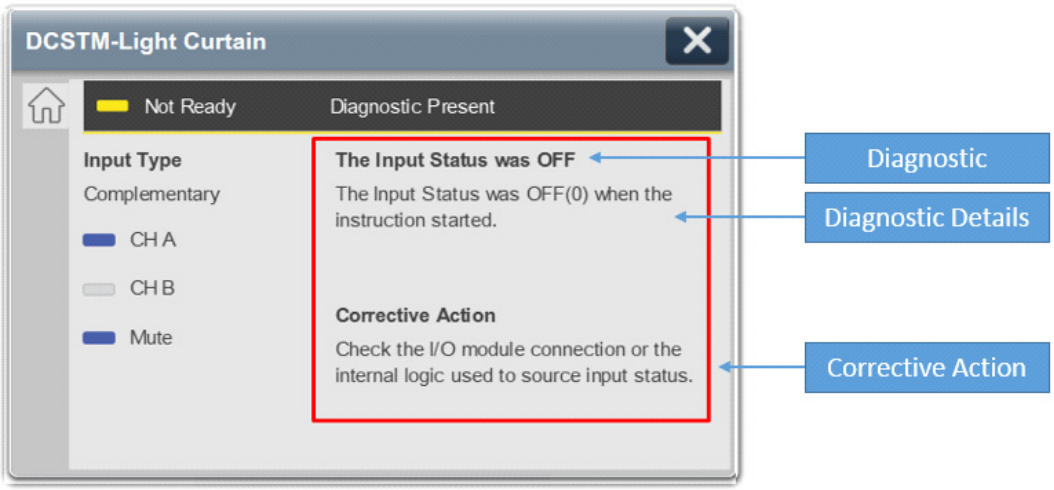
Item	Description
1	Banner- Ready Status
2	Input Type Equivalent = Active High; Inputs are in active state when Channel A and Channel B inputs are 1 Complimentary = Inputs are in active state when Channel A is 1 and Channel B is 0
3	Input Channel A Status OFF (0) = Gray LED ON (1) = Blue LED
4	Input Channel B Status OFF (0) = Gray LED ON (1) = Blue LED
5	Mute Status. The muting lamp output is energized and Output 1 is not de-energeized when the safety device is tripped (Channel A or Channel B eneteres the safe state). When muting is enabled a safety device sensing field can be broken where Channel A and Channel B can go to the safet sate without de-energizing Output 1. OFF (0) = Gray LED. Muting is not activated. ON (1) = Blue LED. Muting is activated.

Home tab screen also provides fault and diagnostic information with required corrective actions on the right plane of the faceplate

When there is fault present, banner with black background is displayed on top of home screen with **Not Ready** and **Fault Present** text.



When there is diagnostic present, banner with black background is displayed on top of home screen with **Not Ready** and **Diagnostic Present** text.



Application Code Manager

Safety Device Library objects can be set-up and configured using Studio 5000® Application Code Manager.

Refer to the section Using Application Code Manager for complete details.

Definition Object: raC_Dvc_DCSTM

This object contains the AOI definition and used as linked library to implement object. This gives flexibility to choose to instantiate only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Implementation Object: raC_LD_Dvc_DCSTM

Parameter Name	Default Value	Instance Name	Definition	Description
RoutineName	{ObjectName}	{RoutineName}	Routine	Enter Routine name. Routine will be created and Object implement rung(s) inserted. A JSR will be inserted in MainRoutine. If routine name already exists, then object will be inserted into existing routine. By default, parameter is set to Object Name.
TagName	{ObjectName}	{TagName}	Backing Tag	Enter the backing tag of the main AOI. This will serve as the base tag name for other tags in this object that are derived from the base.
TagDescription	{ObjectDescription}	{TagDescription}		Tag Description of the main AOI backing tag

Parameter Name	Default Value	Instance Name	Definition	Description
DisplayName	[ObjectName]	[DisplayName]		Faceplate title/group event list description. Please avoid underscore ("_") at the end.
SafetyFunction	USER DEFINED			This parameter provides a text name for how this instruction is being used. It is for information/documentation purposes only. Options: - USER DEFINED - LIGHT CURTAIN - AREA SACANNER - SAFETY MAT
InputType	Equivalent - Active High			This parameter selects input channel behavior. Equivalent - Active High or Complimentary.
DiscrepencyTime	100			The amount of time that the inputs can be in an inconsistent state before an instruction fault is generated. Range 5-3000ms.
RestartType	Automatic			This input configures Output 1 for either Manual or Automatic Restart.
ColdStartType	Automatic			This parameter specifies the Output 1 behavior when applying controller power or mode change to Run. Automatic or Manual.
TestType	None			The operand defined which type of test occurs when Test Request transitions from ON (1) to OFF (0). - NONE - MANUAL - ACTIVE
TestTime	100			The maximum time for an active test to complete. If the test does not complete within this time, a fault is generated. The valid range is 5...1000 ms.
ChannelA	ChA		Input Parameter	Enter the input address (data) tag for Channel A from safety digital input module.
ChannelA_InputStatus	ChA_Sts		Input Parameter	Enter the input address (Status) tag for Channel A from safety digital input module.
ChannelB	ChB		Input Parameter	Enter the input address (data) tag for Channel B from safety digital input module.
ChannelB_InputStatus	ChB_Sts		Input Parameter	Enter the input address (Status) tag for Channel B from safety digital input module.
Mute_Input	MuteInput		Input Parameter	Enter the input address tag used to Mute the safety device from safety digital input module.
MuteLampStatus	MuteInput_Sts		Input Parameter	Enter the input address tag used to Mute the safety device from safety digital input module.
ResetTag	Cmd_Reset		Controller Tag	Enter Tagname for Reset signal. Please avoid underscore ("_") at the end.

Note that if the tag names are manually entered or not linked to input channel tags a red 'X' will be shown beside the parameter. This is acceptable and the program can still be generated.

Linked Libraries

Link Name	Catalog Number	Revision	Solution	Category
raC_Dvc_DCSTM	raC_Dvc_DCSTM	1	(RA-LIB) Device	Safety
raC_Opr_SafetyGroupMember	raC_Opr_SafetyGroupMember	1	(RA-LIB) Device	Safety

Configured HMI Content

HMI Content	Instance Name	Description
Launch Button	{ObjectName}_GO_LaunchFP	Global Object configured callout instance

Attachments

Name	Description	File Name	Extraction Path
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols SE	(raC-1-SE) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewSE\Global Objects
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols ME	(raC-1-ME) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewME\Global Objects
V1_raC_Dvc_DCSTM	Faceplate SE	(raC-1_xx-SE) raC_Dvc_DCSTM-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
V1_raC_Dvc_DCSTM	Faceplate ME	(raC-1_xx-ME) raC_Dvc_DCSTM-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
V1_raC_Dvc_DCSTM	View Designer	(raC-1_xx-VD) raC_Dvc_Safety.vpd	{ProjectName}\Visualization\ViewDesigner
V1_RM_raC_DvcSafety	Reference Manual	DEVICE-RM500C-EN-P.pdf	{ProjectName}\Documentation
V1_Safety_Images	HMI Image Set	Safety_Images.zip	{ProjectName}\Visualization\Images

CROUT - Configurable Redundant Output (raC_Dvc_CROUT)

Overview

The Configurable Redundant Output device object (raC_Dvc_CROUT) includes a faceplate which displays status of redundant safety outputs. The reaction time for output feedback is configurable and the instruction supports both positive and negative feedback signals.

The instruction is ready when redundant outputs are in the active state. The instruction is not ready when one or more of the following events occur:

- A feedback fault occurs
- Input status or output status inputs become invalid (OFF = 0)
- The actuate input turns OFF (0)

Functional Description

The Configurable Redundant Output pre-configured Device Objects:

- Collect, Process and Deliver Data between Smart Devices and Application Logic
- Provide Device Status & Diagnostics Faceplates for Machine Startup, Commissioning, Operations, and Maintenance
- Include Common HMI Faceplate Components, Device States, and Control Interfaces providing Application Development and Operation Efficiency

All these features provide quick feedback, shorten recovery time, and simplify implementation.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following videos for this section:
"Operational_Overview_of_Safety_Device_Object_Faceplates"
"Operational_Overview_of_Safety_Application_Example_in_LogixDesigner"

Required Files

Device Objects include Add-On Instructions (AOIs) and HMI faceplates. The revision number (e.g. 1.01) used in filenames can change as new revisions are created.

Controller Files

Add-On Instructions are reusable code objects that contain encapsulated logic that can streamline implementing your system. This lets you create your own instruction set for programming logic as a supplement to the instruction set

provided natively in the ControlLogix® firmware. An Add-On Instruction is defined once in each controller project, and can be instantiated multiple times in your application code as needed.

The Add-On Instruction must be imported into the controller project to be used in the controller configuration. These can be imported as Add-On Instruction files, or as part of the ROUTINE Import or Import Library Objects wizard.

All Add-On Instruction and ROUTINE Import files can be found in the */Studio 5000 Logix Designer® Files - L5X/* folder in the library.

Device/Item	Add-On Instruction	ROUTINE Import
CROUT	raC_Dvc_CROUT_1.01_AOI.L5X	raC_Dvc_CROUT_1.01_ROUTINE.L5X

FactoryTalk® View HMI Files

FactoryTalk® View ME or SE applications require importing the desired device faceplates in addition to all Global Object (ggfx) files and all images located in the */HMI FactoryTalk View Images - png/* folder of the library. FactoryTalk® View ME files are stored in the */HMI - FactoryTalk View ME/* library folder and FactoryTalk® View SE files are stored in the */HMI - FactoryTalk View SE/* library folder.

Device/Item	Type	FactoryTalk® View ME Faceplate	FactoryTalk® View SE Faceplate
CROUT	Display	(raC-1_01-ME) raC_Dvc_CROUT-Faceplate.gfx	(raC-1_01-SE) raC_Dvc_CROUT-Faceplate.gfx
Graphic Symbols	Global Object	(raC-1-ME) Graphic Symbols - Safety Device	(raC-1-SE) Graphic Symbols - Safety Device.ggfx
Toolbox	Global Object	(raC-1-ME) Toolbox - Safety Device.ggfx	(raC-1-SE) Toolbox - Safety Device.ggfx

Studio 5000 View Designer® HMI Files

All Studio 5000 View Designer® Files can be found in the */HMI - ViewDesigner - vpd/* folder of the library.

Device/Item	Studio 5000 View Designer® Faceplate
CROUT	(raC-1_02-VD) raC_Dvc_Safety.vpd

Studio 5000® Application Code Manager Files

Studio 5000® Application Code Manager (ACM) can be optionally used if it is installed. All devices can be easily registered in the ACM repositories by running the *setup.cmd* file located in the root folder of the library.

Individual HSL4 files are provided as an alternative to running the setup.cmd to allow users to manually register specific implementation objects. Each object has two files - an Asset Control file and a Device file. The Asset Control files include attachments of all required files for that object. The Device files are used to actually add that device into a Studio 5000 project and these reference the Asset Control files.

All Studio 5000® Application Code Manager files can be found in the /*ApplicationCodeManagerLibraries*/ folder of the library. The files included are as follows:

Implementation Object	Asset Control File (.HSL4)	Device File (.HSL4)
CROUT	(RA-LIB)_Device_Asset-Control_Safety_raC_Dvc_CROUT_(1.2)	(RA-LIB)_Device_Device_Safety_raC_LD_CROUT_(1.2)

Operations

The Safety Discrete objects provides only physical operation mode. There is no virtual device mode offered.

Execution

The following table explains the handling of instruction execution conditions.

Condition	Description
EnableIn False (false rung)	Processing for EnableIn False (false rung) is handled the same as if the device were taken out of service by Command. The device outputs are de-energized and the device is shown as Program Out of Service on the HMI. All alarms are cleared.
Powerup (prescan, first scan)	On prescan, any commands that are received before first scan are discarded. The device is de-energized. On first scan, the device is treated as if it were returning from Hand command source: the instruction state is set based on the position feedback that is received from the device. If the feedback is valid for one position, the device is set to that position. If the device does not have position feedback or the position feedback is invalid, the device is set to the 'unknown/powerup' state. The command source is set to its default, either Operator or Program (unlocked).
Postscan	No SFC Postscan logic is provided.

Add-On Instruction I/O Data

InOut Data

InOut	Function / Description	DataType
Inp_CROUT	Input Interface - CROUT Instruction	CONFIGURABLE_ROUT
Out_Ctrl_Inf	Information Interface	raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	Status Interface	raC_UDT_Itf_SafetyDevice_Sts

Input Data

Input	Function/Description	DataType
EnableIn	Enable Input - System Defined Parameter	BOOL

Output Data

Output	Function/Descritpion	DataType
Sts_SafetyDemand	Safety demand is present	BOOL
Sts_ResetRequired	Reset is required	BOOL
Sts_DiagnosticPresent	Diagnostic is present	BOOL
Sts_FaultPresent	Fault is present	BOOL
Sts_Ready	Device is ready	BOOL
Sts_ERR	Instruction Error Code	DINT
Sts_eReady	0: Ready 1: Safety Demand 2: Reset Required 3: Suspended	SINT
raC_Dvc_CROUT	Unique Parameter Name for auto - discovery	BOOL
EnableOut	Enable Output - System Defined Parameter	BOOL

Data Types

The following Safety Common Control Interface tags are the primary device program tags to read and write to when interfacing to safety devices. The value of using these tags in your specific application code is that you may use a number of different safety devices such as CROUT, DCS, etc without having to update your application device interface tags. The primary function of these interfaces within the Safety Device Library is to pass individual instruction information to the Safety Group Monitor instruction.

Refer to the Interfaces section for detailed information on interfaces.

raC_UDT_Itf_SafetyDevice_Inf

Member	Description	Data Type
Type	Provides device object type information e.g. raC_Dvc_SafetyGroupMonitor, raC_Dvc_CROUT, etc.	STR0032
DisplayName	Provides object display name	STR0024
eClass	Provide instruction type information eClass ID: Instruction Type 0: DCS 1: DCSTL 2: DCSTM 3: CROUT 5: SafetyGroupMonitor 6: TSAM 7: TSSM 8: FSBM 9: THRSe 10: SMAT 11:SS2 12: SS1 13:SFX 14:STO	DINT

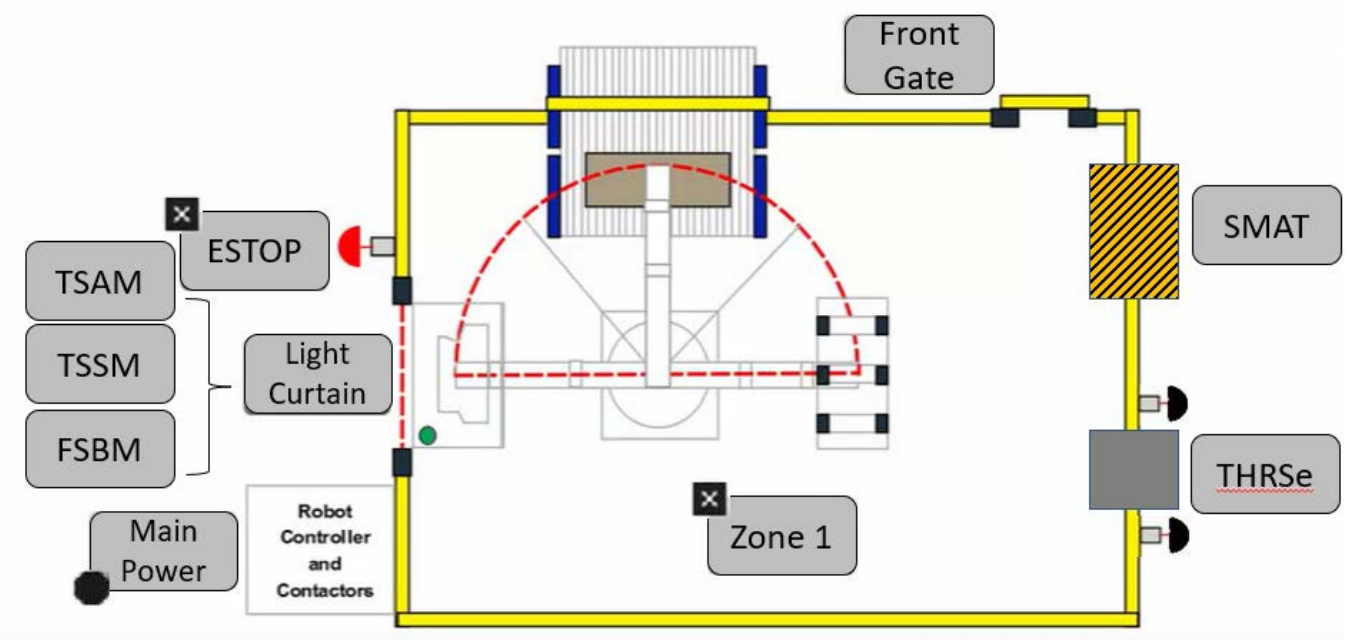
raC_UDT_Itf_SafetyDevice_Sts

This is the Safety Device Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data. The table below shows member names, descriptions, and tag data types.

Input	Description	Data Type
bSts	Bit overlay (Visible) covering all subsequent Boolean members. (Unused)	DINT
DeviceSuspended	Device suspended status (Inp_Suspend = 1)	BOOL
SafetyDemand	Safety demand status ON (1): Safety demand is present on the device object OFF (0): No safety demand	BOOL
ResetRequired	Reset Required status ON (1): Device requires reset signal OFF (0): No reset required	BOOL
DiagnosticPresent	Diagnostic status ON (1): A diagnostic is present in the device object OFF (0): Device is operating normally	BOOL
FaultPresent	Fault Status ON (1): A fault is present in the device object OFF (0): Device is operating normally	BOOL
Muted	Device Muted status (Available for CROUT) ON (1): Muted status is present on one or more safety instructions within a specific safety group OFF (0): None of the instructions within group is muted	BOOL

Programming Example

Fully configured device routine is provided below for reference. This example is taken from the *Safety Program - Outputs* routine in the *SafetytApplication.ACD* example application provided in the library.

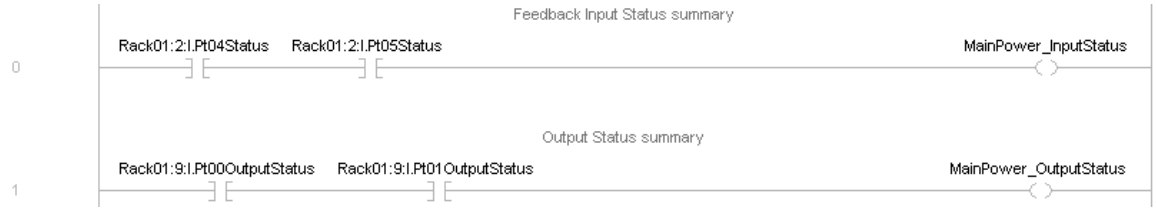


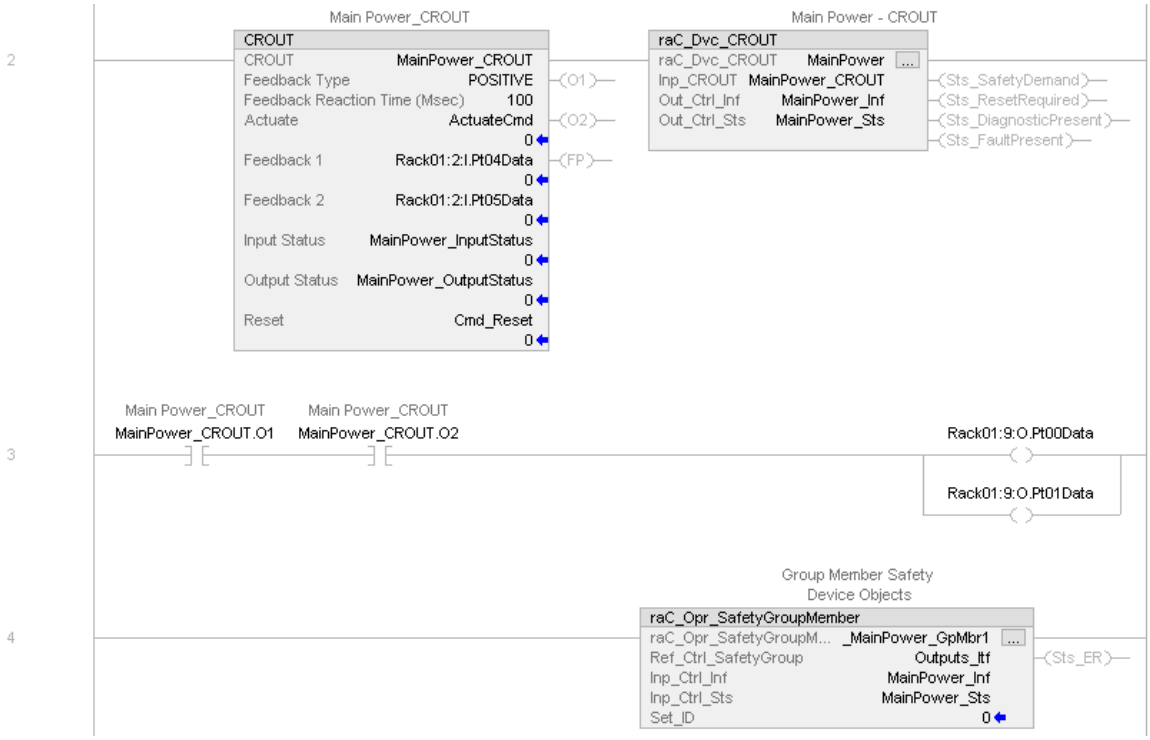
This example uses the CROUT instruction for a main power control function shown as *Main Power* in the safe machine diagram. Feedback 1 and Feedback 2 are tied to safety class discrete inputs. The outputs 01 and 02 from the CROUT pre-defined safety instruction drive safety class discrete outputs.

The input channel statuses are checked on rung 0. The output channel statuses are checked on rung 1. The CROUT and raC_Dvc_CROUT instructions are configured on rung 2. The safety discrete outputs are commanded on rung 3. The raC_Opr_SafetyGroupMember is optional and shown on rung 4.

Rung descriptions are as follows:

- Rung 0: Check input channel statuses
- Rung 1: Check output channel statuses
- Rung 2: Configure CROUT and raC_Dvc_CROUT instructions
- Rung 3: Safety discrete outputs are commanded
- Rung 4: raC_Opr_SafetyGroupMember (optional) assigns MainPower CROUT to Outputs Safety Group at ID #0.





Note that this programming example is the same code that is imported when either importing the supplied ROUTINE .L5X files or when using Application Code Manager or the Studio 5000 Import Library Objects wizard plug-in.

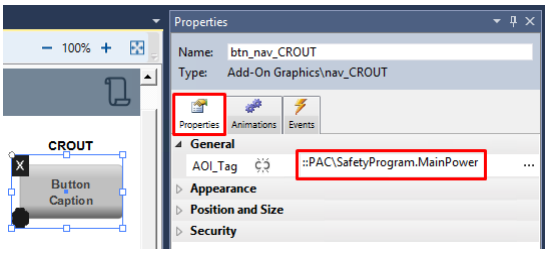
Graphic Symbols

Graphic Symbols are used as launch buttons within HMI applications to open up faceplate displays. See Basic Launch Button Attributes section for details on configuration and indicators

FactoryTalk® View ME/SE Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Global Object Parameter Values
btn_nav_safety		Faceplate navigation button with string tag label. This launch button graphic object allows the user to navigate to the device object faceplate. The text on the button face is set to the tag's .Inf_DisplayName.	#101: AOI Instance (e.g. {::[PAC]Program::SafetyProgram._InstanceName})

Studio 5000 View Designer® Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Property Configuration
Launch		The supplied launch button in View Designer is used to navigate to the faceplate in a user application.	

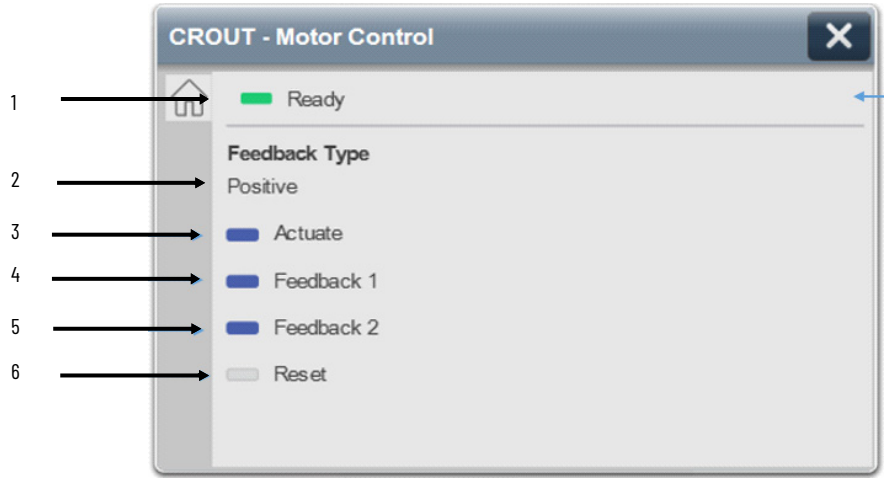
Faceplates

There are basic faceplate attributes that are common across all instructions. See Basic Faceplate Attributes on page 27.

Home

The Home tab is the main tab of the faceplate. It provides the status of the safety input device along with configuration information.

The Home tab shows the input type and input channel A/B statuses. Additionally there is a Ready banner at the top showing the status of the device.

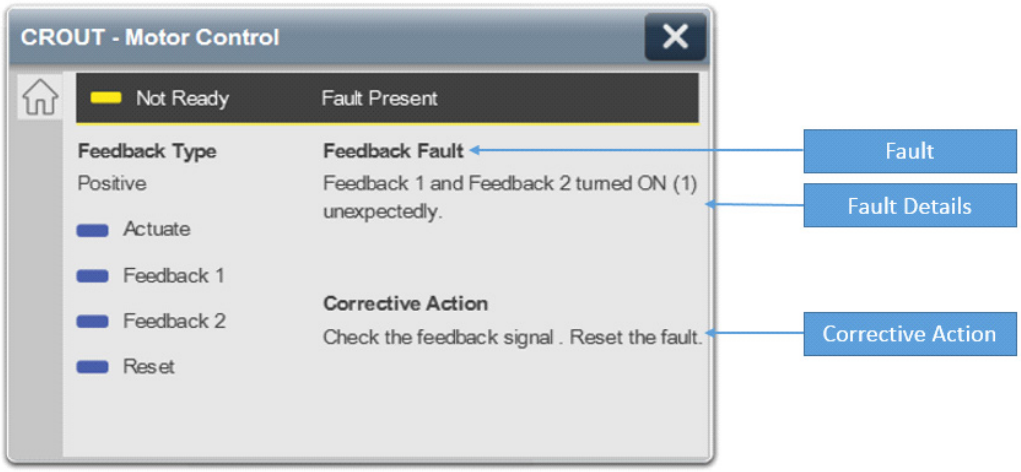


Item	Description
1	Banner- Ready Status
2	Feedback Type Positive = ON(1): Feedbacks ON/Outputs ON; OFF(0): Feedbacks OFF/Outputs OFF Negative = ON(1): Feedbacks OFF/Outputs ON; OFF(0): Feedbacks ON/Outputs OFF
3	Actuate = This input energizes or de-energizes Output 1 and Output 2. OFF (0) = Gray LED. Output 1 and Output 2 are de-energized ON (1) = Blue LED. Output 1 and Output 2 are energized.

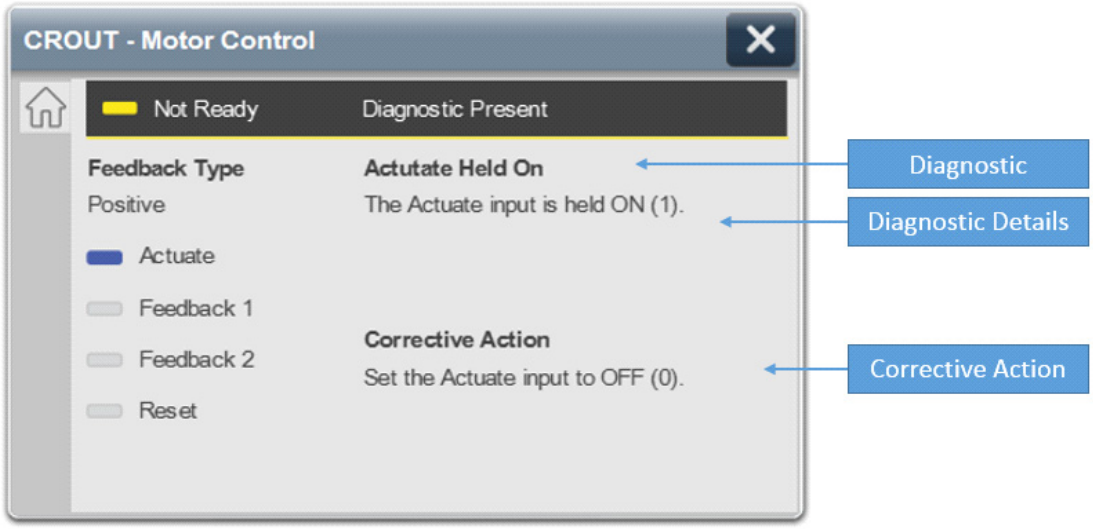
Item	Description
4	Feedback 1 Status. This input is constantly monitored to make sure that it reflects the sate of Output 1. When Output 1 transitions this input must detect the transition within the feedback reaction time. OFF (0) = Gray LED ON (1) = Blue LED
5	Feedback 2 Status. This input is constantly monitored to make sure that it reflects the sate of Output 1. When Output 1 transitions this input must detect the transition within the feedback reaction time. OFF (0) = Gray LED ON (1) = Blue LED
6	Reset Status. Reset clears the instruction faults provided the fault condition is not present. OFF (0) -> ON (1) = The fault is cleared OFF (0) = Gray LED ON (1) = Blue LED

Home tab screen also provides fault and diagnostic information with required corrective actions on the right plane of the faceplate

When there is fault present, banner with black background is displayed on top of home screen with **Not Ready** and **Fault Present** text.



When there is diagnostic present, banner with black background is displayed on top of home screen with **Not Ready** and **Diagnostic Present** text.



Application Code Manager

Safety Device Library objects can be set-up and configured using Studio 5000® Application Code Manager.

Refer to the section Using Application Code Manager for complete details.

Definition Object: raC_Dvc_CROUT

This object contains the AOI definition and used as linked library to implement object. This gives flexibility to choose to instantiate only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Implementation Object: raC_LD_Dvc_CROUT

Parameter Name	Default Value	Instance Name	Definition	Description
RoutineName	{ObjectName}	{RoutineName}	Routine	Enter Routine name. Routine will be created and Object implement rung(s) inserted. A JSR will be inserted in MainRoutine. If routine name already exists, then object will be inserted into existing routine. By default, parameter is set to Object Name.
TagName	{ObjectName}	{TagName}	Backing Tag	Enter the backing tag of the main AOI. This will serve as the base tag name for other tags in this object that are derived from the base.
TagDescription	{ObjectDescription}	{TagDescription}		Tag Description of the main AOI backing tag
DisplayName	[ObjectName]	[DisplayName]		Faceplate title/group event list description. Please avoid underscore ("_") at the end.
FeedbackType	POSITIVE			This operand defines the feedback ON and OFF states.
FeedbackReactionTme	100			This operand specifies the amount of time that the instruction waits for Feedback 1 and Feedback 2 to reflect the state of Output 1 and Output 2 as specified by the configured Feedback Type. The valid range is 5 to 1000 ms.
ActuateCmd	ActuateCmd		Controller Tage	This input energizes or de-energizes Output 1 and Output 2
ColdStartType	Automatic			This parameter specifies the Output 1 behavior when applying controller power or mode change to Run. Automatic or Manual.
Feedback1	Fdbk1		Input Parameter	Enter the input address (data) tag for Feedback 1 from safety digital input module.
Feedback1_InputStatus	Fdbk1_Sts		Input Parameter	Enter the input address (Status) tag for Feedback 1 from safety digital input module.

Parameter Name	Default Value	Instance Name	Definition	Description
Feedback2	Fdbk2		Input Parameter	Enter the input address (data) tag for Feedback 2 from safety digital input module.
Feedback2_InputStatus	Fdbk2_Sts		Input Parameter	Enter the input address (Status) tag for Feedback 2 from safety digital input module.
Output1	Out1		Input Parameter	Enter the output address (data) tag for Output 1 from safety digital output module.
Output1_Status	Out1_Sts		Input Parameter	Enter the output address (Status) tag for Output 1 from safety digital output module.
Output2	Out2		Input Parameter	Enter the output address (data) tag for Output 2 from safety digital output module.
Output2_Status	Out2_Sts		Input Parameter	Enter the output address (Status) tag for Output 2 from safety digital output module.
ResetTag	Cmd_Reset		Controller Tag	Enter Tagname for Reset signal. Please avoid underscore ("_") at the end.

Note that if the tag names are manually entered or not linked to input channel tags a red 'X' will be shown beside the parameter. This is acceptable and the program can still be generated.

Linked Libraries

Link Name	Catalog Number	Revision	Solution	Category
raC_Dvc_CROUT	raC_Dvc_CROUT	1	(RA-LIB) Device	Safety
raC_Opr_SafetyGroupMember	raC_Opr_SafetyGroupMember	1	(RA-LIB) Device	Safety

Configured HMI Content

HMI Content	Instance Name	Description
Launch Button	{ObjectName}_GO_LaunchFP	Global Object configured callout instance

Attachments

Name	Description	File Name	Extraction Path
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols SE	(raC-1-SE) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewSE\Global Objects
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols ME	(raC-1-ME) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewME\Global Objects

V1_raC_Dvc_CROUT	Faceplate SE	(raC-1_xx-SE) raC_Dvc_CROUT-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
V1_raC_Dvc_CROUT	Faceplate ME	(raC-1_xx-ME) raC_Dvc_CROUT-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
V1_raC_Dvc_CROUT	View Designer	(raC-1_xx-VD) raC_Dvc_Safety.vpd	{ProjectName}\Visualization\ViewDesigner
V1_RM_raC_DvcSafety	Reference Manual	DEVICE-RM500C-EN-P.pdf	{ProjectName}\Documentation
V1_Safety_Images	HMI Image Set	Safety_Images.zip	{ProjectName}\Visualization\Images

TSAM - Two Sensor Asymmetrical Muting (raC_Dvc_TSAM)

Overview

The Two Sensor Asymmetrical Muting device object (raC_Dvc_TSAM) includes a faceplate which displays status and configuration information of safety applications using two muting sensors arranged asymmetrically on either side of a light curtain. The primary function is to automatically disable the protective function of a light curtain which allows material to be transported through the light curtain sensing field without stopping the machine.

The instruction is ready when material clears Sensor 1, Sensor 2 and the Light Curtain are in the active state as determined by the Input type operand, and the correct reset actions are carried out.

The TSAM instruction also requires a DCS (Dual Channel Input Stop) instruction.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following videos for this section:
"Operational_Overview_of_Safety_Device_Object_Faceplates"
"Operational_Overview_of_Safety_Application_Example_in_LogixDesigner"

Functional Description

The Two Sensor Asymmetrical Muting pre-configured Device Objects:

- Collect, Process and Deliver Data between Smart Devices and Application Logic
- Provide Device Status & Diagnostics Faceplates for Machine Startup, Commissioning, Operations, and Maintenance
- Include Common HMI Faceplate Components, Device States, and Control Interfaces providing Application Development and Operation Efficiency

All these features provide quick feedback, shorten recovery time, and simplify implementation.

Required Files

Device Objects include Add-On Instructions (AOIs) and HMI faceplates. The revision number (e.g. 1.01) used in filenames can change as new revisions are created.

Controller Files

Add-On Instructions are reusable code objects that contain encapsulated logic that can streamline implementing your system. This lets you create your own instruction set for programming logic as a supplement to the instruction set provided natively in the ControlLogix® firmware. An Add-On Instruction is defined once in each controller project, and can be instantiated multiple times in your application code as needed.

The Add-On Instruction must be imported into the controller project to be used in the controller configuration. These can be imported as Add-On Instruction files, or as part of the ROUTINE Import or Import Library Objects wizard.

All Add-On Instruction and ROUTINE Import files can be found in the */Studio 5000 Logix Designer® Files - L5X/* folder in the library.

Device/Item	Add-On Instruction	ROUTINE Import
TSAM	raC_Dvc_TSAM_1.02_AOI.L5X	raC_Dvc_TSAM_1.02_ROUTINE.L5X

FactoryTalk® View HMI Files

FactoryTalk® View ME or SE applications require importing the desired device faceplates in addition to all Global Object (ggfx) files and all images located in the */HMI FactoryTalk View Images - png/* folder of the library. FactoryTalk® View ME files are stored in the */HMI - FactoryTalk View ME/* library folder and FactoryTalk® View SE files are stored in the */HMI - FactoryTalk View SE/* library folder.

Device/Item	Type	FactoryTalk® View ME Faceplate	FactoryTalk® View SE Faceplate
TSAM	Display	(raC-1_02-ME) raC_Dvc_TSAM-Faceplate.gfx	(raC-1_02-SE) raC_Dvc_TSAM-Faceplate.gfx
Graphic Symbols	Global Object	(raC-1-ME) Graphic Symbols - Safety Device	(raC-1-SE) Graphic Symbols - Safety Device.ggfx
Toolbox	Global Object	(raC-1-ME) Toolbox - Safety Device.ggfx	(raC-1-SE) Toolbox - Safety Device.ggfx

Studio 5000 View Designer® HMI Files

All Studio 5000 View Designer® Files can be found in the */HMI - ViewDesigner - vpd/* folder of the library.

Device/Item	Studio 5000 View Designer® Faceplate
TSAM	(raC-1_02-VD) raC_Dvc_Safety.vpd

Studio 5000® Application Code Manager Files

Studio 5000® Application Code Manager (ACM) can be optionally used if it is installed. All devices can be easily registered in the ACM repositories by running the *setup.cmd* file located in the root folder of the library.

Individual HSL4 files are provided as an alternative to running the setup.cmd to allow users to manually register specific implementation objects. Each object has two files - an Asset Control file and a Device file. The Asset Control files include attachments of all required files for that object. The Device files are used to actually add that device into a Studio 5000 project and these reference the Asset Control files.

All Studio 5000® Application Code Manager files can be found in the /*ApplicationCodeManagerLibraries*/ folder of the library. The files included are as follows:

Implementation Object	Asset Control File (.HSL4)	Device File (.HSL4)
TSAM	(RA-LIB)_Device_Asset-Control_Safety_raC_Dvc_TSAM_(1.2)	(RA-LIB)_Device_Device_Safety_raC_LD_TSAM_(1.2)

Operations

The Safety Discrete objects provides only physical operation mode. There is no virtual device mode offered.

Execution

The following table explains the handling of instruction execution conditions.

Condition	Description
EnableIn False (false rung)	Processing for EnableIn False (false rung) is handled the same as if the device were taken out of service by Command. The device outputs are de-energized and the device is shown as Program Out of Service on the HMI. All alarms are cleared.
Powerup (prescan, first scan)	On prescan, any commands that are received before first scan are discarded. The device is de-energized. On first scan, the device is treated as if it were returning from Hand command source: the instruction state is set based on the position feedback that is received from the device. If the feedback is valid for one position, the device is set to that position. If the device does not have position feedback or the position feedback is invalid, the device is set to the 'unknown/powerup' state. The command source is set to its default, either Operator or Program (unlocked).
Postscan	No SFC Postscan logic is provided.

Add-On Instruction I/O Data

InOut Data

InOut	Function / Description	DataType
Inp_TSAM	Input Interface - TSAM Instruction	DCI_MUTING_TWO_SENSOR_ASYM
Inp_DCS	Input Interface - DCS Instruction	DCI_STOP
Out_Ctrl_Inf	Information Interface	raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	Status Interface	raC_UDT_Itf_SafetyDevice_Sts

Input Data

Input	Function/Description	DataType
EnableIn	Enable Input - System Defined Parameter	BOOL
Inp_Suspended	Suspend device operation	BOOL

Output Data

Output	Function/Descritpion	DataType
Sts_DeviceSuspended	Device is suspended	BOOL
Sts_SafetyDemand	Safety demand is present	BOOL
Sts_ResetRequired	Reset is required	BOOL
Sts_DiagnosticPresent	Diagnostic is present	BOOL
Sts_FaultPresent	Fault is present	BOOL
Sts_MuteLampON	This output indicates the status of the light curtain's protective function. ON (1): The light curtain's protective function is disabled. OFF (0): The light curtain's protective function is enabled.	BOOL
Sts_ClearArea	This output indicates when the light curtain sensing field must be cleared (all muting sensors and the light curtain are ON) before processing can continue. ON (1): The light curtain sensing field must be cleared. OFF (0): The light curtain sensing field is clear.	BOOL
Sts_OAClearingLCA	The light curtain sensing field is not obstructed but the field must be cleared and override is active. ON(1): Override Activate and light Curtain area in clearing state. OFF (0): Light curtain area not in clearing state	BOOL
Sts_OAClearedLCA	The light curtain sensing field is not obstructed, the field is cleared and override is active. ON (1): Override Activate and light Curtain area cleared OFF (0): Light curtain area not cleared	BOOL
Sts_Muted	Muted status ON (1): Device is muted and protective function is disabled OFF (0): Device is not muted and protective function is enabled.	BOOL
Sts_Ready	Device is ready	BOOL
Sts_ERR	Instruction Error Code	DINT
Sts_eReady	0: Ready 1: Safety Demand 2: Reset Required 3: Suspended	SINT
raC_Dvc_TSAM	Unique Parameter Name for auto - discovery	BOOL
EnableOut	Enable Output - System Defined Parameter	BOOL

Data Types

The following Safety Common Control Interface tags are the primary device program tags to read and write to when interfacing to safety devices. The value of using these tags in your specific application code is that you may use a number of different safety devices such as TSAM, TSSM, etc without having to update your application device interface tags. The primary function of these interfaces within the Safety Device Library is to pass individual instruction information to the Safety Group Monitor instruction.

Refer to the Interfaces section for detailed information on interfaces.

raC_UDT_Itf_SafetyDevice_Inf

Member	Description	Data Type
Type	Provides device object type information e.g. raC_Dvc_SafetyGroupMonitor, raC_Dvc_TSAM, raC_Dvc_TSAMTL, etc.	STR0032
DisplayName	Provides object display name	STR0024
eClass	Provide instruction type information eClass ID: Instruction Type 0: DCS 1: DCSTL 2: DCSTM 3: CROUT 5: SafetyGroupMonitor 6: TSAM 7: TSSM 8: FSBM 9: THRSe 10: SMAT 11:SS2 12: SS1 13:SFX 14:STO	DINT

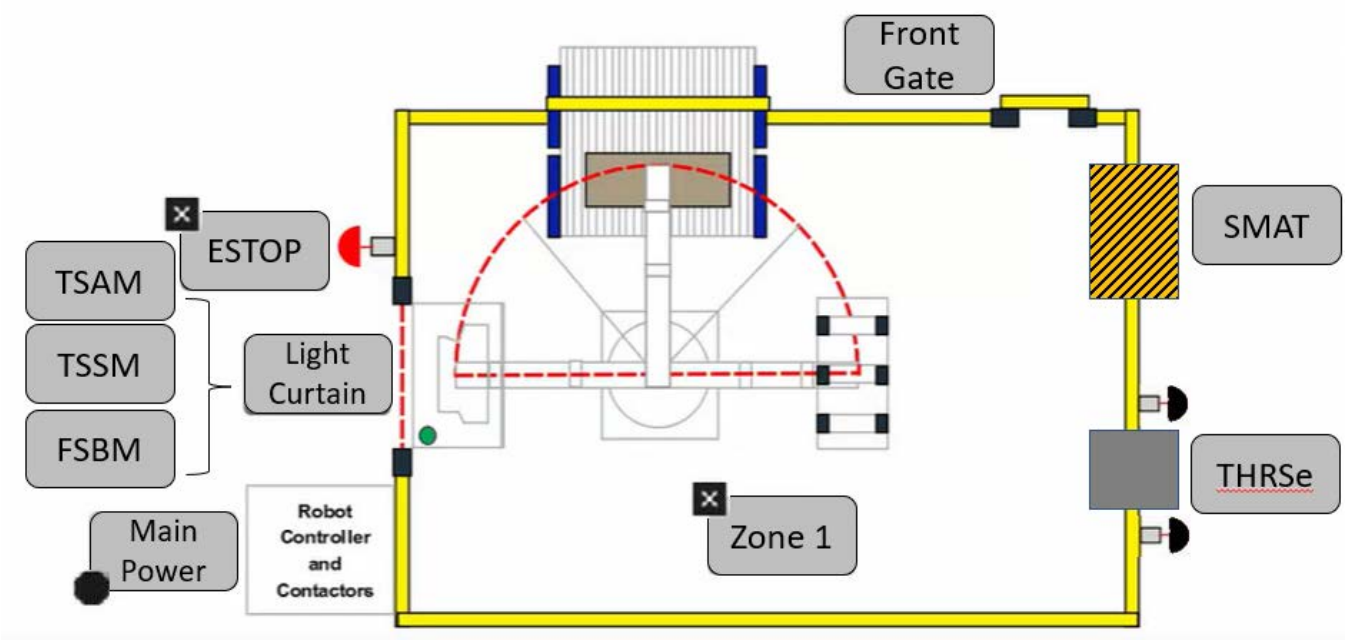
raC_UDT_Itf_SafetyDevice_Sts

This is the Safety Device Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data. The table below shows member names, descriptions, and tag data types.

Input	Description	Data Type
bSts	Bit overlay (Visible) covering all subsequent Boolean members. (Unused)	DINT
DeviceSuspended	Device suspended status (Inp_Suspend = 1)	BOOL
SafetyDemand	Safety demand status ON (1): Safety demand is present on the device object OFF (0): No safety demand	BOOL
ResetRequired	Reset Required status ON (1): Device requires reset signal OFF (0): No reset required	BOOL
DiagnosticPresent	Diagnostic status ON (1): A diagnostic is present in the device object OFF (0): Device is operating normally	BOOL
FaultPresent	Fault Status ON (1): A fault is present in the device object OFF (0): Device is operating normally	BOOL
Muted	Device Muted status (Available for TSAMTM) ON (1): Muted status is present on one or more safety instructions within a specific safety group OFF (0): None of the instructions within group is muted	BOOL

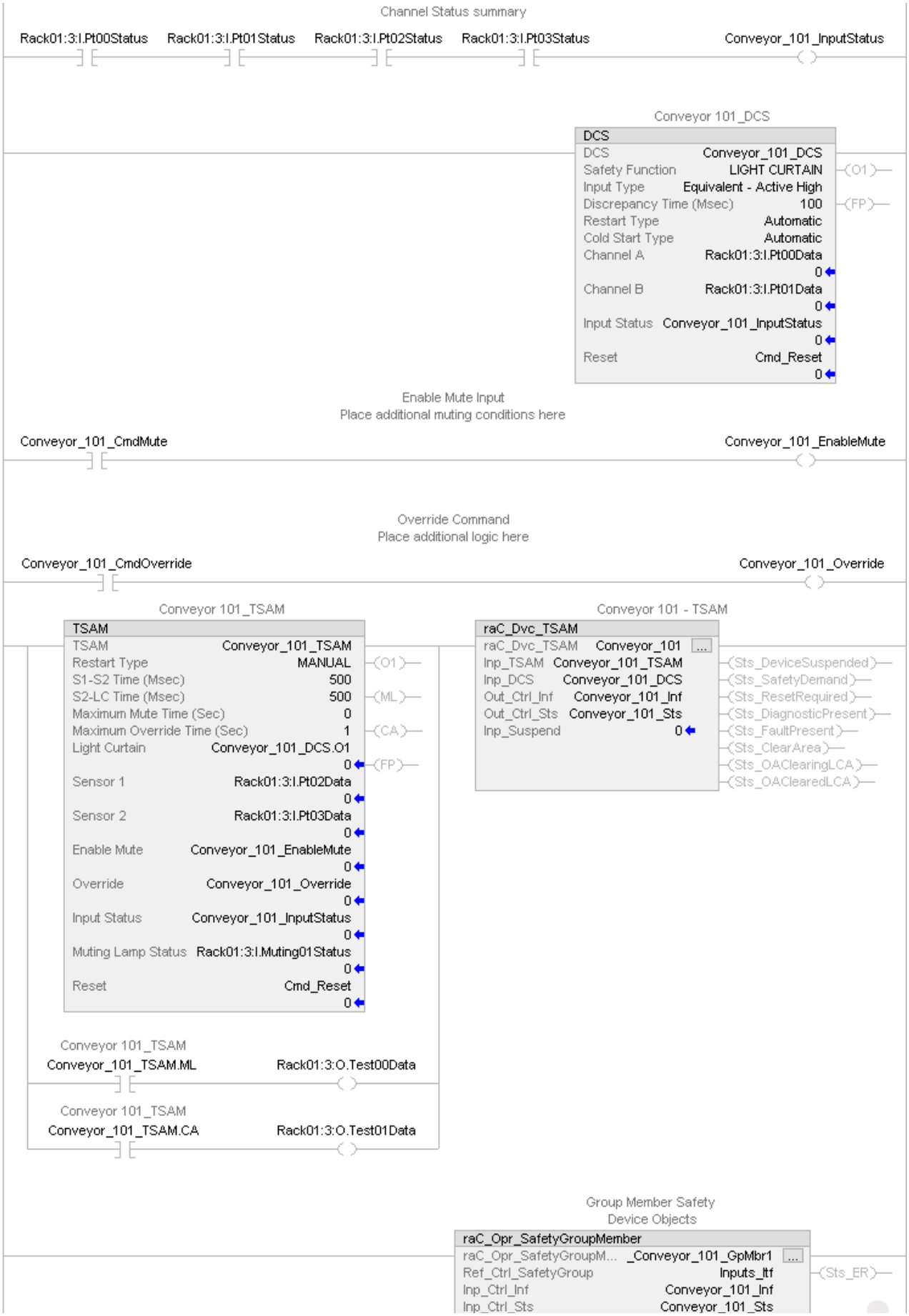
Programming Example

Fully configured device routine is provided below for reference. This example is taken from the *Safety Program - Inputs* routine in the *SafetytApplication.ACD* example application provided in the library.



This example uses the TSAM instruction for a light curtain function shown as *TSAM* in the safe machine diagram. A DCS instruction is also used for the light curtain. This system in the example application code is referred to as Conveyor_101. DCS Channel A and Channel B are tied to safety class discrete inputs. TSAM Sensors 1 and 2 as well as muting lamp status are also connected to safety class discrete inputs. Rung descriptions are as follows:

- Rung 12: Check input statuses
- Rung 13: Configure DCS instruction
- Rung 14: Check mute input status
- Rung 15: Check Override status
- Rung 16: Configure TSAM and raC_Dvc_TSAM instructions
- Rung 17: raC_Opr_SafetyGroupMember (optional) assigns Conveyor_101 TSAM to Inputs Safety Group at ID #4.


Channel Status summary
12
Rack01:3:I.Pt00Status
Rack01:3:I.Pt01Status
Rack01:3:I.Pt02Status
Rack01:3:I.Pt03Status
Conveyor_101_InputStatus
Conveyor 101_DCS
13
DCS
DCS Conveyor_101_DCS
Safety Function LIGHT CURTAIN
Input Type Equivalent - Active High
Discrepancy Time (Msec) 100
Restart Type Automatic
Cold Start Type Automatic
Channel A Rack01:3:I.Pt00Data
0
Channel B Rack01:3:I.Pt01Data
0
Input Status Conveyor_101_InputStatus
0
Reset Cmd_Reset
0
O1
FP
Enable Mute Input
Place additional muting conditions here
14
Conveyor_101_CmdMute
Conveyor_101_EnableMute
Override Command
Place additional logic here
15
Conveyor_101_CmdOverride
Conveyor_101_Override
Conveyor 101_TSAM
16
TSAM
TSAM Conveyor_101_TSAM
Restart Type MANUAL
S1-S2 Time (Msec) 500
S2-LC Time (Msec) 500
Maximum Mute Time (Sec) 0
Maximum Override Time (Sec) 1
Light Curtain Conveyor_101_DCS.O1
0
Sensor 1 Rack01:3:I.Pt02Data
0
Sensor 2 Rack01:3:I.Pt03Data
0
Enable Mute Conveyor_101_EnableMute
0
Override Conveyor_101_Override
0
Input Status Conveyor_101_InputStatus
0
Muting Lamp Status Rack01:3:I.Muting01Status
0
Reset Cmd_Reset
0
O1
ML
CA
FP
Conveyor 101 - TSAM
raC_Dvc_TSAM
raC_Dvc_TSAM Conveyor_101
Inp_TSAM Conveyor_101_TSAM
Inp_DCS Conveyor_101_DCS
Out_Ctrl_Inf Conveyor_101_Inf
Out_Ctrl_Sts Conveyor_101_Sts
Inp_Suspend 0
Sts_DeviceSuspended
Sts_SafetyDemand
Sts_ResetRequired
Sts_DiagnosticPresent
Sts_FaultPresent
Sts_ClearArea
Sts_OAClearingLCA
Sts_OAClearedLCA
Conveyor 101_TSAM
Conveyor_101_TSAM.ML
Rack01:3:O.Test00Data
Conveyor 101_TSAM
Conveyor_101_TSAM.CA
Rack01:3:O.Test01Data
Group Member Safety
Device Objects
17
raC_Opr_SafetyGroupMember
raC_Opr_SafetyGroupM... _Conveyor_101_GpMbr1
Ref_Ctrl_SafetyGroup Inputs_Itf
Inp_Ctrl_Inf Conveyor_101_Inf
Inp_Ctrl_Sts Conveyor_101_Sts
Sts_ER

Note that this programming example is the same code that is imported when either importing the supplied ROUTINE.L5X files or when using Application Code Manager or the Studio 5000 Import Library Objects wizard plug-in.

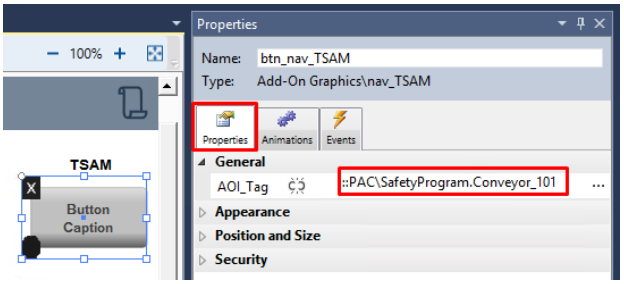
Graphic Symbols

Graphic Symbols are used as launch buttons within HMI applications to open up faceplate displays. See Basic Launch Button Attributes section for details on configuration and indicators

FactoryTalk® View ME/SE Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Global Object Parameter Values
btn_nav_safety		Faceplate navigation button with string tag label. This launch button graphic object allows the user to navigate to the device object faceplate. The text on the button face is set to the tag's .Inf_DisplayName.	#101: AOI Instance (e.g. {::[PAC]Program::SafetyProgram._InstanceName})

Studio 5000 View Designer® Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Property Configuration
Launch		The supplied launch button in View Designer is used to navigate to the faceplate in a user application.	

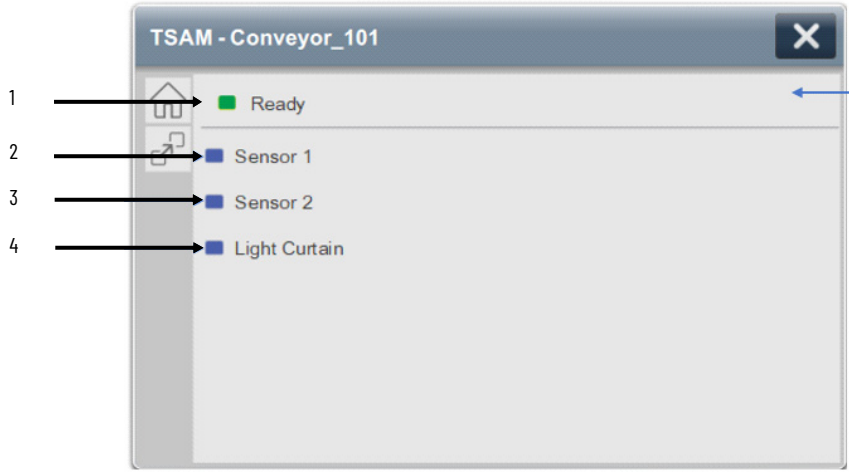
Faceplates

There are basic faceplate attributes that are common across all instructions. See Basic Faceplate Attributes on page 27.

Home

The Home tab is the main tab of the faceplate. It provides the status of the safety input device along with configuration information.

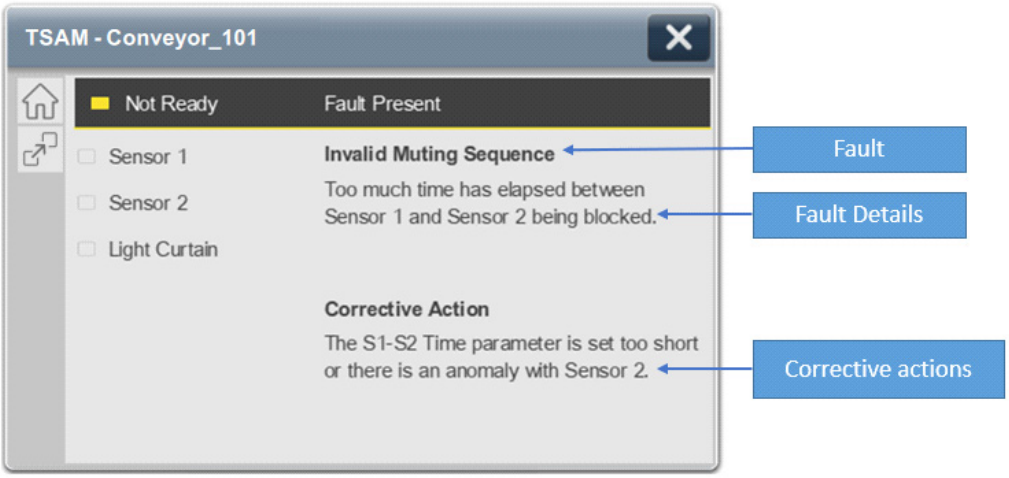
The Home tab shows the Sensor 1, Sensor 2 and Light Curtain cleared status. Additionally there is a Ready banner at the top showing the status of the device.



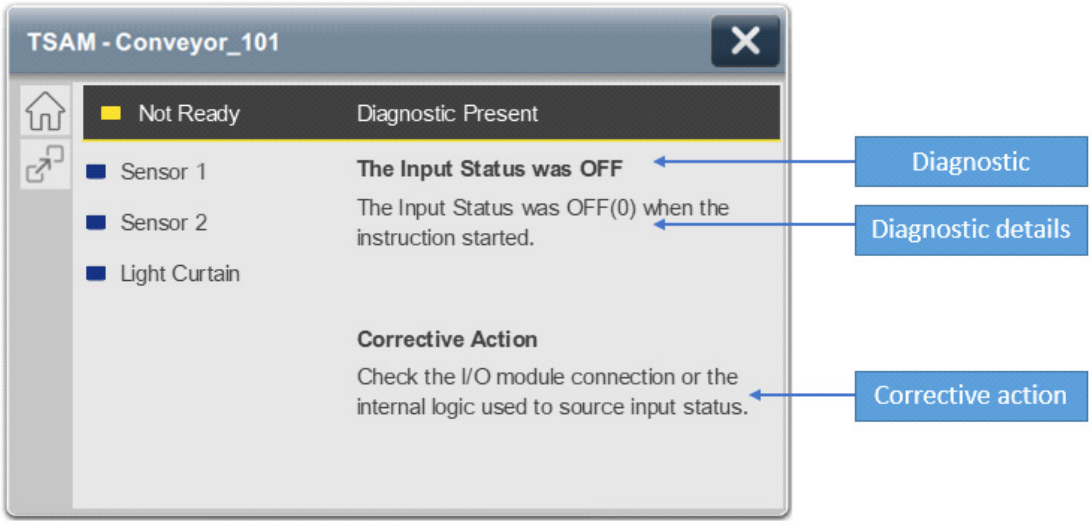
Item	Description
1	Banner- Ready Status
2	Sensor 1 Status. This is one of the two muting sensors; Sensor 1 must be the first sensor to be blocked and the last to be cleared in the muting sequence and its status is indicated by: OFF (0) = Gray LED. Sensor 1 is blocked. ON (1) = Blue LED. Sensor 1 is cleared.
3	Sensor 2 Status. This is one of the two muting sensors; Sensor 2 must be the second sensor to be blocked and the first to be cleared in the muting sequence and its status is indicated by: OFF (0) = Gray LED. Sensor 2 is blocked. ON (1) = Blue LED. Sensor 2 is cleared.
4	Light Curtain Status. This is an input channel which represents the current state of the physical light curtain. Light curtain must be the last input to be blocked and first to be cleared in muting sequence and its status is indicated by: OFF (0) = Gray LED. Light Curtain is blocked ON (1) = Blue LED. Light Curtain is cleared

Home tab screen also provides fault and diagnostic information with required corrective actions on the right plane of the faceplate

When there is fault present, banner with black background is displayed on top of home screen with **Not Ready** and **Fault Present** text.

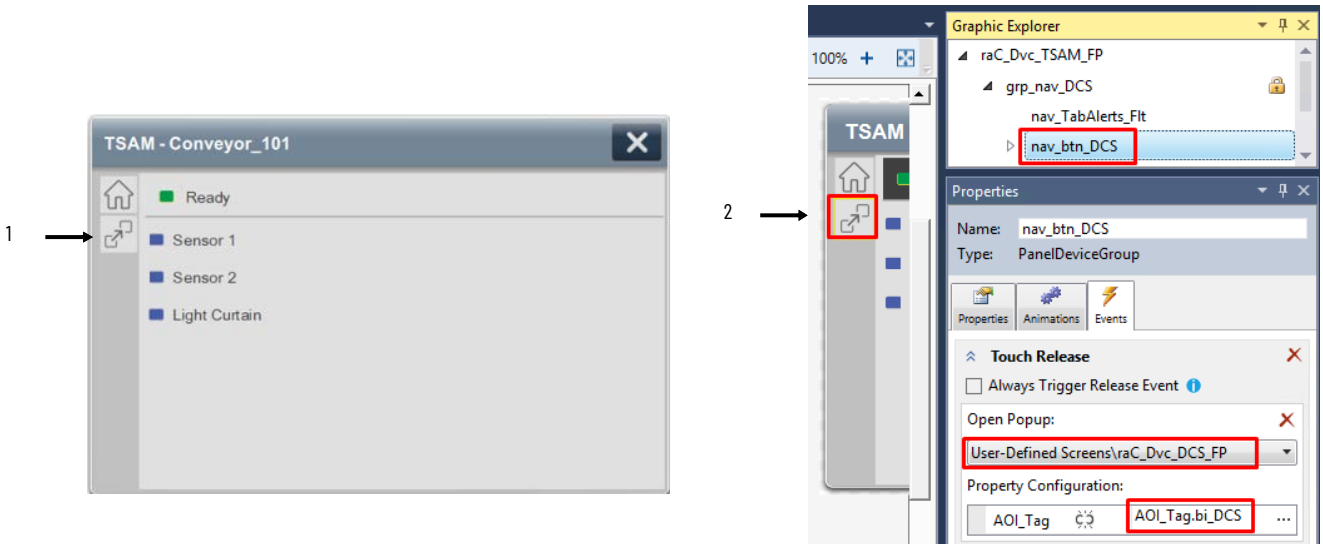


When there is diagnostic present, banner with black background is displayed on top of home screen with **Not Ready** and **Diagnostic Present** text.



Extensions - DCS

Below the Home Tab button there is an Extension button which launches the associated Dual Channel Input Stop (DCS) faceplate. See the section on raC_Dvc_DCS for complete details on the DCS object. Click the 'X' in the top right to close the DCS faceplate and return to the original device faceplate.



Item	Description
1	DCS Extension Navigation Tab Button
2	DCS Extension Navigation Configuration settings

Application Code Manager

Safety Device Library objects can be set-up and configured using Studio 5000® Application Code Manager.

Refer to the section Using Application Code Manager for complete details.

Definition Object: raC_Dvc_TSAM

This object contains the AOI definition and used as linked library to implement object. This gives flexibility to choose to instantiate only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Implementation Object: raC_LD_Dvc_TSAM

Parameter Name	Default Value	Instance Name	Definition	Description
RoutineName	{ObjectName}	{RoutineName}	Routine	Enter Routine name. Routine will be created and Object implement rung(s) inserted. A JSR will be inserted in MainRoutine. If routine name already exists, then object will be inserted into existing routine. By default, parameter is set to Object Name.
TagName	{ObjectName}	{TagName}	Backing Tag	Enter the backing tag of the main AOI. This will serve as the base tag name for other tags in this object that are derived from the base.
TagDescription	{ObjectDescription}	{TagDescription}		Tag Description of the main AOI backing tag
DisplayName	[ObjectName]	[DisplayName]		Faceplate title/group event list description. Please avoid underscore ("_") at the end.
SafetyFunction	LIGHT CURTAIN			This parameter provides a text name for how this instruction is being used. It is for information/documentation purposes only. Options: - USER DEFINED - EMERGENCY STOP - SAFETY GATE - LIGHT CURTAIN - AREA SCANNER - SAFETY MAT - CABLE (ROPE) PULL SWITCH
InputType	Equivalent - Active High			This parameter selects input channel behavior. Equivalent - Active High or Complimentary.
DiscrepencyTime	100			The amount of time that the inputs can be in an inconsistent state before an instruction fault is generated. Range 5-3000ms.
RestartType_DCS	Automatic			This input configures Output 1 for either Manual or Automatic Restart.
ColdStartType_DCS	Automatic			This parameter specifies the Output 1 behavior when applying controller power or mode change to Run. Automatic or Manual.
ChannelA	ChA		Input Parameter	Enter the input address (data) tag for Channel A from safety digital input module.
ChannelA_InputStatus	ChA_Sts		Input Parameter	Enter the input address (Status) tag for Channel A from safety digital input module.
ChannelB	ChB		Input Parameter	Enter the input address (data) tag for Channel B from safety digital input module.

Parameter Name	Default Value	Instance Name	Definition	Description
ChannelB_InputStatus	ChB_Sts		Input Parameter	Enter the input address(Status)tag for Channel B from safety digital input module.
RestartType	Automatic			This input configures Output 1 for either Manual of Automatic Restart
S1S2Time	0			The maximum amount of time allowed between clearing or blocking of the muting sensor inputs Sensor 1 and Sensor 2 before generating a fault. Setting this input to 0 disables the S1-S2 timer. The valid range is 5-180,000 ms.
S1LCTime	0			The maximum amount of time allowed between clearing or blocking of the Sensor 2 muting sensor and the Light Curtain before generating a fault. Setting this input to 0 disables the S2-LC timer. The valid range is 5-180,000 ms.
MaxMuteTime	0			The maximum amount of time during which the instruction lets the protective function of the light curtain be disabled before generating a fault. Setting this input to 0 disables the Maximum Mute timer. The valid range is 0-3,600,000 ms.
MaxOverrideTime	0			The maximum amount of time that the instruction lets the override feature energize the Output 1 output. Setting this input to 0 disables the Maximum Override timer. The valid range is 0-30,000 ms.
Sensor1	Sens1		Input Parameter	Enter the input address (data) tag for Sensor 1 from safety digital input module. That is one of two muting sensors, Sensor 1 must be the first sensor to be blocked and the last to be cleared in the muting sequence.
Sensor1_InputStatus	Sense1_Sts		Input Parameter	Enter the input address (Status) tag for Sensor 1 from safety digital input module.
Sensor2	Sens2		Input Parameter	Enter the input address (data) tag for Sensor 2 from safety digital input module. That is one of two muting sensors, Sensor 2 must be the second sensor to be blocked and the first to be cleared in the muting sequence.
Sensor2_InputStatus	Sense2_Sts		Input Parameter	Enter the input address (Status) tag for Sensor 2 from safety digital input module.
MutingLampStatus	MuteInput_Sts		Input Parameter	Enter the input address tag showing status of the muting lamp.
MutingLamp_Output	ML_Output		Input Parameter	Enter the test output address (Data) tag to drive the Muting Lamp from safety digital input module.
ClearArea_Output	CA_Output		Input Parameter	Enter the test output address (Data) tag to drive the Clear Area Lamp from safety digital input module.
ResetTag	Cmd_Reset		Controller Tag	Enter Tagname for Reset signal. Please avoid underscore("_") at the end.

Note that if the tag names are manually entered or not linked to input channel tags a red 'X' will be shown beside the parameter. This is acceptable and the program can still be generated.

Linked Libraries

Link Name	Catalog Number	Revision	Solution	Category
raC_Dvc_TSAM	raC_Dvc_TSAM	1	(RA-LIB) Device	Safety

raC_Opr_SafetyGroupMember	raC_Opr_SafetyGroupMember	1	(RA-LIB) Device	Safety
raC_Dvc_DCS	raC_Dvc_DCS	1	(RA-LIB) Device	Safety

Configured HMI Content

HMI Content	Instance Name	Description
Launch Button	{ObjectName}_GO_LaunchFP	Global Object configured callout instance

Attachments

Name	Description	File Name	Extraction Path
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols SE	(raC-1-SE) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewSE\Global Objects
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols ME	(raC-1-ME) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewME\Global Objects
V1_raC_Dvc_TSAM	Faceplate SE	(raC-1_xx-SE) raC_Dvc_TSAM-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
V1_raC_Dvc_TSAM	Faceplate ME	(raC-1_xx-ME) raC_Dvc_TSAM-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
V1_raC_Dvc_TSAM	View Designer	(raC-1_xx-VD) raC_Dvc_Safety.vpd	{ProjectName}\Visualization\ViewDesigner
V1_RM_raC_DvcSafety	Reference Manual	DEVICE-RM500C-EN-P.pdf	{ProjectName}\Documentation
V1_Safety_Images	HMI Image Set	Safety_Images.zip	{ProjectName}\Visualization\Images

TSSM - Two Sensor Symmetrical Muting (raC_Dvc_TSSM)

Overview

The Two Sensor Symmetrical Muting device object (raC_Dvc_TSSM) includes a faceplate which displays status and configuration information of safety applications using two muting sensors arranged symmetrically on either side of a light curtain. The instruction provides a temporary automatic disabling of the protective function of a light curtain which allows material to be transported through the light curtain sensing field without stopping the machine.

The instruction is ready when material clears Sensor 1, Sensor 2 and the Light Curtain are in the active state as determined by the Input type operand, and the correct reset actions are carried out.

The TSSM instruction also requires a DCS (Dual Channel Input Stop) instruction.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following videos for this section:
"Operational_Overview_of_Safety_Device_Object_Faceplates"
"Operational_Overview_of_Safety_Application_Example_in_LogixDesigner"

Functional Description

The Two Sensor Symmetrical Muting pre-configured Device Objects:

- Collect, Process and Deliver Data between Smart Devices and Application Logic
- Provide Device Status & Diagnostics Faceplates for Machine Startup, Commissioning, Operations, and Maintenance
- Include Common HMI Faceplate Components, Device States, and Control Interfaces providing Application Development and Operation Efficiency

All these features provide quick feedback, shorten recovery time, and simplify implementation.

Required Files

Device Objects include Add-On Instructions (AOIs) and HMI faceplates. The revision number (e.g. 1.01) used in filenames can change as new revisions are created.

Controller Files

Add-On Instructions are reusable code objects that contain encapsulated logic that can streamline implementing your system. This lets you create your own instruction set for programming logic as a supplement to the instruction set provided natively in the ControlLogix® firmware. An Add-On Instruction is defined once in each controller project, and can be instantiated multiple times in your application code as needed.

The Add-On Instruction must be imported into the controller project to be used in the controller configuration. These can be imported as Add-On Instruction files, or as part of the ROUTINE Import or Import Library Objects wizard.

All Add-On Instruction and ROUTINE Import files can be found in the */Studio 5000 Logix Designer® Files - L5X/* folder in the library.

Device/Item	Add-On Instruction	ROUTINE Import
TSSM	raC_Dvc_TSSM_1.02_AOI.L5X	raC_Dvc_TSSM_1.02_ROUTINE.L5X

FactoryTalk® View HMI Files

FactoryTalk® View ME or SE applications require importing the desired device faceplates in addition to all Global Object (ggfx) files and all images located in the */HMI FactoryTalk View Images - png/* folder of the library. FactoryTalk® View ME files are stored in the */HMI - FactoryTalk View ME/* library folder and FactoryTalk® View SE files are stored in the */HMI - FactoryTalk View SE/* library folder.

Device/Item	Type	FactoryTalk® View ME Faceplate	FactoryTalk® View SE Faceplate
TSSM	Display	(raC-1_02-ME) raC_Dvc_TSSM-Faceplate.gfx	(raC-1_02-SE) raC_Dvc_TSSM-Faceplate.gfx
Graphic Symbols	Global Object	(raC-1-ME) Graphic Symbols - Safety Device	(raC-1-SE) Graphic Symbols - Safety Device.ggfx
Toolbox	Global Object	(raC-1-ME) Toolbox - Safety Device.ggfx	(raC-1-SE) Toolbox - Safety Device.ggfx

Studio 5000 View Designer® HMI Files

All Studio 5000 View Designer® Files can be found in the */HMI - ViewDesigner - vpd/* folder of the library.

Device/Item	Studio 5000 View Designer® Faceplate
TSSM	(raC-1_02-VD) raC_Dvc_Safety.vpd

Studio 5000® Application Code Manager Files

Studio 5000® Application Code Manager (ACM) can be optionally used if it is installed. All devices can be easily registered in the ACM repositories by running the *setup.cmd* file located in the root folder of the library.

Individual HSL4 files are provided as an alternative to running the setup.cmd to allow users to manually register specific implementation objects. Each object has two files - an Asset Control file and a Device file. The Asset Control files include attachments of all required files for that object. The Device files are used to actually add that device into a Studio 5000 project and these reference the Asset Control files.

All Studio 5000® Application Code Manager files can be found in the /*ApplicationCodeManagerLibraries*/ folder of the library. The files included are as follows:

Implementation Object	Asset Control File (.HSL4)	Device File (.HSL4)
TSSM	(RA-LIB)_Device_Asset-Control_Safety_raC_Dvc_TSSM_(1.2)	(RA-LIB)_Device_Device_Safety_raC_LD_TSSM_(1.2)

Operations

The Safety Discrete objects provides only physical operation mode. There is no virtual device mode offered.

Execution

The following table explains the handling of instruction execution conditions.

Condition	Description
EnableIn False (false rung)	Processing for EnableIn False (false rung) is handled the same as if the device were taken out of service by Command. The device outputs are de-energized and the device is shown as Program Out of Service on the HMI. All alarms are cleared.
Powerup (prescan, first scan)	On prescan, any commands that are received before first scan are discarded. The device is de-energized. On first scan, the device is treated as if it were returning from Hand command source: the instruction state is set based on the position feedback that is received from the device. If the feedback is valid for one position, the device is set to that position. If the device does not have position feedback or the position feedback is invalid, the device is set to the 'unknown/powerup' state. The command source is set to its default, either Operator or Program (unlocked).
Postscan	No SFC Postscan logic is provided.

Add-On Instruction I/O Data

InOut Data

InOut	Function / Description	DataType
Inp_TSSM	Input Interface - TSSM Instruction	DCI_MUTING_TWO_SENSOR_SYM
Inp_DCS	Input Interface - DCS Instruction	DCI_STOP
Out_Ctrl_Inf	Information Interface	raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	Status Interface	raC_UDT_Itf_SafetyDevice_Sts

Input Data

Input	Function/Description	DataType
EnableIn	Enable Input - System Defined Parameter	BOOL
Inp_Suspended	Suspend device operation	BOOL

Output Data

Output	Function/Descritpion	DataType
Sts_DeviceSuspended	Device is suspended	BOOL
Sts_SafetyDemand	Safety demand is present	BOOL
Sts_ResetRequired	Reset is required	BOOL
Sts_DiagnosticPresent	Diagnostic is present	BOOL
Sts_FaultPresent	Fault is present	BOOL
Sts_MuteLampON	This output indicates the status of the light curtain's protective function. ON (1): The light curtain's protective function is disabled. OFF (0): The light curtain's protective function is enabled.	BOOL
Sts_ClearArea	This output indicates when the light curtain sensing field must be cleared (all muting sensors and the light curtain are ON) before processing can continue. ON (1): The light curtain sensing field must be cleared. OFF (0): The light curtain sensing field is clear.	BOOL
Sts_OAClearingLCA	The light curtain sensing field is not obstructed but the field must be cleared and override is active. ON(1): Override Activate and light Curtain area in clearing state. OFF (0): Light curtain area not in clearing state	BOOL
Sts_OAClearedLCA	The light curtain sensing field is not obstructed, the field is cleared and override is active. ON (1): Override Activate and light Curtain area cleared OFF (0): Light curtain area not cleared	BOOL
Sts_Muted	Muted status ON (1): Device is muted and protective function is disabled OFF (0): Device is not muted and protective function is enabled.	BOOL
Sts_Ready	Device is ready	BOOL
Sts_ERR	Instruction Error Code	DINT
Sts_eReady	0: Ready 1: Safety Demand 2: Reset Required 3: Suspended	SINT
raC_Dvc_TSSM	Unique Parameter Name for auto - discovery	BOOL
EnableOut	Enable Output - System Defined Parameter	BOOL

Data Types

The following Safety Common Control Interface tags are the primary device program tags to read and write to when interfacing to safety devices. The value of using these tags in your specific application code is that you may use a number of different safety devices such as TSSM, TSAM, etc without having to update your application device interface tags. The primary function of these interfaces within the Safety Device Library is to pass individual instruction information to the Safety Group Monitor instruction.

Refer to the Interfaces section for detailed information on interfaces.

raC_UDT_Itf_SafetyDevice_Inf

Member	Description	Data Type
Type	Provides device object type information e.g. raC_Dvc_SafetyGroupMonitor, raC_Dvc_TSSM, raC_Dvc_TSSMTL, etc.	STR0032
DisplayName	Provides object display name	STR0024
eClass	Provide instruction type information eClass ID: Instruction Type 0: DCS 1: DCSTL 2: DCSTM 3: CROUT 5: SafetyGroupMonitor 6: TSAM 7: TSSM 8: FSBM 9: THRSe 10: SMAT 11:SS2 12: SS1 13:SFX 14:STO	DINT

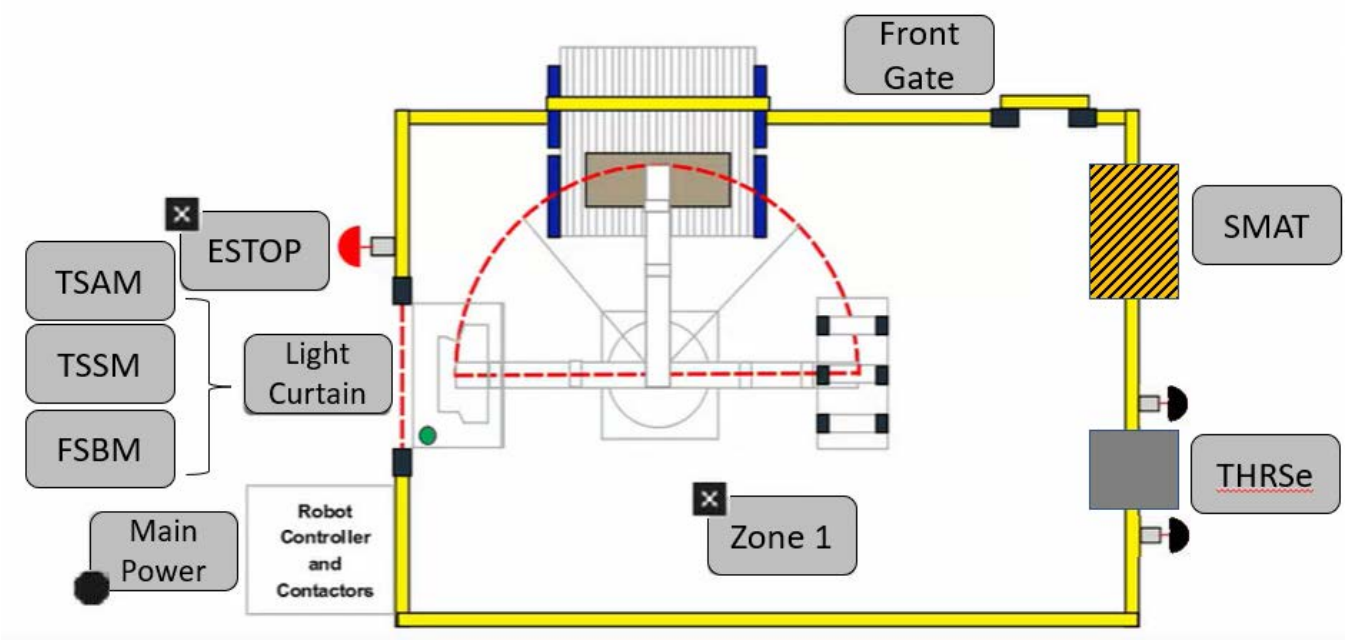
raC_UDT_Itf_SafetyDevice_Sts

This is the Safety Device Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data. The table below shows member names, descriptions, and tag data types.

Input	Description	Data Type
bSts	Bit overlay (Visible) covering all subsequent Boolean members. (Unused)	DINT
DeviceSuspended	Device suspended status (Inp_Suspend = 1)	BOOL
SafetyDemand	Safety demand status ON (1): Safety demand is present on the device object OFF (0): No safety demand	BOOL
ResetRequired	Reset Required status ON (1): Device requires reset signal OFF (0): No reset required	BOOL
DiagnosticPresent	Diagnostic status ON (1): A diagnostic is present in the device object OFF (0): Device is operating normally	BOOL
FaultPresent	Fault Status ON (1): A fault is present in the device object OFF (0): Device is operating normally	BOOL
Muted	Device Muted status (Available for TSSM) ON (1): Muted status is present on one or more safety instructions within a specific safety group OFF (0): None of the instructions within group is muted	BOOL

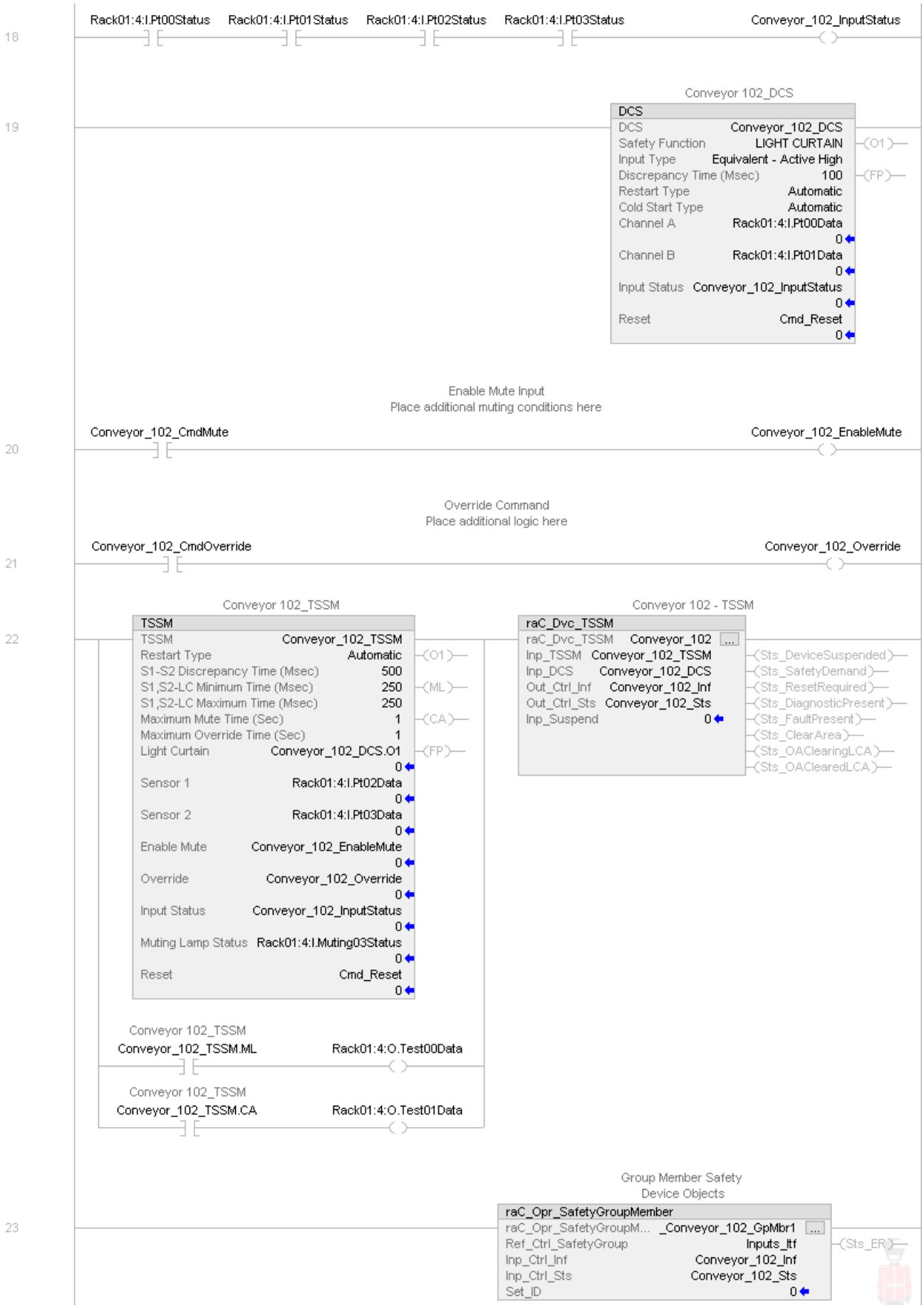
Programming Example

Fully configured device routine is provided below for reference. This example is taken from the *Safety Program - Inputs* routine in the *SafetytApplication.ACD* example application provided in the library.



This example uses the TSSM instruction for a light curtain function shown as *TSSM* in the safe machine diagram. A DCS instruction is also used for the light curtain. This system in the example application code is referred to as Conveyor_102. DCS Channel A and Channel B are tied to safety class discrete inputs. TSSM Sensors 1 and 2 as well as muting lamp status are also connected to safety class discrete inputs. Rung descriptions are as follows:

- Rung 18: Check input statuses
- Rung 19: Configure DCS instruction
- Rung 20: Check mute input status
- Rung 21: Check Override status
- Rung 22: Configure TSSM and raC_Dvc_TSSM instructions
- Rung 23: raC_Opr_SafetyGroupMember (optional) assigns Conveyor_102 TSSM to Inputs Safety Group at ID #0.


18
Rack01:4:I.Pt00Status
Rack01:4:I.Pt01Status
Rack01:4:I.Pt02Status
Rack01:4:I.Pt03Status
Conveyor_102_InputStatus
19
Conveyor 102_DCS
DCS
DCS Conveyor_102_DCS
Safety Function LIGHT CURTAIN
Input Type Equivalent - Active High
Discrepancy Time (Msec) 100
Restart Type Automatic
Cold Start Type Automatic
Channel A Rack01:4:I.Pt00Data
0
Channel B Rack01:4:I.Pt01Data
0
Input Status Conveyor_102_InputStatus
0
Reset Cmd_Reset
0
O1
FP
Enable Mute Input
Place additional muting conditions here
20
Conveyor_102_CmdMute
Conveyor_102_EnableMute
Override Command
Place additional logic here
21
Conveyor_102_CmdOverride
Conveyor_102_Override
22
Conveyor 102_TSSM
TSSM
TSSM Conveyor_102_TSSM
Restart Type Automatic
S1-S2 Discrepancy Time (Msec) 500
S1,S2-LC Minimum Time (Msec) 250
S1,S2-LC Maximum Time (Msec) 250
Maximum Mute Time (Sec) 1
Maximum Override Time (Sec) 1
Light Curtain Conveyor_102_DCS.O1
0
Sensor 1 Rack01:4:I.Pt02Data
0
Sensor 2 Rack01:4:I.Pt03Data
0
Enable Mute Conveyor_102_EnableMute
0
Override Conveyor_102_Override
0
Input Status Conveyor_102_InputStatus
0
Muting Lamp Status Rack01:4:I.Muting03Status
0
Reset Cmd_Reset
0
O1
ML
CA
FP
Conveyor 102 - TSSM
raC_Dvc_TSSM
raC_Dvc_TSSM Conveyor_102
Inp_TSSM Conveyor_102_TSSM
Inp_DCS Conveyor_102_DCS
Out_Ctrl_Inf Conveyor_102_Inf
Out_Ctrl_Sts Conveyor_102_Sts
Inp_Suspend 0
Sts_DeviceSuspended
Sts_SafetyDemand
Sts_ResetRequired
Sts_DiagnosticPresent
Sts_FaultPresent
Sts_ClearArea
Sts_OAClearingLCA
Sts_OAClearedLCA
Conveyor 102_TSSM
Conveyor_102_TSSM.ML
Rack01:4:O.Test00Data
Conveyor 102_TSSM
Conveyor_102_TSSM.CA
Rack01:4:O.Test01Data
23
Group Member Safety
Device Objects
raC_Opr_SafetyGroupMember
raC_Opr_SafetyGroupM... _Conveyor_102_GpMbr1
Ref_Ctrl_SafetyGroup Inputs_Itf
Inp_Ctrl_Inf Conveyor_102_Inf
Inp_Ctrl_Sts Conveyor_102_Sts
Set_ID 0
Sts_ER

Note that this programming example is the same code that is imported when either importing the supplied ROUTINE .L5X files or when using Application Code Manager or the Studio 5000 Import Library Objects wizard plug-in.

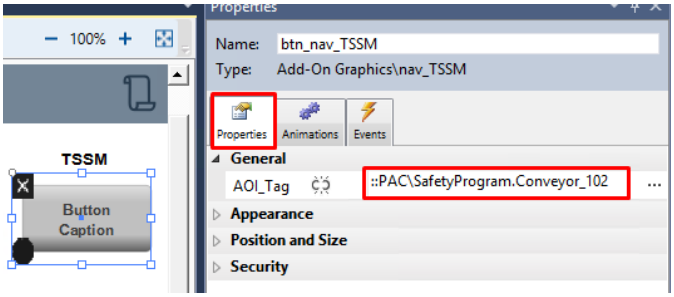
Graphic Symbols

Graphic Symbols are used as launch buttons within HMI applications to open up faceplate displays. See Basic Launch Button Attributes section for details on configuration and indicators

FactoryTalk® View ME/SE Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Global Object Parameter Values
btn_nav_safety		Faceplate navigation button with string tag label. This launch button graphic object allows the user to navigate to the device object faceplate. The text on the button face is set to the tag's .Inf_DisplayName.	#101: AOI Instance (e.g. {::[PAC]Program::SafetyProgram._InstanceName})

Studio 5000 View Designer® Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Property Configuration
Launch		The supplied launch button in View Designer is used to navigate to the faceplate in a user application.	

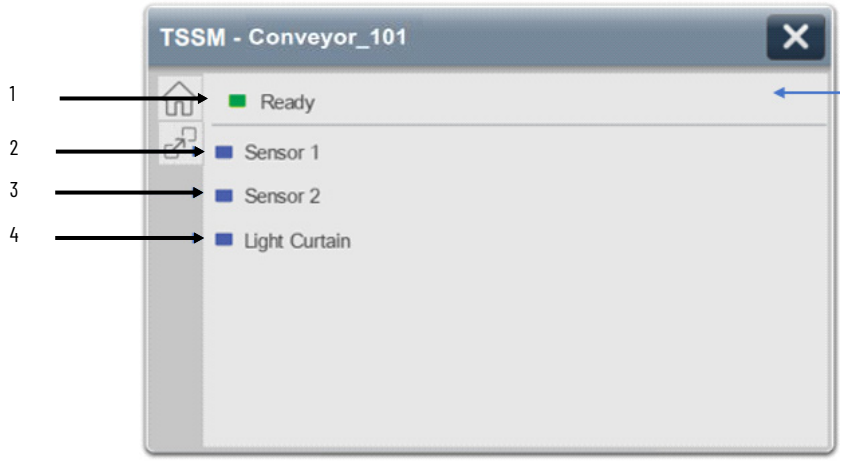
Faceplates

There are basic faceplate attributes that are common across all instructions. See Basic Faceplate Attributes on page 27.

Home

The Home tab is the main tab of the faceplate. It provides the status of the safety input device along with configuration information.

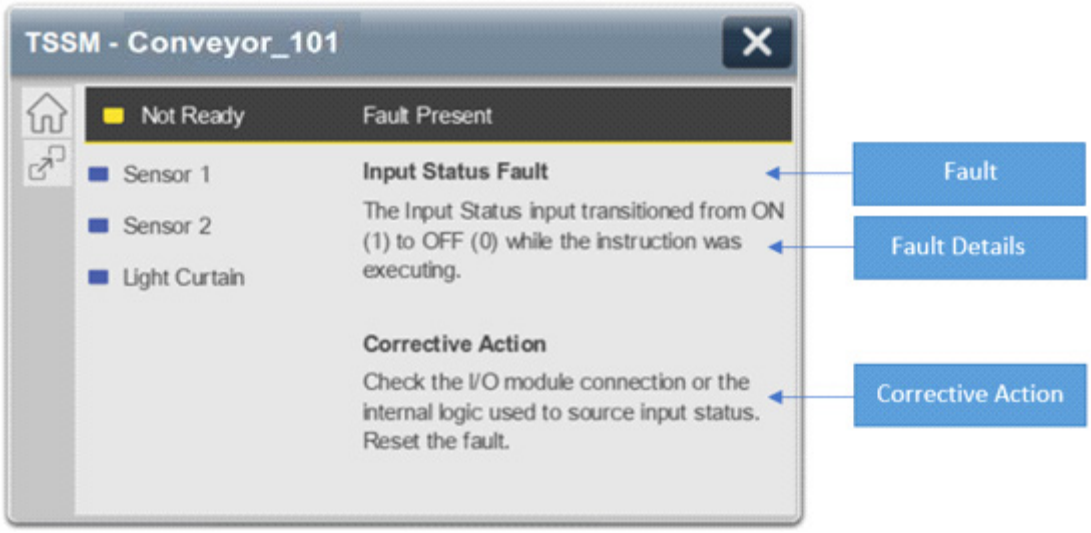
The Home tab shows the Sensor 1, Sensor 2 and Light Curtain cleared status. Additionally there is a Ready banner at the top showing the status of the device.



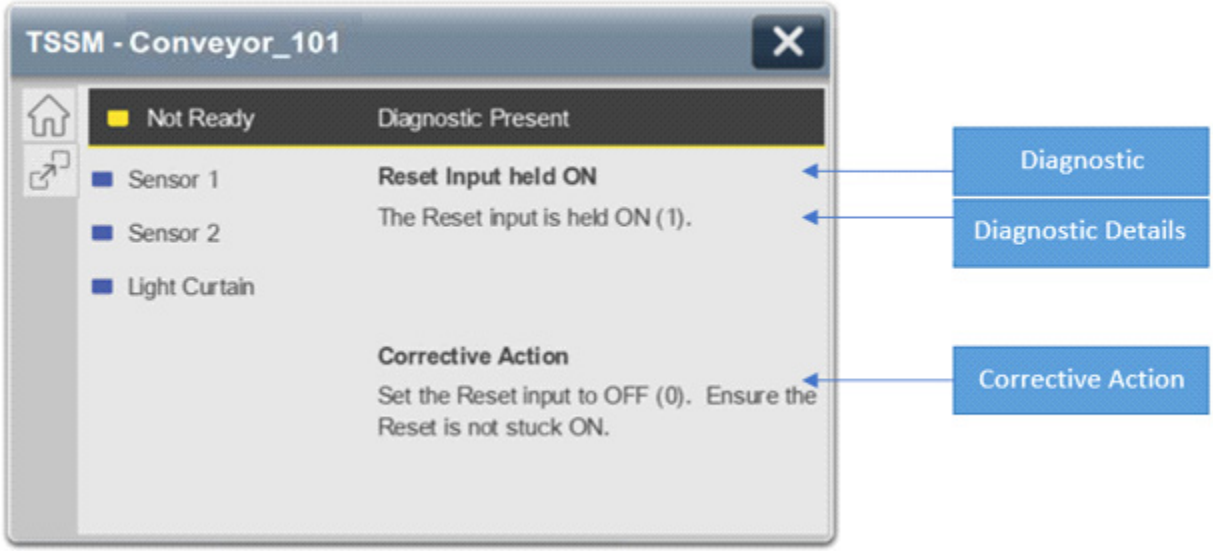
Item	Description
1	Banner- Ready Status
2	Sensor 1 Status. This is one of the two muting sensors; Sensor 1 must be blocked or cleared within the S1S2 Discrepancy Time of Sensor 2 being blocked or cleared and its status is indicated by: OFF (0) = Gray LED. Sensor 1 is blocked. ON (1) = Blue LED. Sensor 1 is cleared.
3	Sensor 2 Status. This is one of the two muting sensors; Sensor 2 must be blocked or cleared within the S1S2 Discrepancy Time of Sensor 1 being blocked or cleared and its status is indicated by: OFF (0) = Gray LED. Sensor 2 is blocked. ON (1) = Blue LED. Sensor 2 is cleared.
4	Light Curtain Status. This is an input channel which represents the current state of the physical light curtain. The Light Curtain must be blocked or cleared within the S1, S2-LC minimum or maximum time of light curtain being blocked or cleared and its status is indicated by: OFF (0) = Gray LED. Light Curtain is blocked ON (1) = Blue LED. Light Curtain is cleared

Home tab screen also provides fault and diagnostic information with required corrective actions on the right plane of the faceplate

When there is fault present, banner with black background is displayed on top of home screen with **Not Ready** and **Fault Present** text.

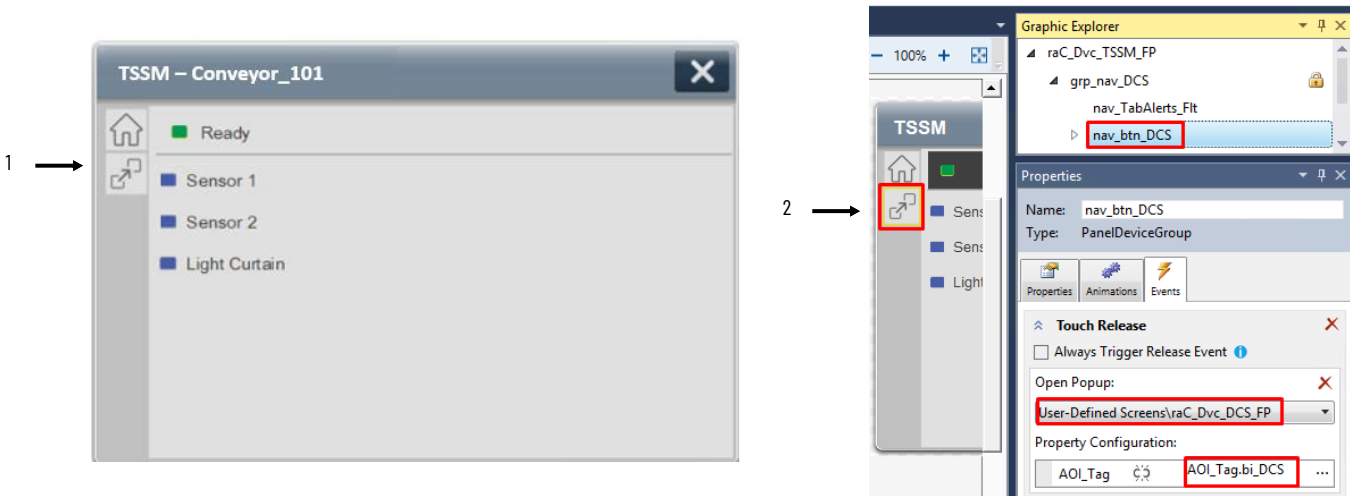


When there is diagnostic present, banner with black background is displayed on top of home screen with **Not Ready** and **Diagnostic Present** text.



Extensions - DCS

Below the Home Tab button there is an Extension button which launches the associated Dual Channel Input Stop (DCS) faceplate. See the section on raC_Dvc_DCS for complete details on the DCS object. Click the 'X' in the top right to close the DCS faceplate and return to the original device faceplate.



Item	Description
1	DCS Extension Navigation Tab Button
2	DCS Extension Navigation Configuration settings

Application Code Manager

Safety Device Library objects can be set-up and configured using Studio 5000® Application Code Manager.

Refer to the section Using Application Code Manager for complete details.

Definition Object: raC_Dvc_TSSM

This object contains the AOI definition and used as linked library to implement object. This gives flexibility to choose to instantiate only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Implementation Object: raC_LD_Dvc_TSSM

Parameter Name	Default Value	Instance Name	Definition	Description
RoutineName	{ObjectName}	{RoutineName}	Routine	Enter Routine name. Routine will be created and Object implement rung(s) inserted. A JSR will be inserted in MainRoutine. If routine name already exists, then object will be inserted into existing routine. By default, parameter is set to Object Name.
TagName	{ObjectName}	{TagName}	Backing Tag	Enter the backing tag of the main AOI. This will serve as the base tag name for other tags in this object that are derived from the base.
TagDescription	{ObjectDescription}	{TagDescription}		Tag Description of the main AOI backing tag
DisplayName	[ObjectName]	[DisplayName]		Faceplate title/group event list description. Please avoid underscore ("_") at the end.
SafetyFunction	LIGHT CURTAIN			This parameter provides a text name for how this instruction is being used. It is for information/documentation purposes only. Options: - USER DEFINED - EMERGENCY STOP - SAFETY GATE - LIGHT CURTAIN - AREA SCANNER - SAFETY MAT - CABLE (ROPE) PULL SWITCH
InputType	Equivalent - Active High			This parameter selects input channel behavior. Equivalent - Active High or Complimentary.
DiscrepencyTime	100			The amount of time that the inputs can be in an inconsistent state before an instruction fault is generated. Range 5-3000ms.
RestartType_DCS	Automatic			This input configures Output 1 for either Manual or Automatic Restart.
ColdStartType_DCS	Automatic			This parameter specifies the Output 1 behavior when applying controller power or mode change to Run. Automatic or Manual.
ChannelA	ChA		Input Parameter	Enter the input address (data) tag for Channel A from safety digital input module.

Parameter Name	Default Value	Instance Name	Definition	Description
ChannelA_InputStatus	ChA_Sts		Input Parameter	Enter the input address (Status) tag for Channel A from safety digital input module.
ChannelB	ChB		Input Parameter	Enter the input address (data) tag for Channel B from safety digital input module.
ChannelB_InputStatus	ChB_Sts		Input Parameter	Enter the input address (Status) tag for Channel B from safety digital input module.
RestartType	Automatic			This input configures Output 1 for either Manual of Automatic Restart
S1S2Time	0			The maximum amount of time allowed between clearing or blocking of the muting sensor inputs Sensor 1 and Sensor 2 before generating a fault. Setting this input to 0 disables the S1-S2 timer. The valid range is 5-180,000 ms.
S1LCTime	0			The maximum amount of time allowed between clearing or blocking of the Sensor 2 muting sensor and the Light Curtain before generating a fault. Setting this input to 0 disables the S2-LC timer. The valid range is 5-180,000 ms.
MaxMuteTime	0			The maximum amount of time during which the instruction lets the protective function of the light curtain be disabled before generating a fault. Setting this input to 0 disables the Maximum Mute timer. The valid range is 0-3,600,000 ms.
MaxOverrideTime	0			The maximum amount of time that the instruction lets the override feature energize the Output 1 output. Setting this input to 0 disables the Maximum Override timer. The valid range is 0-30,000 ms.
Sensor1	Sens1		Input Parameter	Enter the input address (data) tag for Sensor 1 from safety digital input module. That is one of two muting sensors, Sensor 1 must be the first sensor to be blocked and the last to be cleared in the muting sequence.
Sensor1_InputStatus	Sense1_Sts		Input Parameter	Enter the input address (Status) tag for Sensor 1 from safety digital input module.
Sensor2	Sens2		Input Parameter	Enter the input address (data) tag for Sensor 2 from safety digital input module. That is one of two muting sensors, Sensor 2 must be the second sensor to be blocked and the first to be cleared in the muting sequence.
Sensor2_InputStatus	Sense2_Sts		Input Parameter	Enter the input address (Status) tag for Sensor 2 from safety digital input module.
MutingLampStatus	MuteInput_Sts		Input Parameter	Enter the input address tag showing status of the muting lamp.
MutingLamp_Output	ML_Output		Input Parameter	Enter the test output address (Data) tag to drive the Muting Lamp from safety digital input module.
ClearArea_Output	CA_Output		Input Parameter	Enter the test output address (Data) tag to drive the Clear Area Lamp from safety digital input module.
ResetTag	Cmd_Reset		Controller Tag	Enter Tagname for Reset signal. Please avoid underscore ("_") at the end.

Note that if the tag names are manually entered or not linked to input channel tags a red 'X' will be shown beside the parameter. This is acceptable and the program can still be generated.

Linked Libraries

Link Name	Catalog Number	Revision	Solution	Category
raC_Dvc_TSSM	raC_Dvc_TSSM	1	(RA-LIB) Device	Safety
raC_Opr_SafetyGroupMember	raC_Opr_SafetyGroupMember	1	(RA-LIB) Device	Safety
raC_Dvc_DCS	raC_Dvc_DCS	1	(RA-LIB) Device	Safety

Configured HMI Content

HMI Content	Instance Name	Description
Launch Button	{ObjectName}_GO_LaunchFP	Global Object configured callout instance

Attachments

Name	Description	File Name	Extraction Path
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols SE	(raC-1-SE) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewSE\Global Objects
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols ME	(raC-1-ME) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewME\Global Objects
V1_raC_Dvc_TSSM	Faceplate SE	(raC-1_xx-SE) raC_Dvc_TSSM-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
V1_raC_Dvc_TSSM	Faceplate ME	(raC-1_xx-ME) raC_Dvc_TSSM-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
V1_raC_Dvc_TSSM	View Designer	(raC-1_xx-VD) raC_Dvc_Safety.vpd	{ProjectName}\Visualization\ViewDesigner
V1_RM_raC_DvcSafety	Reference Manual	DEVICE-RM500C-EN-P.pdf	{ProjectName}\Documentation
V1_Safety_Images	HMI Image Set	Safety_Images.zip	{ProjectName}\Visualization\Images

FSBM - Four Sensor Bi-directional Muting (raC_Dvc_FSBM)

Overview

The Four Sensor Bi-directional Muting device object (raC_Dvc_FSBM) includes a faceplate which displays status and configuration information of safety applications using four muting sensors arranged sequentially before and after the light curtain's center of the protected opening. The primary function is to automatically disable the protective function of a light curtain which allows material to be transported through the light curtain sensing field without stopping the machine.

The instruction is ready when material clears Sensor 1, Sensor 2, Light Curtain, Sensor 3, and Sensor 4 and are in the active state as determined by the Input type operand, and the correct reset actions are carried out.

The FSBM instruction also requires a DCS (Dual Channel Input Stop) instruction.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following videos for this section:
"Operational_Overview_of_Safety_Device_Object_Faceplates"
"Operational_Overview_of_Safety_Application_Example_in_LogixDesigner"

Functional Description

The Four Sensor Bi-directional Muting pre-configured Device Objects:

- Collect, Process and Deliver Data between Smart Devices and Application Logic
- Provide Device Status & Diagnostics Faceplates for Machine Startup, Commissioning, Operations, and Maintenance
- Include Common HMI Faceplate Components, Device States, and Control Interfaces providing Application Development and Operation Efficiency

All these features provide quick feedback, shorten recovery time, and simplify implementation.

Required Files

Device Objects include Add-On Instructions (AOIs) and HMI faceplates. The revision number (e.g. 1.01) used in filenames can change as new revisions are created.

Controller Files

Add-On Instructions are reusable code objects that contain encapsulated logic that can streamline implementing your system. This lets you create your own instruction set for programming logic as a supplement to the instruction set provided natively in the ControlLogix® firmware. An Add-On Instruction is defined once in each controller project, and can be instantiated multiple times in your application code as needed.

The Add-On Instruction must be imported into the controller project to be used in the controller configuration. These can be imported as Add-On Instruction files, or as part of the ROUTINE Import or Import Library Objects wizard.

All Add-On Instruction and ROUTINE Import files can be found in the */Studio 5000 Logix Designer® Files - L5X/* folder in the library.

Device/Item	Add-On Instruction	ROUTINE Import
FSBM	raC_Dvc_FSBM_1.02_AOI.L5X	raC_Dvc_FSBM_1.02_ROUTINE.L5X

FactoryTalk® View HMI Files

FactoryTalk® View ME or SE applications require importing the desired device faceplates in addition to all Global Object (ggfx) files and all images located in the */HMI FactoryTalk View Images - png/* folder of the library. FactoryTalk® View ME files are stored in the */HMI - FactoryTalk View ME/* library folder and FactoryTalk® View SE files are stored in the */HMI - FactoryTalk View SE/* library folder.

Device/Item	Type	FactoryTalk® View ME Faceplate	FactoryTalk® View SE Faceplate
FSBM	Display	(raC-1_02-ME) raC_Dvc_FSBM-Faceplate.gfx	(raC-1_02-SE) raC_Dvc_FSBM-Faceplate.gfx
Graphic Symbols	Global Object	(raC-1-ME) Graphic Symbols - Safety Device	(raC-1-SE) Graphic Symbols - Safety Device.ggfx
Toolbox	Global Object	(raC-1-ME) Toolbox - Safety Device.ggfx	(raC-1-SE) Toolbox - Safety Device.ggfx

Studio 5000 View Designer® HMI Files

All Studio 5000 View Designer® Files can be found in the */HMI - ViewDesigner - vpd/* folder of the library.

Device/Item	Studio 5000 View Designer® Faceplate
FSBM	(raC-1_02-VD) raC_Dvc_Safety.vpd

Studio 5000® Application Code Manager Files

Studio 5000® Application Code Manager (ACM) can be optionally used if it is installed. All devices can be easily registered in the ACM repositories by running the *setup.cmd* file located in the root folder of the library.

Individual HSL4 files are provided as an alternative to running the setup.cmd to allow users to manually register specific implementation objects. Each object has two files - an Asset Control file and a Device file. The Asset Control files include attachments of all required files for that object. The Device files are used to actually add that device into a Studio 5000 project and these reference the Asset Control files.

All Studio 5000® Application Code Manager files can be found in the /*ApplicationCodeManagerLibraries*/ folder of the library. The files included are as follows:

Implementation Object	Asset Control File (.HSL4)	Device File (.HSL4)
FSBM	(RA-LIB)_Device_Asset-Control_Safety_raC_Dvc_FSBM_(1.2)	(RA-LIB)_Device_Device_Safety_raC_LD_FSBM_(1.2)

Operations

The Safety Discrete objects provides only physical operation mode. There is no virtual device mode offered.

Execution

The following table explains the handling of instruction execution conditions.

Condition	Description
EnableIn False (false rung)	Processing for EnableIn False (false rung) is handled the same as if the device were taken out of service by Command. The device outputs are de-energized and the device is shown as Program Out of Service on the HMI. All alarms are cleared.
Powerup (prescan, first scan)	On prescan, any commands that are received before first scan are discarded. The device is de-energized. On first scan, the device is treated as if it were returning from Hand command source: the instruction state is set based on the position feedback that is received from the device. If the feedback is valid for one position, the device is set to that position. If the device does not have position feedback or the position feedback is invalid, the device is set to the 'unknown/powerup' state. The command source is set to its default, either Operator or Program (unlocked).
Postscan	No SFC Postscan logic is provided.

Add-On Instruction I/O Data

InOut Data

InOut	Function / Description	DataType
Inp_FSBM	Input Interface - FSBM Instruction	DCI_MUTING_TWO_SENSOR_ASYM
Inp_DCS	Input Interface - DCS Instruction	DCI_STOP
Out_Ctrl_Inf	Information Interface	raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	Status Interface	raC_UDT_Itf_SafetyDevice_Sts

Input Data

Input	Function/Description	DataType
EnableIn	Enable Input - System Defined Parameter	BOOL
Inp_Suspended	Suspend device operation	BOOL

Output Data

Output	Function/Descritpion	DataType
Sts_DeviceSuspended	Device is suspended	BOOL
Sts_SafetyDemand	Safety demand is present	BOOL
Sts_ResetRequired	Reset is required	BOOL
Sts_DiagnosticPresent	Diagnostic is present	BOOL
Sts_FaultPresent	Fault is present	BOOL
Sts_MuteLampON	This output indicates the status of the light curtain's protective function. ON (1): The light curtain's protective function is disabled. OFF (0): The light curtain's protective function is enabled.	BOOL
Sts_ClearArea	This output indicates when the light curtain sensing field must be cleared (all muting sensors and the light curtain are ON) before processing can continue. ON (1): The light curtain sensing field must be cleared. OFF (0): The light curtain sensing field is clear.	BOOL
Sts_OAClearingLCA	The light curtain sensing field is not obstructed but the field must be cleared and override is active. ON(1): Override Activate and light Curtain area in clearing state. OFF (0): Light curtain area not in clearing state	BOOL
Sts_OAClearedLCA	The light curtain sensing field is not obstructed, the field is cleared and override is active. ON (1): Override Activate and light Curtain area cleared OFF (0): Light curtain area not cleared	BOOL
Sts_Muted	Muted status ON (1): Device is muted and protective function is disabled OFF (0): Device is not muted and protective function is enabled.	BOOL
Sts_Ready	Device is ready	BOOL
Sts_ERR	Instruction Error Code	DINT
Sts_eReady	0: Ready 1: Safety Demand 2: Reset Required 3: Suspended	SINT
raC_Dvc_FSBM	Unique Parameter Name for auto - discovery	BOOL
EnableOut	Enable Output - System Defined Parameter	BOOL

Data Types

The following Safety Common Control Interface tags are the primary device program tags to read and write to when interfacing to safety devices. The value of using these tags in your specific application code is that you may use a number of different safety devices such as FSBM, TSSM, etc without having to update your application device interface tags. The primary function of these interfaces within the Safety Device Library is to pass individual instruction information to the Safety Group Monitor instruction.

Refer to the Interfaces section for detailed information on interfaces.

raC_UDT_Itf_SafetyDevice_Inf

Member	Description	Data Type
Type	Provides device object type information e.g. raC_Dvc_SafetyGroupMonitor, raC_Dvc_FSBM, raC_Dvc_FSBMTL, etc.	STR0032
DisplayName	Provides object display name	STR0024
eClass	Provide instruction type information eClass ID: Instruction Type 0: DCS 1: DCSTL 2: DCSTM 3: CROUT 5: SafetyGroupMonitor 6: TSAM 7: TSSM 8: FSBM 9: THRSe 10: SMAT 11:SS2 12: SS1 13:SFX 14:STO	DINT

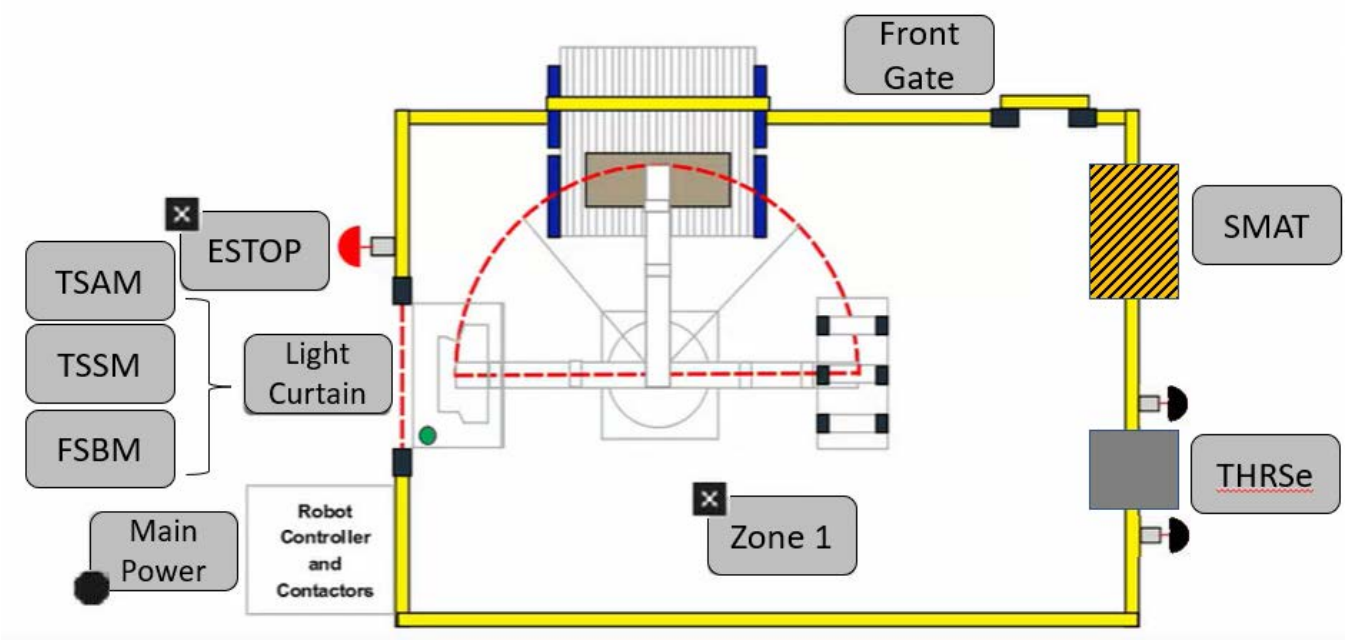
raC_UDT_Itf_SafetyDevice_Sts

This is the Safety Device Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data. The table below shows member names, descriptions, and tag data types.

Input	Description	Data Type
bSts	Bit overlay (Visible) covering all subsequent Boolean members. (Unused)	DINT
DeviceSuspended	Device suspended status (Inp_Suspend = 1)	BOOL
SafetyDemand	Safety demand status ON (1): Safety demand is present on the device object OFF (0): No safety demand	BOOL
ResetRequired	Reset Required status ON (1): Device requires reset signal OFF (0): No reset required	BOOL
DiagnosticPresent	Diagnostic status ON (1): A diagnostic is present in the device object OFF (0): Device is operating normally	BOOL
FaultPresent	Fault Status ON (1): A fault is present in the device object OFF (0): Device is operating normally	BOOL
Muted	Device Muted status (Available for FSBMTM) ON (1): Muted status is present on one or more safety instructions within a specific safety group OFF (0): None of the instructions within group is muted	BOOL

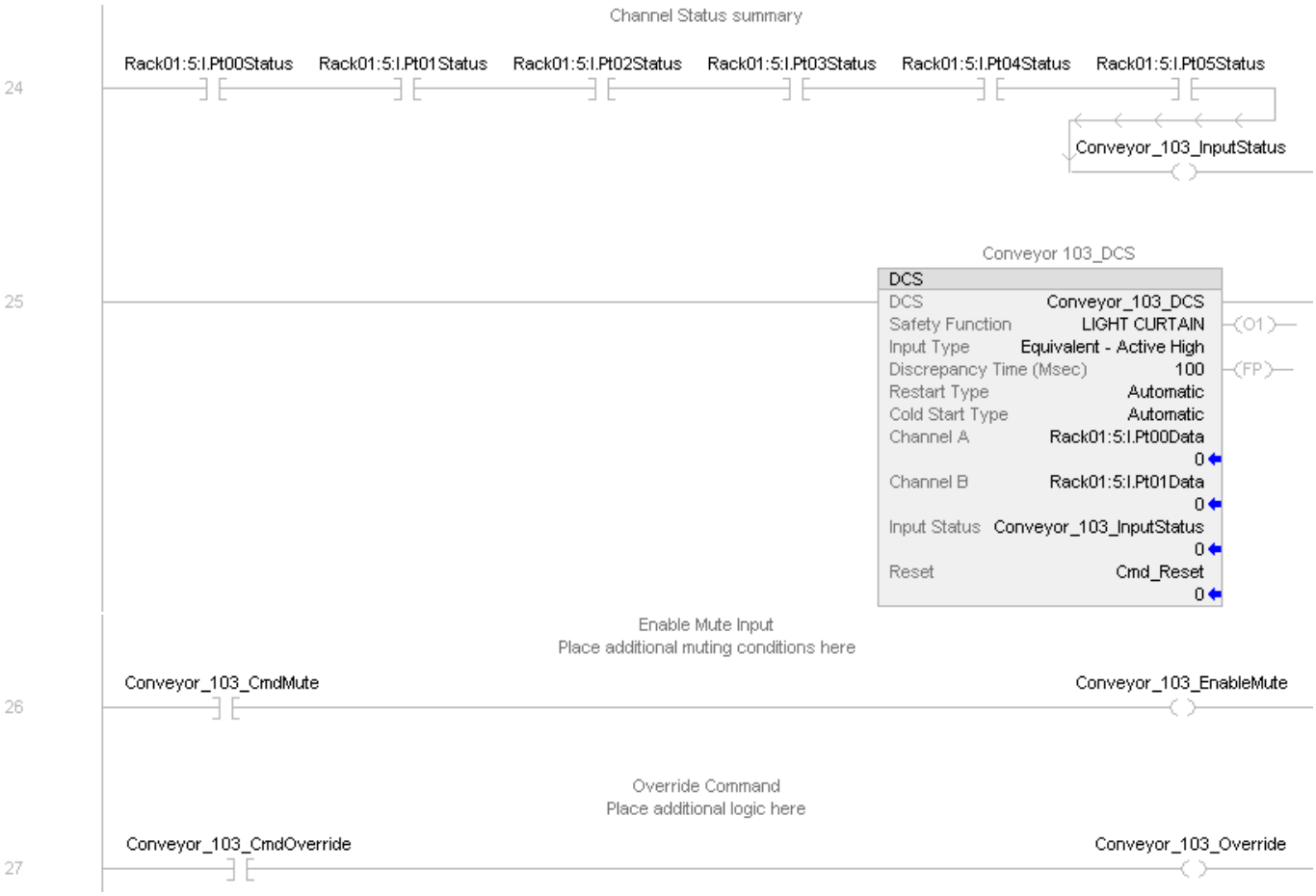
Programming Example

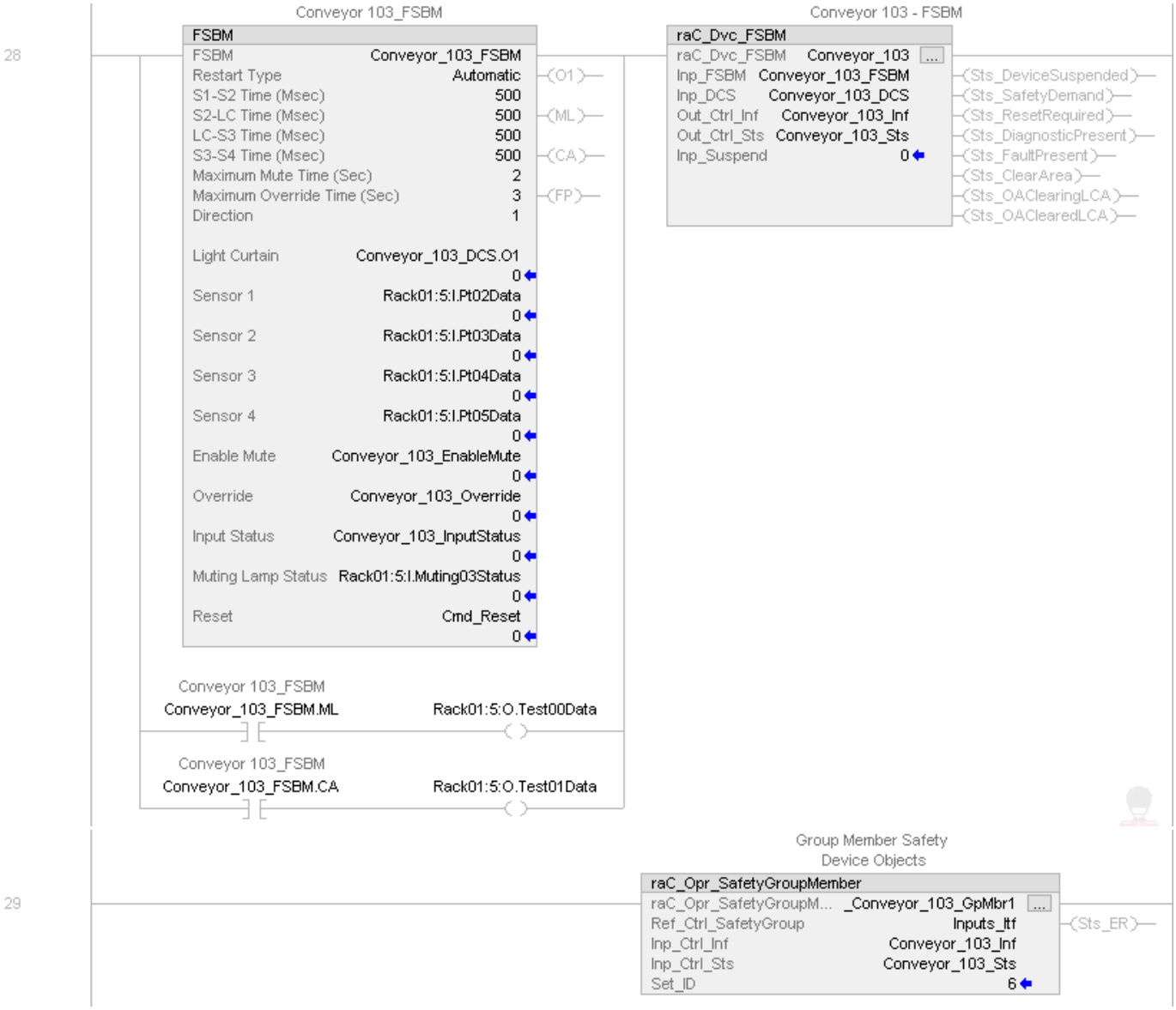
Fully configured device routine is provided below for reference. This example is taken from the *Safety Program - Inputs* routine in the *SafetytApplication.ACD* example application provided in the library.



This example uses the FSBM instruction for a light curtain function shown as *FSBM* in the safe machine diagram. A DCS instruction is also used for the light curtain. This system in the example application code is referred to as Conveyor_103. DCS Channel A and Channel B are tied to safety class discrete inputs. FSBM Sensors 1, 2, 3, and 4, as well as muting lamp status are also connected to safety class discrete inputs. Rung descriptions are as follows:

- Rung 24: Check input statuses
- Rung 25: Configure DCS instruction
- Rung 26: Check mute input status
- Rung 27: Check Override status
- Rung 28: Configure FSBM and raC_Dvc_FSBM instructions
- Rung 29: raC_Opr_SafetyGroupMember (optional) assigns Conveyor_101 FSBM to Inputs Safety Group at ID #6.


Channel Status summary
24
Rack01:5:I.Pt00Status
Rack01:5:I.Pt01Status
Rack01:5:I.Pt02Status
Rack01:5:I.Pt03Status
Rack01:5:I.Pt04Status
Rack01:5:I.Pt05Status
Conveyor_103_InputStatus
Conveyor 103_DCS
25
DCS
DCS Conveyor_103_DCS
Safety Function LIGHT CURTAIN
Input Type Equivalent - Active High
Discrepancy Time (Msec) 100
Restart Type Automatic
Cold Start Type Automatic
Channel A Rack01:5:I.Pt00Data
0
Channel B Rack01:5:I.Pt01Data
0
Input Status Conveyor_103_InputStatus
0
Reset Cmd_Reset
0
O1
FP
Enable Mute Input
Place additional muting conditions here
26
Conveyor_103_CmdMute
Conveyor_103_EnableMute
Override Command
Place additional logic here
27
Conveyor_103_CmdOverride
Conveyor_103_Override



Note that this programming example is the same code that is imported when either importing the supplied ROUTINE .L5X files or when using Application Code Manager or the Studio 5000 Import Library Objects wizard plug-in.

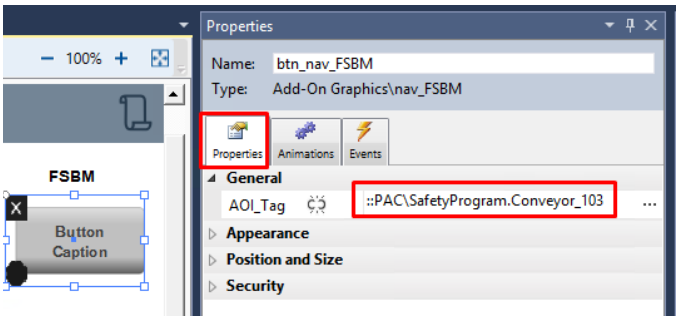
Graphic Symbols

Graphic Symbols are used as launch buttons within HMI applications to open up faceplate displays. See Basic Launch Button Attributes section for details on configuration and indicators

FactoryTalk® View ME/SE Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Global Object Parameter Values
btn_nav_safety		Faceplate navigation button with string tag label. This launch button graphic object allows the user to navigate to the device object faceplate. The text on the button face is set to the tag's .Inf_DisplayName.	#101: AOI Instance (e.g. {::[PAC]Program::SafetyProgram._InstanceName})

Studio 5000 View Designer® Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Property Configuration
Launch		The supplied launch button in View Designer is used to navigate to the faceplate in a user application.	

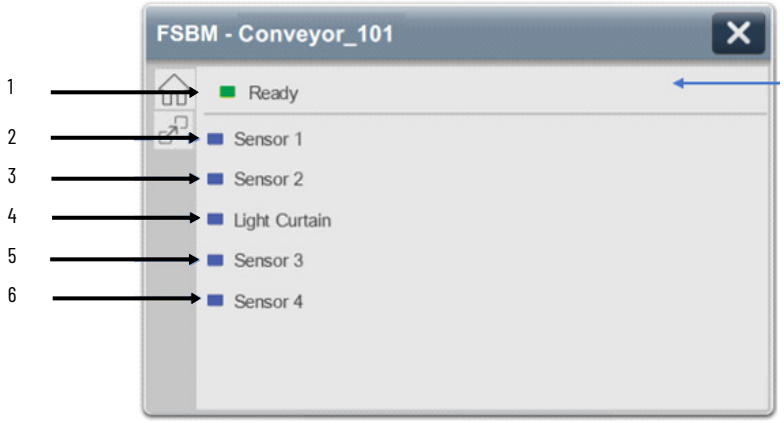
Faceplates

There are basic faceplate attributes that are common across all instructions. See Basic Faceplate Attributes on page 27.

Home

The Home tab is the main tab of the faceplate. It provides the status of the safety input device along with configuration information.

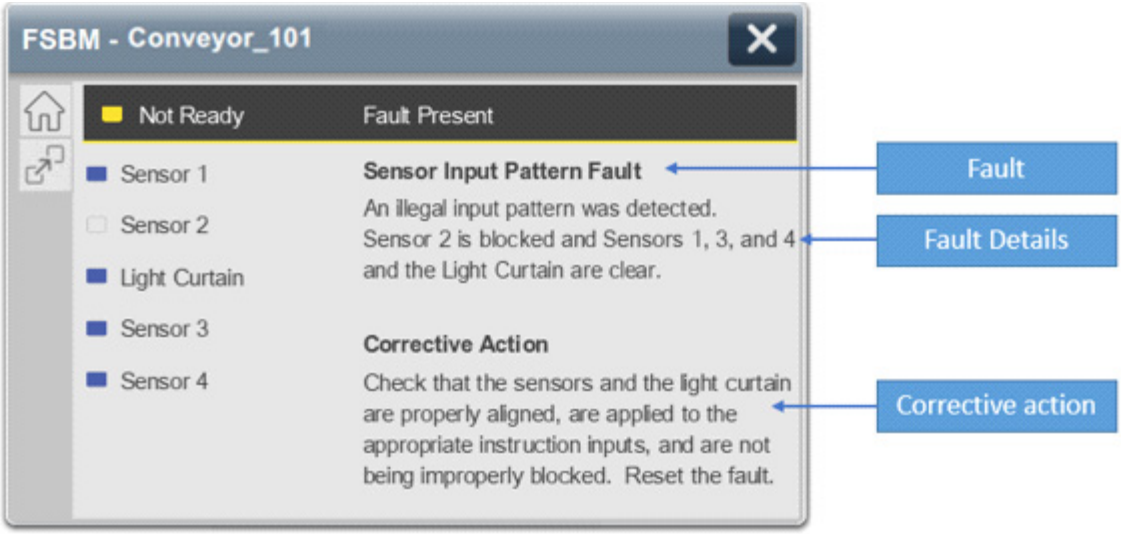
The Home tab shows the Sensor 1, Sensor 2, Sensor 3, Sensor 4, and Light Curtain cleared status. Additionally there is a Ready banner at the top showing the status of the device.



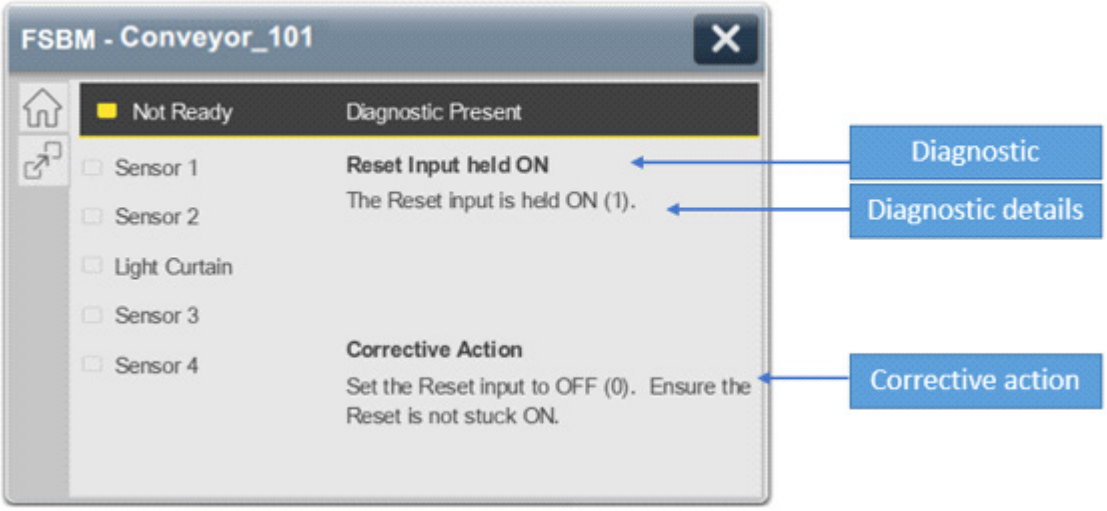
Item	Description
1	Banner- Ready Status
2	Sensor 1 Status. This is one of the two muting sensors; Sensor 1 must be the first sensor to be blocked and the last to be cleared in the muting sequence and its status is indicated by: OFF (0) = Gray LED. Sensor 1 is blocked. ON (1) = Blue LED. Sensor 1 is cleared.
3	Sensor 2 Status. This is one of the two muting sensors; Sensor 2 must be the second sensor to be blocked and the second last to be cleared in the muting sequence and its status is indicated by: OFF (0) = Gray LED. Sensor 2 is blocked. ON (1) = Blue LED. Sensor 2 is cleared.
4	Light Curtain Status. This is an input channel which represents the current state of the physical light curtain. Light curtain must be the third input to be blocked and third to be cleared in muting sequence and its status is indicated by: OFF (0) = Gray LED. Light Curtain is blocked ON (1) = Blue LED. Light Curtain is cleared
5	Sensor 3 Status. This is one of the two muting sensors; Sensor 3 must be the fourth sensor to be blocked and the second to be cleared in the muting sequence and its status is indicated by: OFF (0) = Gray LED. Sensor 3 is blocked. ON (1) = Blue LED. Sensor 3 is cleared.
6	Sensor 4 Status. This is one of the two muting sensors; Sensor 4 must be the last sensor to be blocked and the first to be cleared in the muting sequence and its status is indicated by: OFF (0) = Gray LED. Sensor 4 is blocked. ON (1) = Blue LED. Sensor 4 is cleared.

Home tab screen also provides fault and diagnostic information with required corrective actions on the right plane of the faceplate

When there is fault present, banner with black background is displayed on top of home screen with **Not Ready** and **Fault Present** text.

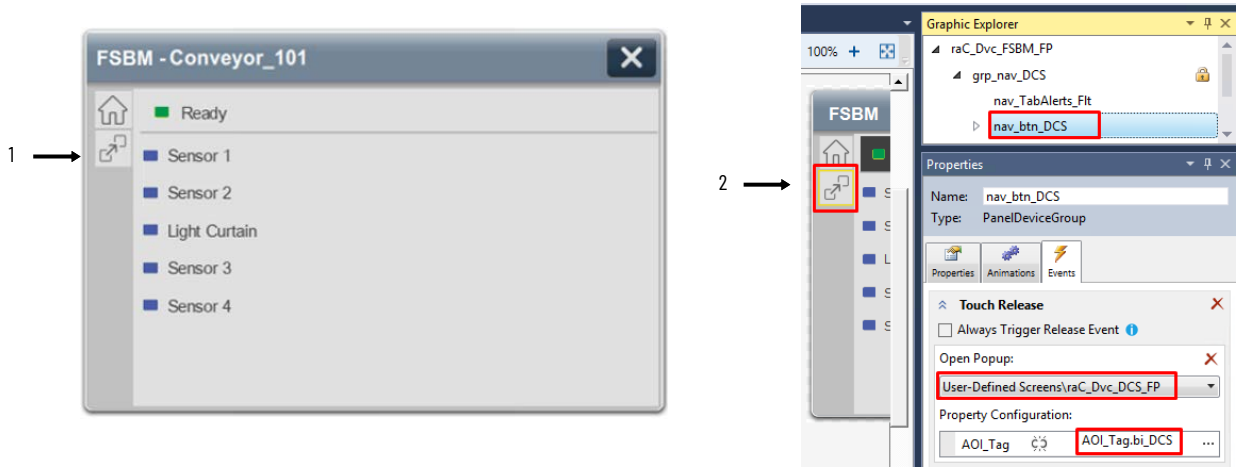


When there is diagnostic present, banner with black background is displayed on top of home screen with **Not Ready** and **Diagnostic Present** text.



Extensions - DCS

Below the Home Tab button there is an Extension button which launches the associated Dual Channel Input Stop (DCS) faceplate. See the section on raC_Dvc_DCS for complete details on the DCS object. Click the 'X' in the top right to close the DCS faceplate and return to the original device faceplate.



Item	Description
1	DCS Extension Navigation Tab Button
2	DCS Extension Navigation Configuration settings

Application Code Manager

Safety Device Library objects can be set-up and configured using Studio 5000® Application Code Manager.

Refer to the section Using Application Code Manager for complete details.

Definition Object: raC_Dvc_FSBM

This object contains the AOI definition and used as linked library to implement object. This gives flexibility to choose to instantiate only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Implementation Object: raC_LD_Dvc_FSBM

Parameter Name	Default Value	Instance Name	Definition	Description
RoutineName	{ObjectName}	{RoutineName}	Routine	Enter Routine name. Routine will be created and Object implement rung(s) inserted. A JSR will be inserted in MainRoutine. If routine name already exists, then object will be inserted into existing routine. By default, parameter is set to Object Name.
TagName	{ObjectName}	{TagName}	Backing Tag	Enter the backing tag of the main AOI. This will serve as the base tag name for other tags in this object that are derived from the base.

Parameter Name	Default Value	Instance Name	Definition	Description
TagDescription	{ObjectDescription}	{TagDescription}		Tag Description of the main AOI backing tag
DisplayName	[ObjectName]	[DisplayName]		Faceplate title/group event list description. Please avoid underscore ("_") at the end.
SafetyFunction	LIGHT CURTAIN			This parameter provides a text name for how this instruction is being used. It is for information/documentation purposes only. Options: - USER DEFINED - EMERGENCY STOP - SAFETY GATE - LIGHT CURTAIN - AREA SCANNER - SAFETY MAT - CABLE (ROPE) PULL SWITCH
InputType	Equivalent - Active High			This parameter selects input channel behavior. Equivalent - Active High or Complimentary.
DiscrepencyTime	100			The amount of time that the inputs can be in an inconsistent state before an instruction fault is generated. Range 5-3000ms.
RestartType_DCS	Automatic			This input configures Output 1 for either Manual or Automatic Restart.
ColdStartType_DCS	Automatic			This parameter specifies the Output 1 behavior when applying controller power or mode change to Run. Automatic or Manual.
ChannelA	ChA		Input Parameter	Enter the input address (data) tag for Channel A from safety digital input module.
ChannelA_InputStatus	ChA_Sts		Input Parameter	Enter the input address (Status) tag for Channel A from safety digital input module.
ChannelB	ChB		Input Parameter	Enter the input address (data) tag for Channel B from safety digital input module.
ChannelB_InputStatus	ChB_Sts		Input Parameter	Enter the input address (Status) tag for Channel B from safety digital input module.
RestartType	Automatic			This input configures Output 1 for either Manual of Automatic Restart
S1S2Time	0			The maximum amount of time allowed between clearing or blocking of the muting sensor inputs Sensor 1 and Sensor 2 before generating a fault. Setting this input to 0 disables the S1-S2 timer. The valid range is 5-180,000 ms.
S2LCTime	0			The maximum amount of time allowed between clearing or blocking of the Sensor 2 muting sensor and the Light Curtain before generating a fault. Setting this input to 0 disables the S2-LC timer. The valid range is 5-180,000 ms.
LCS3Time	0			The maximum amount of time allowed between clearing or blocking of the Light Curtain and the Sensor 2 muting sensor before generating a fault. Setting this input to 0 disables the LC-S3 timer. The valid range is 5-180,000 ms.
S3S4Time	0			The maximum amount of time allowed between clearing or blocking of the muting sensor inputs Sensor 3 and Sensor 4 before generating a fault. Setting this input to 0 disables the S3-S4 timer. The valid range is 5-180,000 ms.
MaxMuteTime	0			The maximum amount of time during which the instruction lets the protective function of the light curtain be disabled before generating a fault. Setting this input to 0 disables the Maximum Mute timer. The valid range is 0-3,600,000 ms.

Parameter Name	Default Value	Instance Name	Definition	Description
MaxOverrideTime	0			The maximum amount of time that the instruction lets the override feature energize the Output 1 output. Setting this input to 0 disables the Maximum Override timer. The valid range is 0-30,000 ms.
Direction	Forward			This input specifies the sequencing direction. Forward: The muting sequence begins with the block of Sensor 1. Reverse: The muting sequence begins with the blocking of Sensor 4.
Sensor1	Sens1		Input Parameter	Enter the input address (data) tag for Sensor 1 from safety digital input module. That is one of two muting sensors, Sensor 1 must be the first sensor to be blocked and the last to be cleared in the muting sequence.
Sensor1_InputStatus	Sense1_Sts		Input Parameter	Enter the input address (Status) tag for Sensor 1 from safety digital input module.
Sensor2	Sens2		Input Parameter	Enter the input address (data) tag for Sensor 2 from safety digital input module. That is one of two muting sensors, Sensor 2 must be the second sensor to be blocked and the first to be cleared in the muting sequence.
Sensor2_InputStatus	Sense2_Sts		Input Parameter	Enter the input address (Status) tag for Sensor 2 from safety digital input module.
MutingLampStatus	MuteInput_Sts		Input Parameter	Enter the input address tag showing status of the muting lamp.
MutingLamp_Output	ML_Output		Input Parameter	Enter the test output address (Data) tag to drive the Muting Lamp from safety digital input module.
ClearArea_Output	CA_Output		Input Parameter	Enter the test output address (Data) tag to drive the Clear Area Lamp from safety digital input module.
ResetTag	Cmd_Reset		Controller Tag	Enter Tagname for Reset signal. Please avoid underscore("_") at the end.

Note that if the tag names are manually entered or not linked to input channel tags a red 'X' will be shown beside the parameter. This is acceptable and the program can still be generated.

Linked Libraries

Link Name	Catalog Number	Revision	Solution	Category
raC_Dvc_FSBM	raC_Dvc_FSBM	1	(RA-LIB) Device	Safety
raC_Opr_SafetyGroupMember	raC_Opr_SafetyGroupMember	1	(RA-LIB) Device	Safety
raC_Dvc_DCS	raC_Dvc_DCS	1	(RA-LIB) Device	Safety

Configured HMI Content

HMI Content	Instance Name	Description
Launch Button	{ObjectName}_GO_LaunchFP	Global Object configured callout instance

Attachments

Name	Description	File Name	Extraction Path
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols SE	(raC-1-SE) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewSE\Global Objects
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols ME	(raC-1-ME) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewME\Global Objects
V1_raC_Dvc_FSBM	Faceplate SE	(raC-1_xx-SE) raC_Dvc_FSBM-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
V1_raC_Dvc_FSBM	Faceplate ME	(raC-1_xx-ME) raC_Dvc_FSBM-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
V1_raC_Dvc_FSBM	View Designer	(raC-1_xx-VD) raC_Dvc_Safety.vpd	{ProjectName}\Visualization\ViewDesigner
V1_RM_raC_DvcSafety	Reference Manual	DEVICE-RM500C-EN-P.pdf	{ProjectName}\Documentation
V1_Safety_Images	HMI Image Set	Safety_Images.zip	{ProjectName}\Visualization\Images

THRSe - Two Hand Run Station Enhanced (raC_Dvc_THRSe)

Overview

The Two Hand Run Station Enhanced device object (raC_Dvc_THRSe) includes a faceplate which displays status and configuration information of a run station. The run station has two buttons which each have both normally closed (N.C.) and normally open (N.O.) contacts. To energize Output 1, the instruction must be enabled and connected with no faults present. Both the left and right buttons must then be pressed within 500 ms of one another.

The instruction is ready when both the left and right buttons are pressed within 500 ms of one another and both the left and right buttons are released within 500 ms of one another.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following videos for this section:
"Operational_Overview_of_Safety_Device_Object_Faceplates"
"Operational_Overview_of_Safety_Application_Example_in_LogixDesigner"

Functional Description

The Two Hand Run Station Enhanced pre-configured Device Objects:

- Collect, Process and Deliver Data between Smart Devices and Application Logic
- Provide Device Status & Diagnostics Faceplates for Machine Startup, Commissioning, Operations, and Maintenance
- Include Common HMI Faceplate Components, Device States, and Control Interfaces providing Application Development and Operation Efficiency

All these features provide quick feedback, shorten recovery time, and simplify implementation.

Required Files

Device Objects include Add-On Instructions (AOIs) and HMI faceplates. The revision number (e.g. 1.01) used in filenames can change as new revisions are created.

Controller Files

Add-On Instructions are reusable code objects that contain encapsulated logic that can streamline implementing your system. This lets you create your own instruction set for programming logic as a supplement to the instruction set provided natively in the ControlLogix® firmware. An Add-On Instruction is

defined once in each controller project, and can be instantiated multiple times in your application code as needed.

The Add-On Instruction must be imported into the controller project to be used in the controller configuration. These can be imported as Add-On Instruction files, or as part of the ROUTINE Import or Import Library Objects wizard.

All Add-On Instruction and ROUTINE Import files can be found in the */Studio 5000 Logix Designer® Files - L5X/* folder in the library.

Device/Item	Add-On Instruction	ROUTINE Import
THRSe	raC_Dvc_THRSe_1.02_AOI.L5X	raC_Dvc_THRSe_1.02_ROUTINE.L5X

FactoryTalk® View HMI Files

FactoryTalk® View ME or SE applications require importing the desired device faceplates in addition to all Global Object (ggfx) files and all images located in the */HMI FactoryTalk View Images - png/* folder of the library. FactoryTalk® View ME files are stored in the */HMI - FactoryTalk View ME/* library folder and FactoryTalk® View SE files are stored in the */HMI - FactoryTalk View SE/* library folder.

Device/Item	Type	FactoryTalk® View ME Faceplate	FactoryTalk® View SE Faceplate
THRSe	Display	(raC-1_02-ME) raC_Dvc_THRSe-Faceplate.gfx	(raC-1_02-SE) raC_Dvc_THRSe-Faceplate.gfx
Graphic Symbols	Global Object	(raC-1-ME) Graphic Symbols - Safety Device	(raC-1-SE) Graphic Symbols - Safety Device.ggfx
Toolbox	Global Object	(raC-1-ME) Toolbox - Safety Device.ggfx	(raC-1-SE) Toolbox - Safety Device.ggfx

Studio 5000 View Designer® HMI Files

All Studio 5000 View Designer® Files can be found in the */HMI - ViewDesigner - vpd/* folder of the library.

Device/Item	Studio 5000 View Designer® Faceplate
THRSe	(raC-1_02-VD) raC_Dvc_Safety.vpd

Studio 5000® Application Code Manager Files

Studio 5000® Application Code Manager (ACM) can be optionally used if it is installed. All devices can be easily registered in the ACM repositories by running the *setup.cmd* file located in the root folder of the library.

Individual HSL4 files are provided as an alternative to running the setup.cmd to allow users to manually register specific implementation objects. Each

object has two files - an Asset Control file and a Device file. The Asset Control files include attachments of all required files for that object. The Device files are used to actually add that device into a Studio 5000 project and these reference the Asset Control files.

All Studio 5000® Application Code Manager files can be found in the /*ApplicationCodeManagerLibraries*/ folder of the library. The files included are as follows:

Implementation Object	Asset Control File (.HSL4)	Device File (.HSL4)
THRSe	(RA-LIB)_Device_Asset-Control_Safety_raC_Dvc_THRSe_(1.2)	(RA-LIB)_Device_Device_Safety_raC_LD_THRSe_(1.2)

Operations

The Safety Discrete objects provides only physical operation mode. There is no virtual device mode offered.

Execution

The following table explains the handling of instruction execution conditions.

Condition	Description
EnableIn False (false rung)	Processing for EnableIn False (false rung) is handled the same as if the device were taken out of service by Command. The device outputs are de-energized and the device is shown as Program Out of Service on the HMI. All alarms are cleared.
Powerup (prescan, first scan)	On prescan, any commands that are received before first scan are discarded. The device is de-energized. On first scan, the device is treated as if it were returning from Hand command source: the instruction state is set based on the position feedback that is received from the device. If the feedback is valid for one position, the device is set to that position. If the device does not have position feedback or the position feedback is invalid, the device is set to the 'unknown/powerup' state. The command source is set to its default, either Operator or Program (unlocked).
Postscan	No SFC Postscan logic is provided.

Add-On Instruction I/O Data

InOut Data

InOut	Function / Description	DataType
Inp_THRSe	Input Interface - THRSe Instruction	DCI_STOP
Out_Ctrl_Inf	Information Interface	raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	Status Interface	raC_UDT_Itf_SafetyDevice_Sts

Input Data

Input	Function/Description	DataType
EnableIn	Enable Input - System Defined Parameter	BOOL
Inp_Suspended	Suspend device operation	BOOL

Output Data

Output	Function/Descritpion	DataType
Sts_DeviceSuspended	Device is suspended	BOOL
Sts_SafetyDemand	Safety demand is present	BOOL
Sts_ResetRequired	Reset is required	BOOL
Sts_DiagnosticPresent	Diagnostic is present	BOOL
Sts_FaultPresent	Fault is present	BOOL
Sts_Ready	Device is ready	BOOL
Sts_ERR	Instruction Error Code	DINT
Sts_eReady	0: Ready 1: Safety Demand 2: Reset Required 3: Suspended	SINT
raC_Dvc_THRSe	Unique Parameter Name for auto - discovery	BOOL
EnableOut	Enable Output - System Defined Parameter	BOOL
Sts_StationBypassed	Station is bypassed	BOOL

Data Types

The following Safety Common Control Interface tags are the primary device program tags to read and write to when interfacing to safety devices. The value of using these tags in your specific application code is that you may use a number of different safety devices such as THRSe, DCS, etc without having to update your application device interface tags.

Refer to the Interfaces section for detailed information on interfaces.

raC_UDT_Itf_SafetyDevice_Inf

Member	Description	Data Type
Type	Provides device object type information e.g. raC_Dvc_SafetyGroupMonitor,raC_Dvc_THRSe,raC_Dvc_THRSeTL,etc.	STR0032
DisplayName	Provides object display name	STR0024
eClass	Provide instruction type information eClass ID: Instruction Type 0: DCS 1: DCSTL 2: DCSTM 3: CROUT 5: SafetyGroupMonitor 6: TSAM 7: TSSM 8: FSBM 9: THRSe 10: SMAT 11:SS2 12: SS1 13:SFX 14:STO	DINT

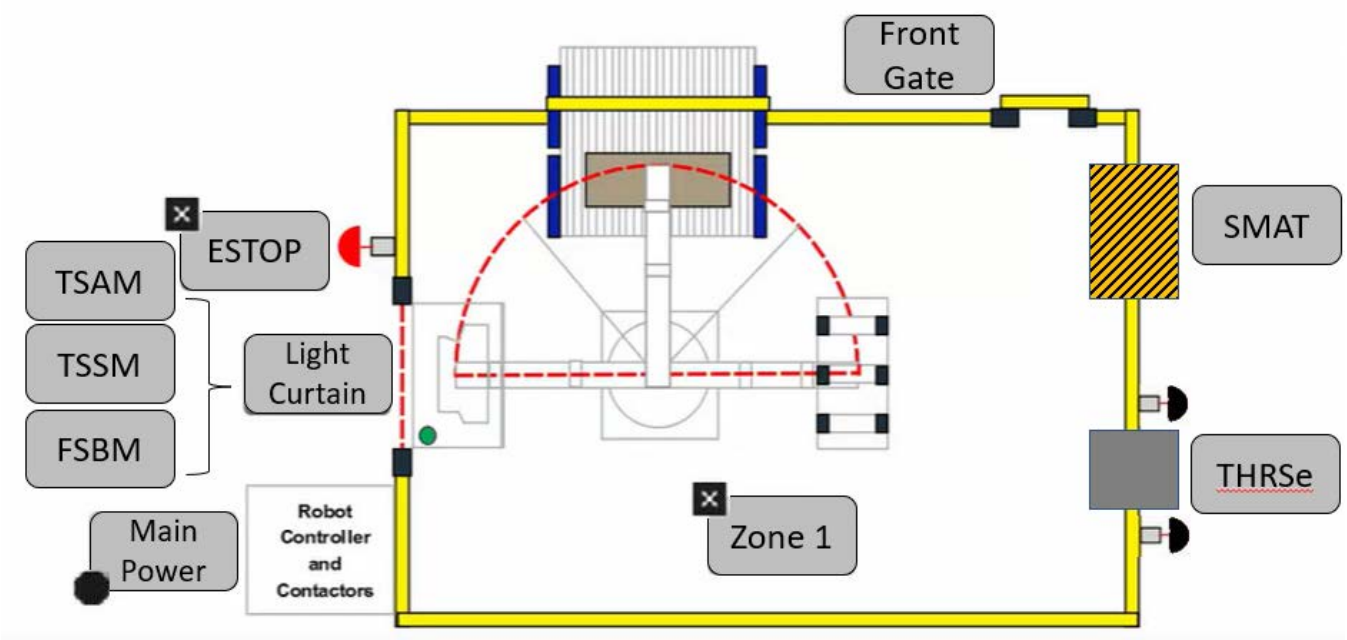
raC_UDT_Itf_SafetyDevice_Sts

This is the Safety Device Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data. The table below shows member names, descriptions, and tag data types.

Input	Description	Data Type
bSts	Bit overlay (Visible) covering all subsequent Boolean members. (Unused)	DINT
DeviceSuspended	Device suspended status (Inp_Suspend = 1)	BOOL
SafetyDemand	Safety demand status ON (1): Safety demand is present on the device object OFF (0): No safety demand	BOOL
ResetRequired	Reset Required status ON (1): Device requires reset signal OFF (0): No reset required	BOOL
DiagnosticPresent	Diagnostic status ON (1): A diagnostic is present in the device object OFF (0): Device is operating normally	BOOL
FaultPresent	Fault Status ON (1): A fault is present in the device object OFF (0): Device is operating normally	BOOL
Muted	Device Muted status (Available for THRSeTM) ON (1): Muted status is present on one or more safety instructions within a specific safety group OFF (0): None of the instructions within group is muted	BOOL

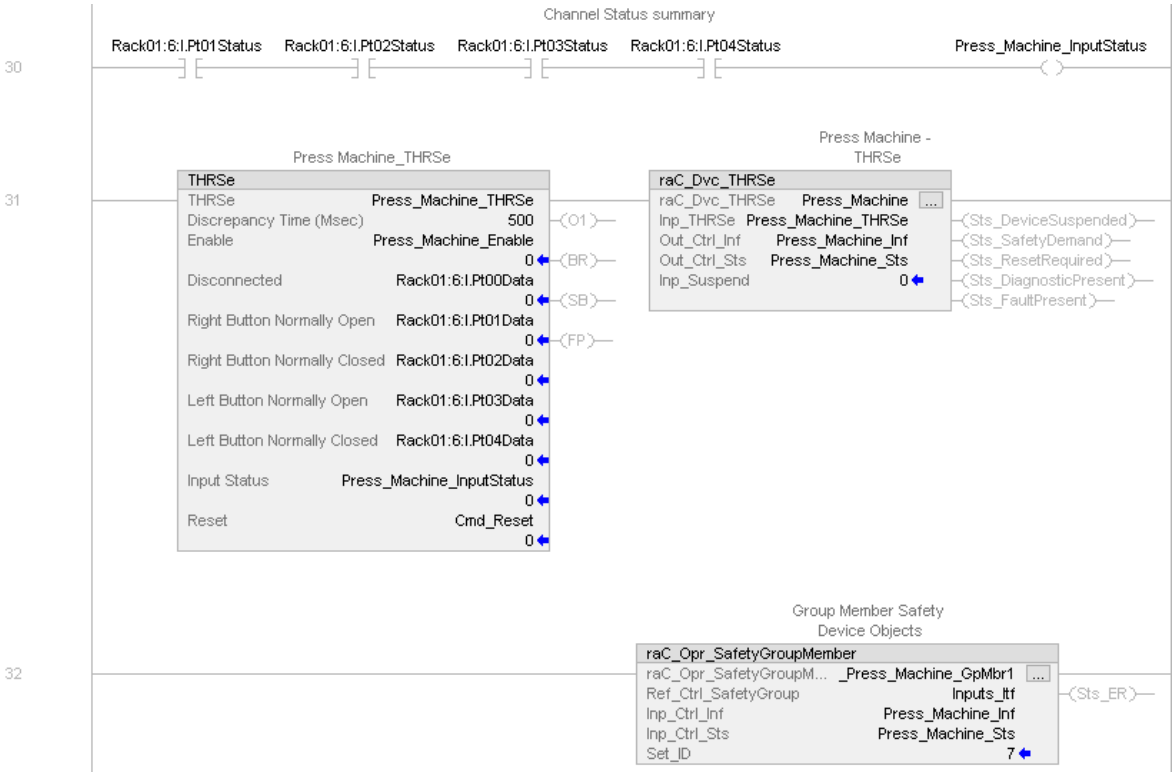
Programming Example

Fully configured device routine is provided below for reference. This example is taken from the *Safety Program - Inputs* routine in the *SafetytApplication.ACD* example application provided in the library.



This example uses the THRSe instruction for an two handed run station function shown as *THRSe* in the safe machine diagram. In the example application this system is referred to as Press_Machine. The right and left button normally open and normally closed contacts are tied to safety class discrete inputs. Rung descriptions are as follows:

- Rung 30: Check input statuses
- Rung 31: Configure THRSe and raC_Dvc_THRSe instructions
- Rung 32: raC_Opr_SafetyGroupMember (optional) assigns Press_Machine THRSe to Inputs Safety Group at ID #7.



Note that this programming example is the same code that is imported when either importing the supplied ROUTINE .L5X files or when using Application Code Manager or the Studio 5000 Import Library Objects wizard plug-in.

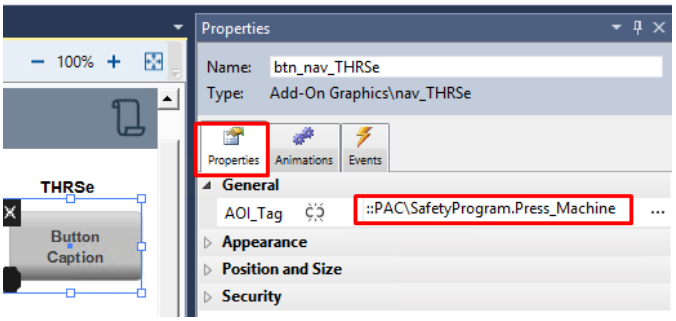
Graphic Symbols

Graphic Symbols are used as launch buttons within HMI applications to open up faceplate displays. See Basic Launch Button Attributes section for details on configuration and indicators

FactoryTalk® View ME/SE Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Global Object Parameter Values
btn_nav_safety		Faceplate navigation button with string tag label. This launch button graphic object allows the user to navigate to the device object faceplate. The text on the button face is set to the tag's .Inf_DisplayName.	#101: AOI Instance (e.g. {::[PAC]Program::SafetyProgram._InstanceName})

Studio 5000 View Designer® Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Property Configuration
Launch		The supplied launch button in View Designer is used to navigate to the faceplate in a user application.	

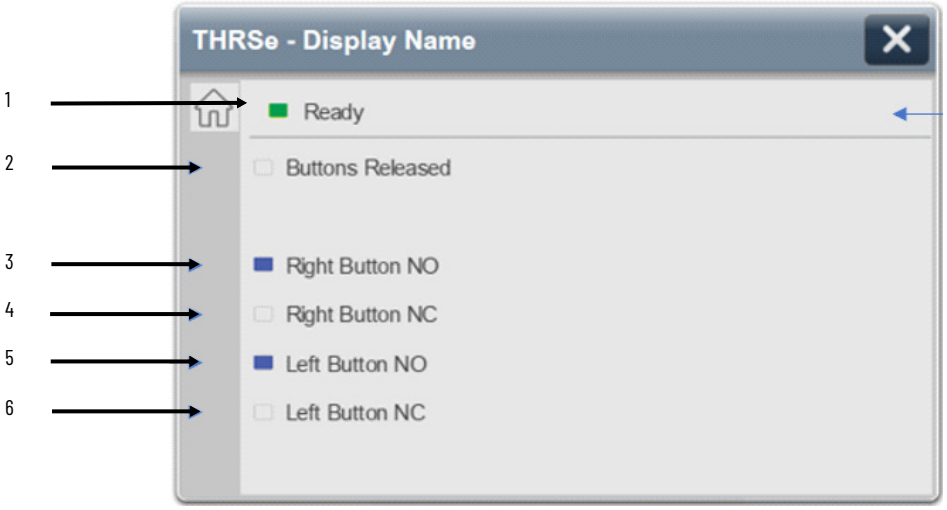
Faceplates

There are basic faceplate attributes that are common across all instructions. See Basic Faceplate Attributes on page 27.

Home

The Home tab is the main tab of the faceplate. It provides the status of the safety input device along with configuration information.

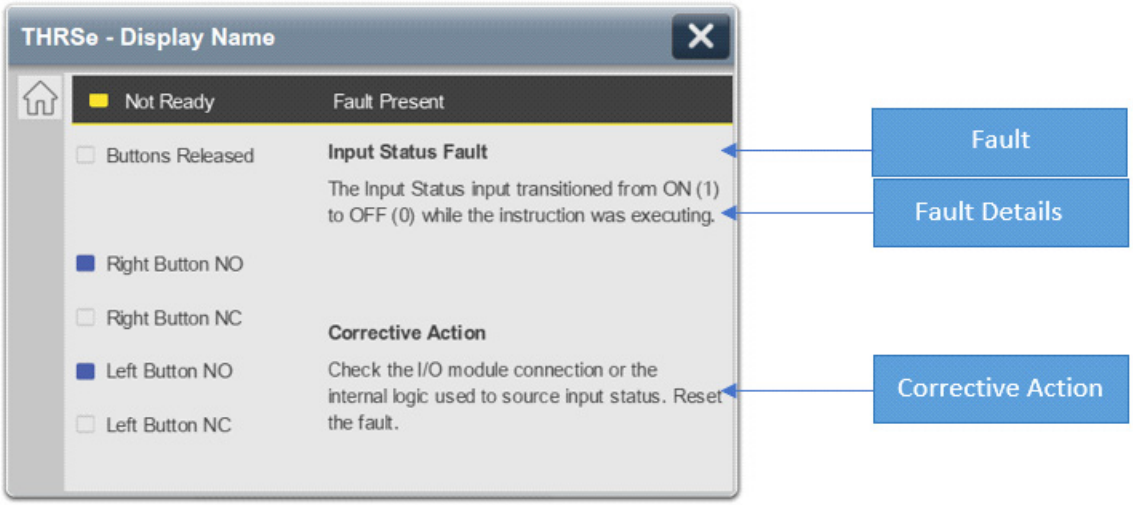
The Home tab shows the input type and input channel A/B statuses. Additionally there is a Ready banner at the top showing the status of the device.



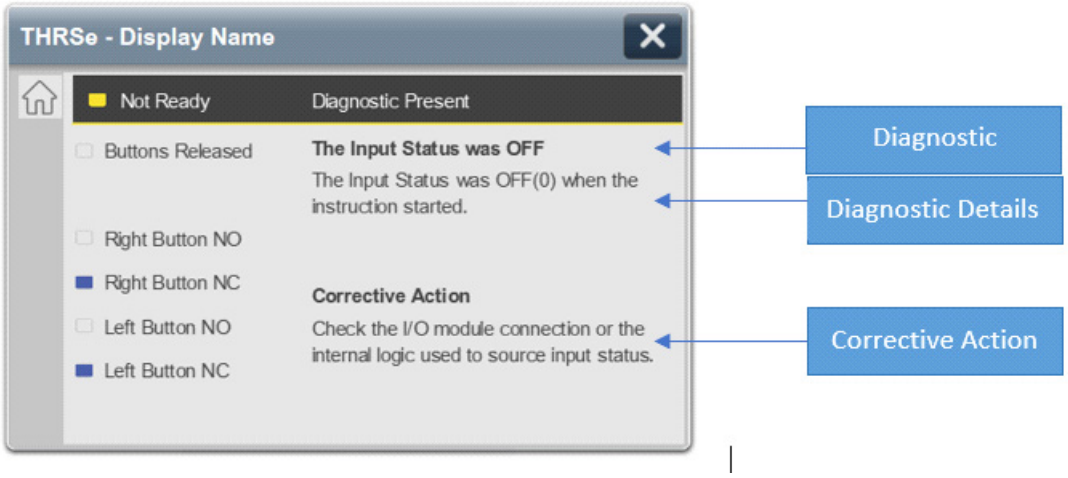
Item	Description
1	Banner- Ready Status
2	Buttons Released: The Buttons Released (BR) output turns ON (1) whenever the Two-hand Run Station is connected and enabled, no faults are present, and both the right and left buttons are in the released (safe) state. In this case, all four contacts are in the safe state. OFF (0) = Grey LED, The left and right buttons are not released. ON (1) = Blue LED, the left and right buttons are released.
3	Right Button N.O. Status: Normally Open contact for the right button. OFF (0) = Gray LED. Right button is not pressed. ON (1) = Blue LED. Right button is pressed.
4	Right Button N.C. Status: Normally Closed contact for the right button. OFF (0) = Gray LED. Right button is not released ON (1) = Blue LED. Right button is released
5	Left Button N.O. Status: Normally Open contact for the leftbutton. OFF (0) = Gray LED. Left button is not pressed. ON (1) = Blue LED. Left button is pressed.
6	Left Button N.C. Status: Normally Closed contact for the left button. OFF (0) = Gray LED. Left button is not released ON (1) = Blue LED. Left button is released

Home tab screen also provides fault and diagnostic information with required corrective actions on the right plane of the faceplate

When there is fault present, banner with black background is displayed on top of home screen with **Not Ready** and **Fault Present** text.



When there is diagnostic present, banner with black background is displayed on top of home screen with **Not Ready** and **Diagnostic Present** text.



Application Code Manager

Safety Device Library objects can be set-up and configured using Studio 5000® Application Code Manager.

Refer to the section Using Application Code Manager for complete details.

Definition Object: raC_Dvc_THRSe

This object contains the AOI definition and used as linked library to implement object. This gives flexibility to choose to instantiate only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Implementation Object: raC_LD_Dvc_THRSe

Parameter Name	Default Value	Instance Name	Definition	Description
RoutineName	{ObjectName}	{RoutineName}	Routine	Enter Routine name. Routine will be created and Object implement rung(s) inserted. A JSR will be inserted in MainRoutine. If routine name already exists, then object will be inserted into existing routine. By default, parameter is set to Object Name.
TagName	{ObjectName}	{TagName}	Backing Tag	Enter the backing tag of the main AOI. This will serve as the base tag name for other tags in this object that are derived from the base.
TagDescription	{ObjectDescription}	{TagDescription}		Tag Description of the main AOI backing tag

Parameter Name	Default Value	Instance Name	Definition	Description
DisplayName	[ObjectName]	[DisplayName]		Faceplate title/group event list description. Please avoid underscore ("_") at the end.
DiscrepencyTime	100			The amount of time that the inputs can be in an inconsistent state before an instruction fault is generated. Range 100-3000ms.
Enable	Enable_Station		Controller Tag	Enter Tagname that allows the device to be enabled.
Disconnected	Disconnected		Input Parameter	Enter the input address (Data) tag for Disconnected from a safety digital input module.
RightButtonNormallyOpen	RBNO		Input Parameter	Enter the input address (data) tag for the Right Button Normally Open contact from safety digital input module.
RightButtonNormallyOpen_InputStatus	RBNO_Status		Input Parameter	Enter the input address (Status) tag for the Right Button Normally Open contact from safety digital input module.
RightButtonNormallyClosed	RBNO		Input Parameter	Enter the input address (data) tag for the Right Button Normally Closed contact from safety digital input module.
RightButtonNormallyClosed_InputStatus	RBNO_Status		Input Parameter	Enter the input address (Status) tag for the Right Button Normally Closed contact from safety digital input module.
LeftButtonNormallyOpen	RBNO		Input Parameter	Enter the input address (data) tag for the Left Button Normally Open contact from safety digital input module.
LeftButtonNormallyOpen_InputStatus	RBNO_Status		Input Parameter	Enter the input address (Status) tag for the Left Button Normally Open contact from safety digital input module.
LeftButtonNormallyClosed	RBNO		Input Parameter	Enter the input address (data) tag for the Left Button Normally Closed contact from safety digital input module.
LeftButtonNormallyClosed_InputStatus	RBNO_Status		Input Parameter	Enter the input address (Status) tag for the Left Button Normally Closed contact from safety digital input module.
ResetTag	Cmd_Reset		Controller Tag	Enter Tagname for Reset signal. Please avoid underscore ("_") at the end.

Note that if the tag names are manually entered or not linked to input channel tags a red 'X' will be shown beside the parameter. This is acceptable and the program can still be generated.

Linked Libraries

Link Name	Catalog Number	Revision	Solution	Category
raC_Dvc_THRSe	raC_Dvc_THRSe	1	(RA-LIB) Device	Safety
raC_Opr_SafetyGroupMember	raC_Opr_SafetyGroupMember	1	(RA-LIB) Device	Safety

Configured HMI Content

HMI Content	Instance Name	Description
Launch Button	{ObjectName}_GO_LaunchFP	Global Object configured callout instance

Attachments

Name	Description	File Name	Extraction Path
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols SE	(raC-1-SE) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewSE\Global Objects
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols ME	(raC-1-ME) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewME\Global Objects
V1_raC_Dvc_THRSe	Faceplate SE	(raC-1_xx-SE) raC_Dvc_THRSe-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
V1_raC_Dvc_THRSe	Faceplate ME	(raC-1_xx-ME) raC_Dvc_THRSe-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
V1_raC_Dvc_THRSe	View Designer	(raC-1_xx-VD) raC_Dvc_Safety.vpd	{ProjectName}\Visualization\ViewDesigner
V1_RM_raC_DvcSafety	Reference Manual	DEVICE-RM500C-EN-P.pdf	{ProjectName}\Documentation
V1_Safety_Images	HMI Image Set	Safety_Images.zip	{ProjectName}\Visualization\Images

SMAT - Safety Mat (raC_Dvc_SMAT)

Overview

The Safety Mat device object (raC_Dvc_SMAT) includes a faceplate which displays occupied status and fault information of a safety mat device.

The instruction is ready when the safety mat is unoccupied, all safety demand conditions are cleared, no fault is present, and the correct reset actions are carried out.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following videos for this section:
"Operational_Overview_of_Safety_Device_Object_Faceplates"
"Operational_Overview_of_Safety_Application_Example_in_LogixDesigner"

Functional Description

The Safety Mat pre-configured Device Objects:

- Collect, Process and Deliver Data between Smart Devices and Application Logic
- Provide Device Status & Diagnostics Faceplates for Machine Startup, Commissioning, Operations, and Maintenance
- Include Common HMI Faceplate Components, Device States, and Control Interfaces providing Application Development and Operation Efficiency

All these features provide quick feedback, shorten recovery time, and simplify implementation.

Required Files

Device Objects include Add-On Instructions (AOIs) and HMI faceplates. The revision number (e.g. 1.01) used in filenames can change as new revisions are created.

Controller Files

Add-On Instructions are reusable code objects that contain encapsulated logic that can streamline implementing your system. This lets you create your own instruction set for programming logic as a supplement to the instruction set provided natively in the ControlLogix® firmware. An Add-On Instruction is defined once in each controller project, and can be instantiated multiple times in your application code as needed.

The Add-On Instruction must be imported into the controller project to be used in the controller configuration. These can be imported as Add-On

Instruction files, or as part of the ROUTINE Import or Import Library Objects wizard.

All Add-On Instruction and ROUTINE Import files can be found in the */Studio 5000 Logix Designer® Files - L5X/* folder in the library.

Device/Item	Add-On Instruction	ROUTINE Import
SMAT	raC_Dvc_SMAT_1.02_AOI.L5X	raC_Dvc_SMAT_1.02_ROUTINE.L5X

FactoryTalk® View HMI Files

FactoryTalk® View ME or SE applications require importing the desired device faceplates in addition to all Global Object (ggfx) files and all images located in the */HMI FactoryTalk View Images - png/* folder of the library. FactoryTalk® View ME files are stored in the */HMI - FactoryTalk View ME/* library folder and FactoryTalk® View SE files are stored in the */HMI - FactoryTalk View SE/* library folder.

Device/Item	Type	FactoryTalk® View ME Faceplate	FactoryTalk® View SE Faceplate
SMAT	Display	(raC-1_02-ME) raC_Dvc_SMAT-Faceplate.gfx	(raC-1_02-SE) raC_Dvc_SMAT-Faceplate.gfx
Graphic Symbols	Global Object	(raC-1-ME) Graphic Symbols - Safety Device	(raC-1-SE) Graphic Symbols - Safety Device.ggfx
Toolbox	Global Object	(raC-1-ME) Toolbox - Safety Device.ggfx	(raC-1-SE) Toolbox - Safety Device.ggfx

Studio 5000 View Designer® HMI Files

All Studio 5000 View Designer® Files can be found in the */HMI - ViewDesigner - vpd/* folder of the library.

Device/Item	Studio 5000 View Designer® Faceplate
SMAT	(raC-1_02-VD) raC_Dvc_Safety.vpd

Studio 5000® Application Code Manager Files

Studio 5000® Application Code Manager (ACM) can be optionally used if it is installed. All devices can be easily registered in the ACM repositories by running the *setup.cmd* file located in the root folder of the library.

Individual HSL4 files are provided as an alternative to running the setup.cmd to allow users to manually register specific implementation objects. Each object has two files - an Asset Control file and a Device file. The Asset Control files include attachments of all required files for that object. The Device files are used to actually add that device into a Studio 5000 project and these reference the Asset Control files.

All Studio 5000® Application Code Manager files can be found in the /*ApplicationCodeManagerLibraries*/ folder of the library. The files included are as follows:

Implementation Object	Asset Control File (.HSL4)	Device File (.HSL4)
SMAT	(RA-LIB)_Device_Asset-Control_Safety_raC_Dvc_SMAT_(1.2)	(RA-LIB)_Device_Device_Safety_raC_LD_SMAT_(1.2)

Operations

The Safety Discrete objects provides only physical operation mode. There is no virtual device mode offered.

Execution

The following table explains the handling of instruction execution conditions.

Condition	Description
EnableIn False (false rung)	Processing for EnableIn False (false rung) is handled the same as if the device were taken out of service by Command. The device outputs are de-energized and the device is shown as Program Out of Service on the HMI. All alarms are cleared.
Powerup (prescan, first scan)	On prescan, any commands that are received before first scan are discarded. The device is de-energized. On first scan, the device is treated as if it were returning from Hand command source: the instruction state is set based on the position feedback that is received from the device. If the feedback is valid for one position, the device is set to that position. If the device does not have position feedback or the position feedback is invalid, the device is set to the 'unknown/powerup' state. The command source is set to its default, either Operator or Program (unlocked).
Postscan	No SFC Postscan logic is provided.

Add-On Instruction I/O Data

InOut Data

InOut	Function / Description	DataType
Inp_SMAT	Input Interface - SMAT Instruction	DCI_STOP
Out_Ctrl_Inf	Information Interface	raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	Status Interface	raC_UDT_Itf_SafetyDevice_Sts

Input Data

Input	Function/Description	DataType
EnableIn	Enable Input - System Defined Parameter	BOOL
Inp_Suspended	Suspend device operation	BOOL

Output Data

Output	Function/Descritpion	DataType
Sts_DeviceSuspended	Device is suspended	BOOL
Sts_SafetyDemand	Safety demand is present	BOOL
Sts_ResetRequired	Reset is required	BOOL
Sts_DiagnosticPresent	Diagnostic is present	BOOL
Sts_FaultPresent	Fault is present	BOOL
Sts_Ready	Device is ready	BOOL
Sts_ERR	Instruction Error Code	DINT
Sts_eReady	0: Ready 1: Safety Demand 2: Reset Required 3: Suspended	SINT
raC_Dvc_SMAT	Unique Parameter Name for auto - discovery	BOOL
EnableOut	Enable Output - System Defined Parameter	BOOL

Data Types

The following Safety Common Control Interface tags are the primary device program tags to read and write to when interfacing to safety devices. The value of using these tags in your specific application code is that you may use a number of different safety devices such as SMAT, DCS, etc without having to update your application device interface tags.

Refer to the Interfaces section for detailed information on interfaces.

raC_UDT_Itf_SafetyDevice_Inf

Member	Description	Data Type
Type	Provides device object type information e.g. raC_Dvc_SafetyGroupMonitor, raC_Dvc_SMAT, raC_Dvc_SMATTL, etc.	STR0032
DisplayName	Provides object display name	STR0024
eClass	Provide instruction type information eClass ID: Instruction Type 0: DCS 1: DCSTL 2: DCSTM 3: CROUT 5: SafetyGroupMonitor 6: TSAM 7: TSSM 8: FSBM 9: THRSe 10: SMAT 11:SS2 12: SS1 13:SFX 14:STO	DINT

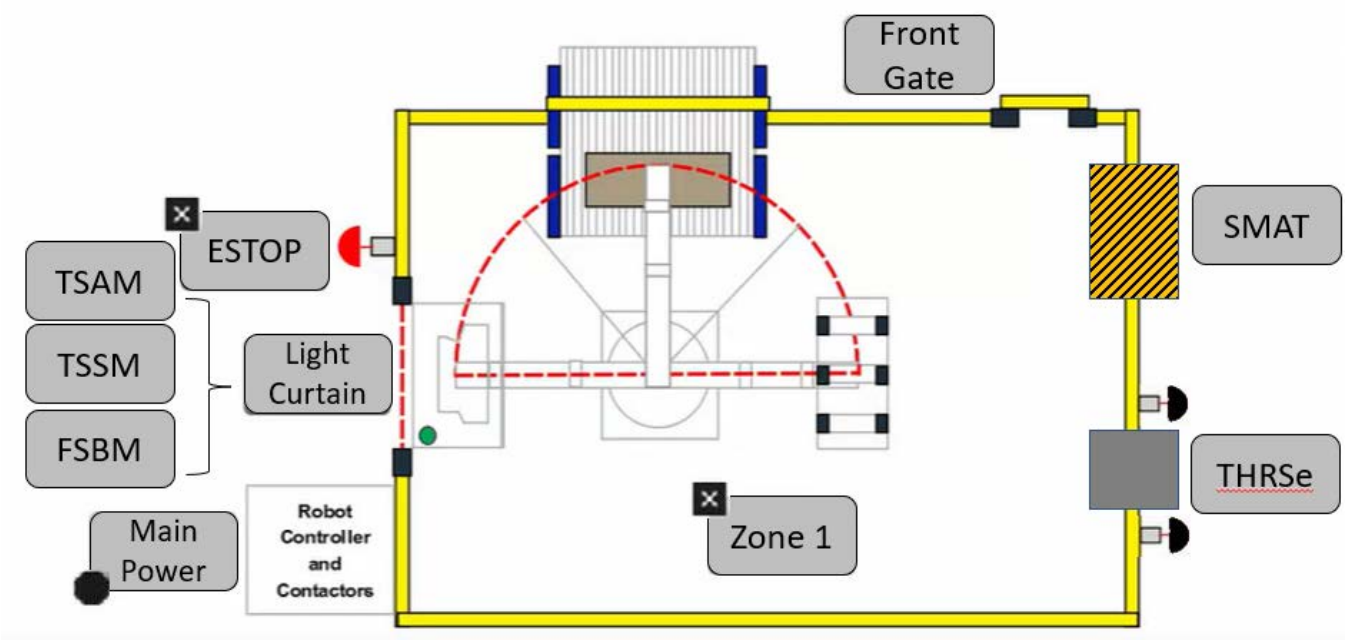
raC_UDT_Itf_SafetyDevice_Sts

This is the Safety Device Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data. The table below shows member names, descriptions, and tag data types.

Input	Description	Data Type
bSts	Bit overlay (Visible) covering all subsequent Boolean members. (Unused)	DINT
DeviceSuspended	Device suspended status (Inp_Suspend = 1)	BOOL
SafetyDemand	Safety demand status ON (1): Safety demand is present on the device object OFF (0): No safety demand	BOOL
ResetRequired	Reset Required status ON (1): Device requires reset signal OFF (0): No reset required	BOOL
DiagnosticPresent	Diagnostic status ON (1): A diagnostic is present in the device object OFF (0): Device is operating normally	BOOL
FaultPresent	Fault Status ON (1): A fault is present in the device object OFF (0): Device is operating normally	BOOL
Muted	Device Muted status (Available for SMATTM) ON (1): Muted status is present on one or more safety instructions within a specific safety group OFF (0): None of the instructions within group is muted	BOOL

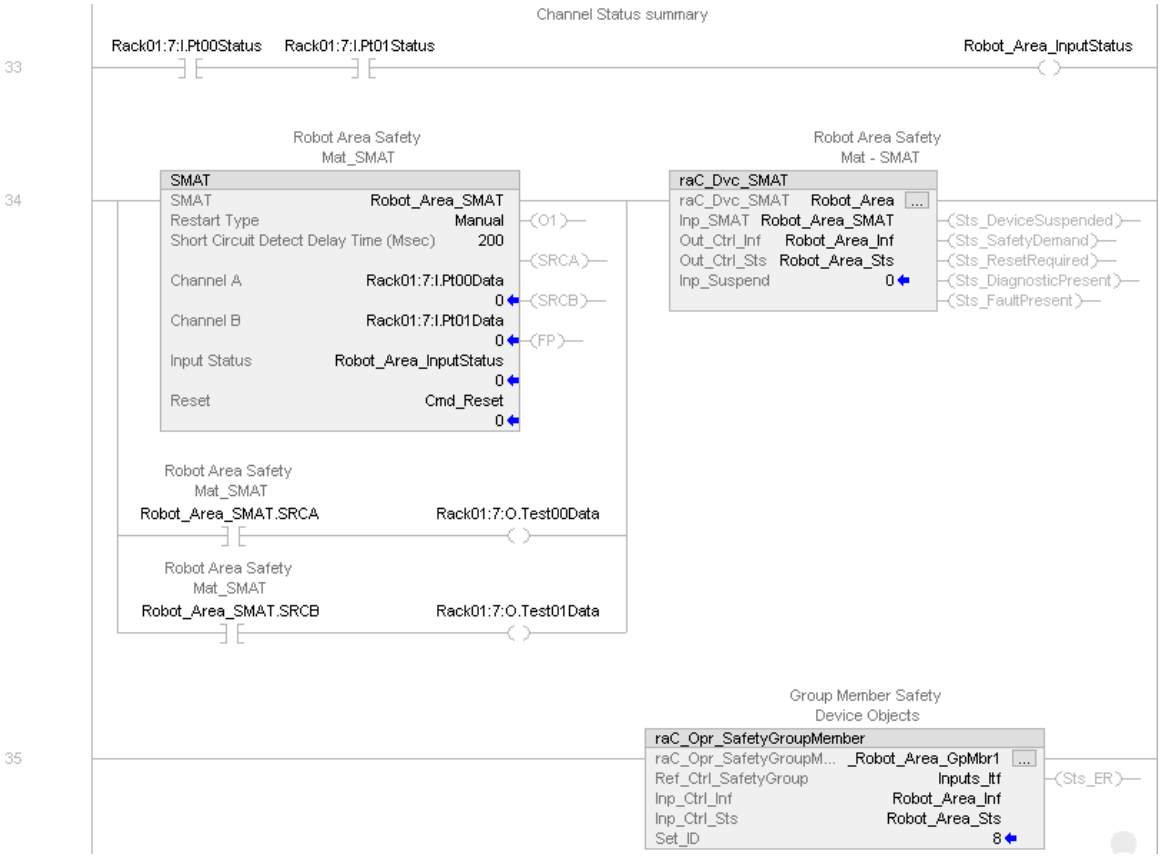
Programming Example

Fully configured device routine is provided below for reference. This example is taken from the *Safety Program - Inputs* routine in the *SafetytApplication.ACD* example application provided in the library.



This example uses the SMAT instruction for safety mat function shown as *SMAT* in the safe machine diagram. In the example application this system is referred to as Robot_Area. Channel A and Channel B are tied to safety class discrete inputs. Rung descriptions are as follows:

- Rung 33: Check input statuses
- Rung 34: Configure SMAT and raC_Dvc_SMAT instructions
- Rung 35: raC_Opr_SafetyGroupMember (optional) assigns Robot_Area SMAT to Inputs Safety Group at ID #8.



Note that this programming example is the same code that is imported when either importing the supplied ROUTINE.L5X files or when using Application Code Manager or the Studio 5000 Import Library Objects wizard plug-in.

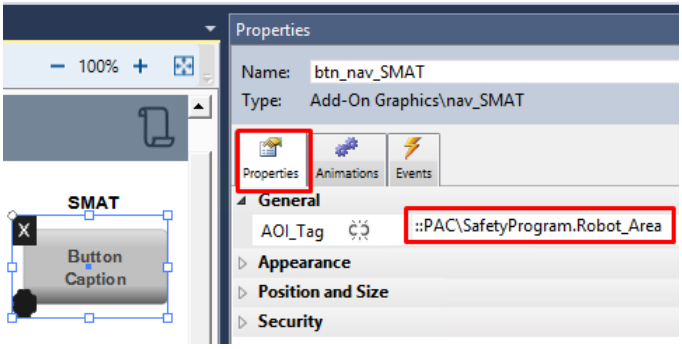
Graphic Symbols

Graphic Symbols are used as launch buttons within HMI applications to open up faceplate displays. See Basic Launch Button Attributes section for details on configuration and indicators

FactoryTalk® View ME/SE Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Global Object Parameter Values
btn_nav_safety		Faceplate navigation button with string tag label. This launch button graphic object allows the user to navigate to the device object faceplate. The text on the button face is set to the tag's .Inf_DisplayName.	#101: AOI Instance (e.g. {::[PAC]Program::SafetyProgram._InstanceName})

Studio 5000 View Designer® Graphic Symbols

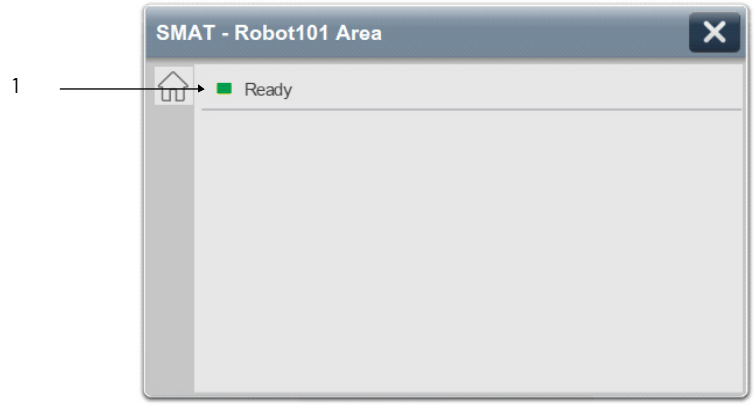
Graphic Symbol Name	Graphic Symbol	Description	Property Configuration
Launch		The supplied launch button in View Designer is used to navigate to the faceplate in a user application.	

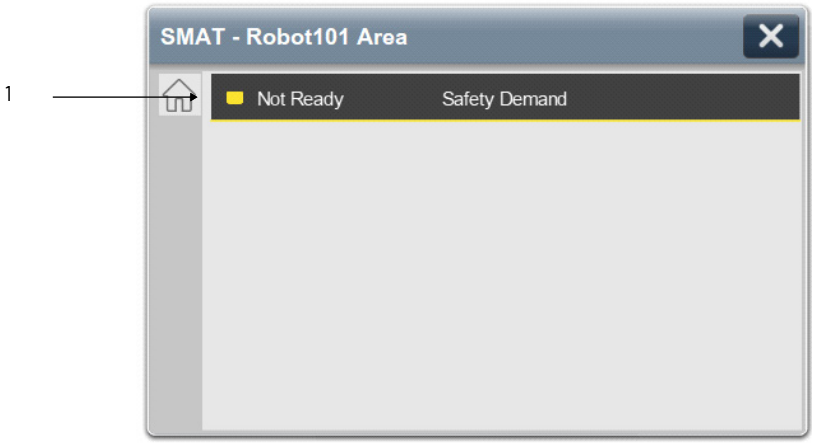
Faceplates

There are basic faceplate attributes that are common across all instructions. See Basic Faceplate Attributes on page 27.

Home

The Home tab is the main tab of the faceplate. It provides the status of the safety mat input device in the Ready banner at the top. The device status is either Ready or Not Ready with a Safety Demand.

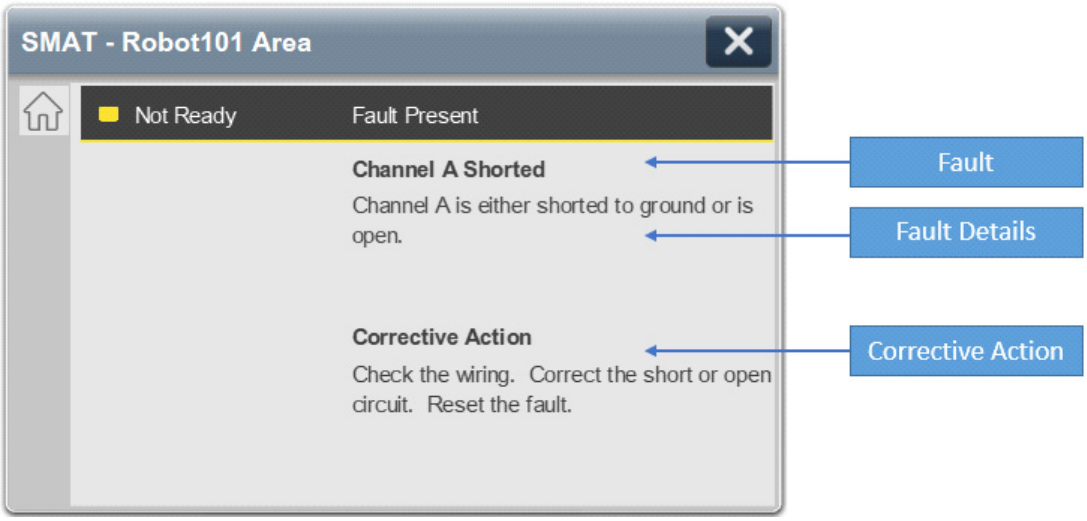




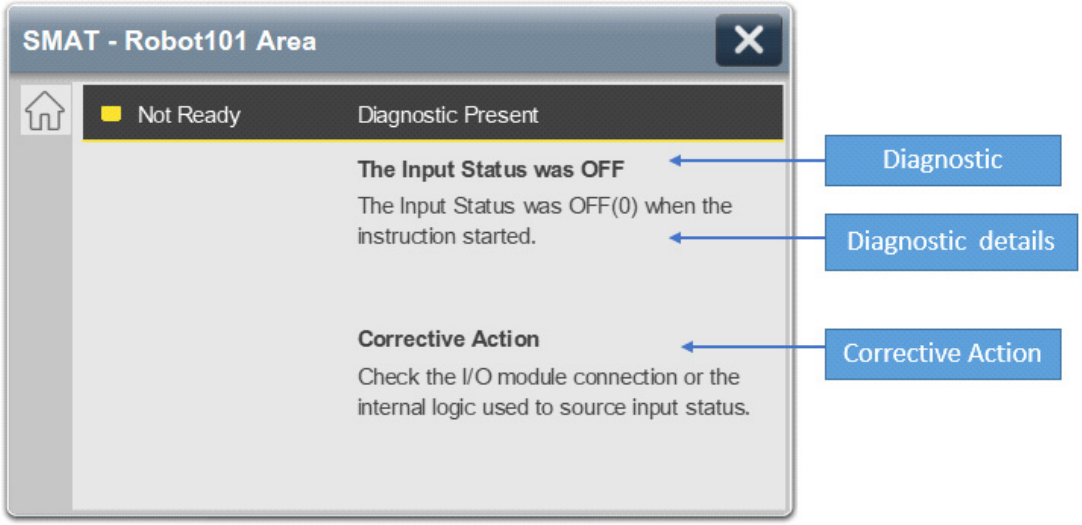
Item	Description
1	Banner- Ready Status

Home tab screen also provides fault and diagnostic information with required corrective actions on the right plane of the faceplate

When there is fault present, banner with black background is displayed on top of home screen with **Not Ready** and **Fault Present** text.



When there is diagnostic present, banner with black background is displayed on top of home screen with **Not Ready** and **Diagnostic Present** text.



Application Code Manager

Safety Device Library objects can be set-up and configured using Studio 5000® Application Code Manager.

Refer to the section Using Application Code Manager for complete details.

Definition Object: raC_Dvc_SMAT

This object contains the AOI definition and used as linked library to implement object. This gives flexibility to choose to instantiate only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Implementation Object: raC_LD_Dvc_SMAT

Parameter Name	Default Value	Instance Name	Definition	Description
RoutineName	{ObjectName}	{RoutineName}	Routine	Enter Routine name. Routine will be created and Object implement rung(s) inserted. A JSR will be inserted in MainRoutine. If routine name already exists, then object will be inserted into existing routine. By default, parameter is set to Object Name.
TagName	{ObjectName}	{TagName}	Backing Tag	Enter the backing tag of the main AOI. This will serve as the base tag name for other tags in this object that are derived from the base.
TagDescription	{ObjectDescription}	{TagDescription}		Tag Description of the main AOI backing tag
DisplayName	[ObjectName]	[DisplayName]		Faceplate title/group event list description. Please avoid underscore ("_") at the end.
RestartType	Manual			This input configures Output 1 for either Manual or Automatic Restart.
ShortCircuitDetectDelayTime	5			This parameter is the time that the instruction uses to determine the difference between a short circuit and the safety mat being occupied. The valid range is 5-250 ms.
ChannelA	ChA		Input Parameter	Enter the input address (data) tag for Channel A from safety digital input module.
ChannelA_InputStatus	ChA_Sts		Input Parameter	Enter the input address (Status) tag for Channel A from safety digital input module.
ChannelA_TestOutput	ChA_TestOut		Input Parameter	Enter the test output address (Data) tag for Channel A from safety digital input module.
ChannelB	ChB		Input Parameter	Enter the input address (data) tag for Channel B from safety digital input module.
ChannelB_InputStatus	ChB_Sts		Input Parameter	Enter the input address (Status) tag for Channel B from safety digital input module.
ChannelB_TestOutput	ChB_TestOut		Input Parameter	Enter the test output address (Data) tag for Channel B from safety digital input module.

Parameter Name	Default Value	Instance Name	Definition	Description
ResetTag	Cmd_Reset		Controller Tag	Enter Tagname for Reset signal. Please avoid underscore("_") at the end.

Note that if the tag names are manually entered or not linked to input channel tags a red 'X' will be shown beside the parameter. This is acceptable and the program can still be generated.

Linked Libraries

Link Name	Catalog Number	Revision	Solution	Category
raC_Dvc_SMAT	raC_Dvc_SMAT	1	(RA-LIB) Device	Safety
raC_Opr_SafetyGroupMember	raC_Opr_SafetyGroupMember	1	(RA-LIB) Device	Safety

Configured HMI Content

HMI Content	Instance Name	Description
Launch Button	{ObjectName}_GO_LaunchFP	Global Object configured callout instance

Attachments

Name	Description	File Name	Extraction Path
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols SE	(raC-1-SE) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewSE\Global Objects
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols ME	(raC-1-ME) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewME\Global Objects
V1_raC_Dvc_SMAT	Faceplate SE	(raC-1_xx-SE) raC_Dvc_SMAT-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
V1_raC_Dvc_SMAT	Faceplate ME	(raC-1_xx-ME) raC_Dvc_SMAT-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
V1_raC_Dvc_SMAT	View Designer	(raC-1_xx-VD) raC_Dvc_Safety.vpd	{ProjectName}\Visualization\ViewDesigner
V1_RM_raC_DvcSafety	Reference Manual	DEVICE-RM500C-EN-P.pdf	{ProjectName}\Documentation
V1_Safety_Images	HMI Image Set	Safety_Images.zip	{ProjectName}\Visualization\Images

SFX - Safety Feedback Interface (raC_Dvc_SFX)

Overview

The Safety Feedback Interface device object (raC_Dvc_SFX) includes a faceplate which displays SFX Output status, Axis Homed Status, Information about number of Actual Cycles etc. It also includes trends and fault information of a Safety Feedback Interface device.

The instruction is considered ready when the following conditions are met:

- There are no faults present.
- The correct reset actions have been executed.
- SFX Output is in the ON State.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following videos for this section:
"Operational_Overview_of_Safety_Device_Object_Faceplates"
"Operational_Overview_of_Safety_Application_Example_in_LogixDesigner"

Functional Description

The Safety Feedback Interface pre-configured Device Objects:

- Analyze and Process Axis Data using SFX instruction and Application Logic
- Provide Device Status & Diagnostics Faceplates for Machine Safety and Operations.
- Include Common HMI Faceplate Components, Device States, and Control Interfaces providing Application Development and Operation Efficiency

All these features provide quick feedback, shorten recovery time, and simplify implementation.

Required Files

Device Objects include Add-On Instructions (AOIs) and HMI faceplates. The revision number (e.g. 1.01) used in filenames can change as new revisions are created.

Controller Files

Add-On Instructions are reusable code objects that contain encapsulated logic that can streamline implementing your system. This lets you create your own instruction set for programming logic as a supplement to the instruction set provided natively in the ControlLogix® firmware. An Add-On Instruction is defined once in each controller project, and can be instantiated multiple times in your application code as needed.

The Add-On Instruction must be imported into the controller project to be used in the controller configuration. These can be imported as Add-On Instruction files, or as part of the ROUTINE Import or Import Library Objects wizard.

All Add-On Instruction and ROUTINE Import files can be found in the */Studio 5000 Logix Designer® Files - L5X/* folder in the library.

Device/Item	Add-On Instruction	ROUTINE Import
SFX	raC_Dvc_SFX_1.04_AOI.L5X	raC_Dvc_SFX_1.04_ROUTINE.L5X

FactoryTalk® View HMI Files

FactoryTalk® View ME or SE applications require importing the desired device faceplates in addition to all Global Object (ggfx) files and all images located in the */HMI FactoryTalk View Images - png/* folder of the library. FactoryTalk® View ME files are stored in the */HMI - FactoryTalk View ME/* library folder and FactoryTalk® View SE files are stored in the */HMI - FactoryTalk View SE/* library folder.

Device/Item	Type	FactoryTalk® View ME Faceplate	FactoryTalk® View SE Faceplate
SFX	Display	(raC-1_04-ME) raC_Dvc_SFX-Faceplate.gfx	(raC-1_04-SE) raC_Dvc_SFX-Faceplate.gfx
SFX - Trend	Display	(raC-1_04-ME) raC_Dvc_SFX-Trend.gfx	(raC-1_04-SE) raC_Dvc_SFX-Trend.gfx
Graphic Symbols	Global Object	(raC-1-ME) Graphic Symbols - Safety Device	(raC-1-SE) Graphic Symbols - Safety Device.ggfx
Toolbox	Global Object	(raC-1-ME) Toolbox - Safety Device.ggfx	(raC-1-SE) Toolbox - Safety Device.ggfx

Studio 5000 View Designer® HMI Files

All Studio 5000 View Designer® Files can be found in the */HMI - ViewDesigner - vpd/* folder of the library.

Device/Item	Studio 5000 View Designer® Faceplate
SFX	(raC-1_04-VD) raC_Dvc_Safety.vpd

Studio 5000® Application Code Manager Files

Studio 5000® Application Code Manager (ACM) can be optionally used if it is installed. All devices can be easily registered in the ACM repositories by running the *setup.cmd* file located in the root folder of the library.

Individual HSL4 files are provided as an alternative to running the setup.cmd to allow users to manually register specific implementation objects. Each object has two files - an Asset Control file and a Device file. The Asset Control files include attachments of all required files for that object. The Device files

are used to actually add that device into a Studio 5000 project and these reference the Asset Control files.

All Studio 5000® Application Code Manager files can be found in the /*ApplicationCodeManagerLibraries*/ folder of the library. The files included are as follows:

Implementation Object	Asset Control File (.HSL4)	Device File (.HSL4)
SFX	(RA-LIB)_Device_Asset-Control_Safety_raC_Dvc_SFX_(1.4)	(RA-LIB)_Device_Device_Safety_raC_LD_SFX_(1.4)

Operations

The Safety motion monitoring objects provides only physical operation mode. There is no virtual device mode offered.

Execution

The following table explains the handling of instruction execution conditions.

Condition	Description
EnableIn False (false rung)	Processing for EnableIn False (false rung) is handled the same as if the Execution of the rung which contains SFX Instruction were removed by Command or Logical Condition, the Instruction and AOI outputs are de-energized and the object is shown as SFX is OFF Scan on the HMI. All the Status bits "_bsts.xx" are also cleared to zero.
Powerup (prescan, first scan)	On prescan, any commands that are received before first scan are discarded. The device object is de-energized. On first scan, the Object Common Services, Internal control Enable in true and Internal control Enable in False bits are de-energized. All the Status bits "_bsts.xx" and ONS bits are cleared to zero.
Postscan	No SFC Postscan logic is provided.

Add-On Instruction I/O Data

InOut Data

InOut	Function / Description	DataType
Inp_SFX	Input Interface - SFX Instruction	SAFETY_FEEDBACK_INTERFACE
Out_Ctrl_Inf	Information Interface	raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	Status Interface	raC_UDT_Itf_SafetyDevice_Sts

Input Data

Input	Function/Description	DataType
EnableIn	Enable Input - System Defined Parameter	BOOL
Cfg_TrendReset_SpdPercent	Set the percentage of run speed to reset the trend Y-axis. Set to a level that speed variations while stopped do not reset the Y-axis of trend. Default 1%.	REAL
Cfg_TrendReset_PosPercent	Set the percentage of run position to reset the trend Y-axis. Set to a level that position variations while position reaches to zero do not reset the Y-axis of trend. Default 1%.	REAL
Inp_ResetTrend	1 = Reset All Trend's Y-Axis scale to zero.	BOOL

Output Data

Output	Function/Descritpion	DataType
raC_Dvc_SFX	Unique Parameter Name for auto - discovery	BOOL
Sts_DiagnosticPresent	Diagnostic is present	BOOL
Sts_FaultPresent	Fault is present	BOOL
Sts_Safe_Homed	Homing Completed	BOOL
Sts_FeedbackValid	Feedback Position and Feedback Velocity are valid for use. OFF(0): Not Valid ON(1): Valid	BOOL
Sts_ConnectionFaulted	Connection status OFF(0): OK ON(1): Fault	BOOL
Sts_eReady	0: Ready 1: Safety Demand 2: Reset Required 3: Suspended (Status 1, 2, 3 are not applicable for SFX)	SINT
Sts_eNotReady	0: Ready 1: Fault Present 2: Diagnostic Present	SINT
Sts_Output	Output is ON (Ready State)	BOOL
Sts_Executing	Executing state of Instruction	BOOL
Val_ActualCycles	Actual Cycles Value	REAL
Val_ActualPosition	Actual Position Value	REAL
Val_ActualSpeed	Actual Speed Value	REAL

Output	Function/Descritpion	DataType
Val_DiagnosticCode	Diagnostic codes for different diagnostic condition	DINT
Val_FaultType	Fault codes	SINT
EnableOut	Enable Output - System Defined Parameter	BOOL

Data Types

The following Safety Common Control Interface tags are the primary device program tags to read and write to when interfacing to safety devices. The value of using these tags in your specific application code is that you may use a number of different safety Instructions such as SS2, SFX, etc without having to update your application device interface tags.

Refer to the Interfaces section for detailed information on interfaces.

raC_UDT_Itf_SafetyDevice_Inf

Member	Description	Data Type
Type	Provides device object type information e.g. raC_Dvc_SafetyGroupMonitor, raC_Dvc_SFX, etc.	STR0032
DisplayName	Provides object display name	STR0024
eClass	Provide instruction type information eClass ID: Instruction Type 0: DCS 1: DCSTL 2: DCSTM 3: CROUT 5: SafetyGroupMonitor 6: TSAM 7: TSSM 8: FSBM 9: THRSe 10: SMAT 11:SS2 12: SS1 13:SFX 14:STO	DINT

raC_UDT_Itf_SafetyDevice_Sts

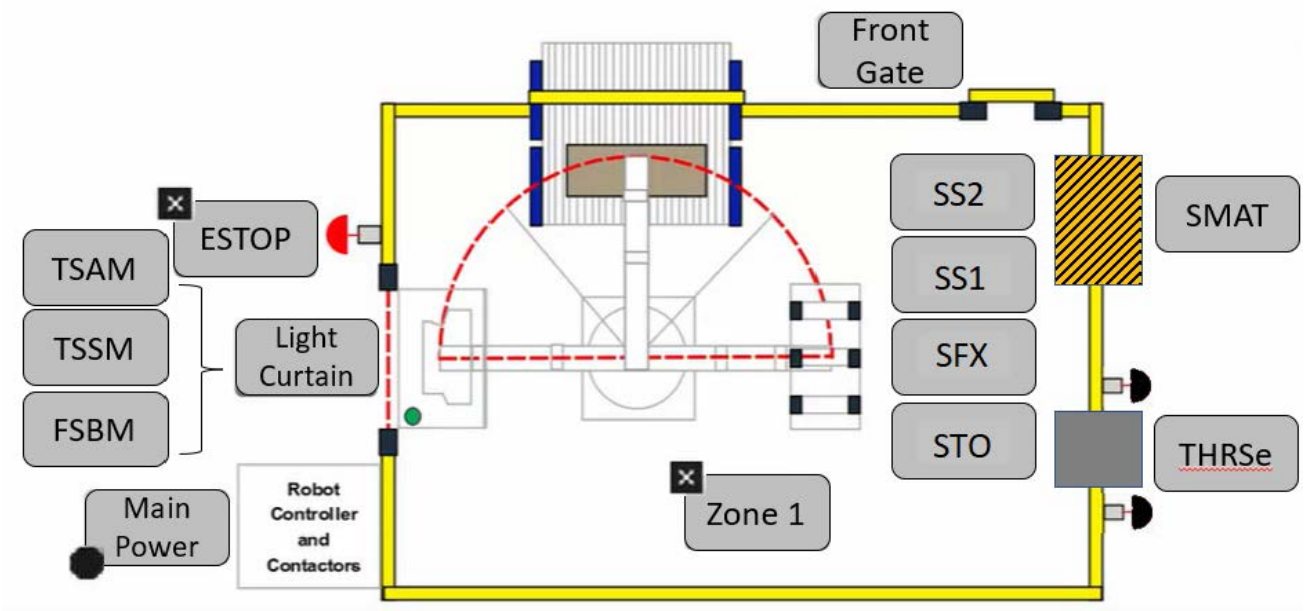
This is the Safety Device Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data. The table below shows member names, descriptions, and tag data types.

Input	Description	Data Type
bSts	Bit overlay (Visible) covering all subsequent Boolean members.	DINT
DeviceSuspended	Device suspended status (Inp_Suspend = 1) Note:- For SFX this value is always zero.	BOOL
SafetyDemand	Safety demand status ON (1): Safety demand is present on the device object OFF (0): No safety demand Note:- For SFX this value is always zero.	BOOL

Input	Description	Data Type
ResetRequired	Reset Required status ON (1): Device requires reset signal OFF (0): No reset required Note:- For SFX this value is always zero.	BOOL
DiagnosticPresent	Diagnostic status ON (1): A diagnostic is present in the device object OFF (0): Device is operating normally	BOOL
FaultPresent	Fault Status ON (1): A fault is present in the device object OFF (0): Device is operating normally	BOOL
Muted	Device Muted status (Available for TSAMTM) ON (1): Muted status is present on one or more safety instructions within a specific safety group OFF (0): None of the instructions within group is muted Note:- For SFX this value is always zero.	BOOL

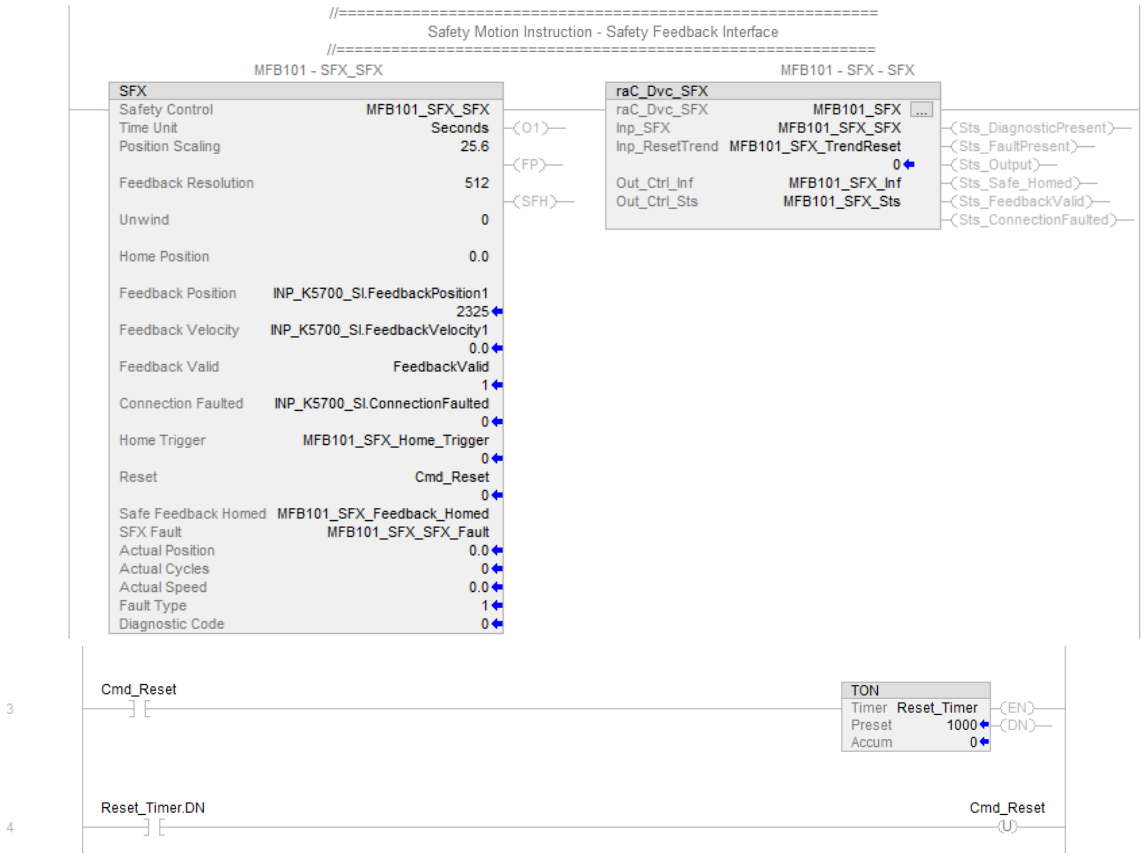
Programming Example

Fully configured device routine is provided below for reference. This example is taken from the ***Safety Program - SafetyMotionInstruction* routine** in the *SafetytApplication.ACD* example application provided in the library.



This example uses the SFX instruction to read and analyze Safety Feedback Interface data of Axis. In the example application, this system is referred to as Robot_Area. The SFX instruction is mainly used to read the feedback data like Actual Position, Actual Speed, Speed Limit, Actual Cycles of the Axis and utilizing the same data to plot the trends on the faceplate. Rung descriptions are as follows:

- Rung 0: Configure SFX and raC_Dvc_SFX instructions
- Rung 3: Common Cmd_Reset Rung for SFX, SS2 and SS1
- Rung 4: Common Unlatch Cmd_Reset for SFX, SS2 and SS1

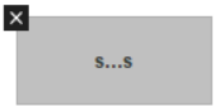


Note that this programming example is the same code that is imported when either importing the supplied ROUTINE.L5X files or when using Application Code Manager or the Studio 5000 Import Library Objects wizard plug-in.

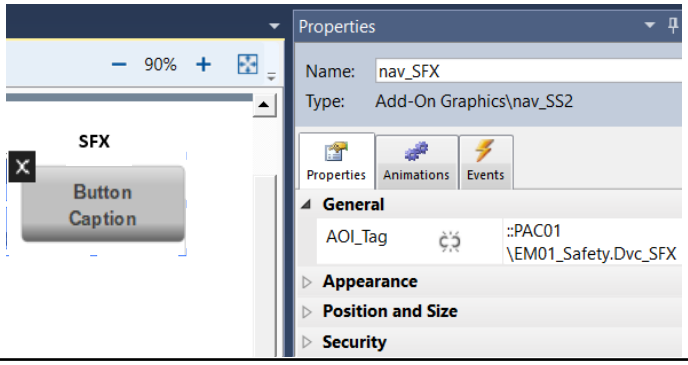
Graphic Symbols

Graphic Symbols are used as launch buttons within HMI applications to open up faceplate displays. See Basic Launch Button Attributes section for details on configuration and indicators

FactoryTalk® View ME/SE Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Global Object Parameter Values
btn_nav_SFX		Faceplate navigation button with string tag label. This launch button graphic object allows the user to navigate to the device object faceplate. The text on the button face is set to the tag's .Inf_DisplayName.	#101: AOI Instance (e.g. {::[PAC]Program::SafetyProgram._InstanceName})

Studio 5000 View Designer® Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Property Configuration
Launch		The supplied launch button in View Designer is used to navigate to the faceplate in a user application.	

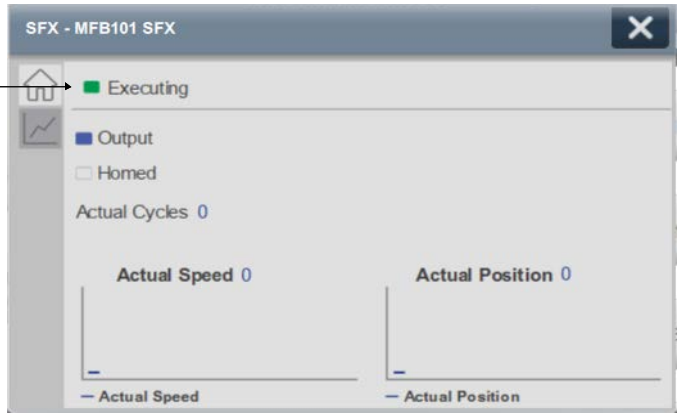
Faceplates

There are basic faceplate attributes that are common across all instructions. See Basic Faceplate Attributes on page 27.

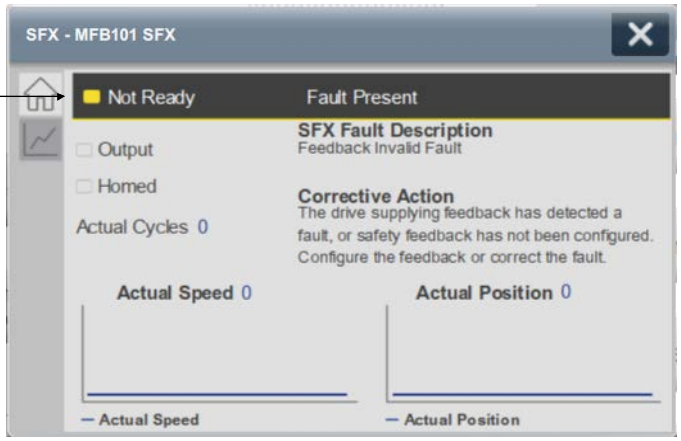
Home

The Home tab is the main tab of the faceplate. It provides the status of the Safety Feedback Interface object in the Ready banner at the top of the faceplate. The Ready Banner has four states of Condition **SFX is OFF Scan**, **Executing** (Ready), **Not Ready** (Faulted)., **SFX Device Object is OFF Scan**.

1

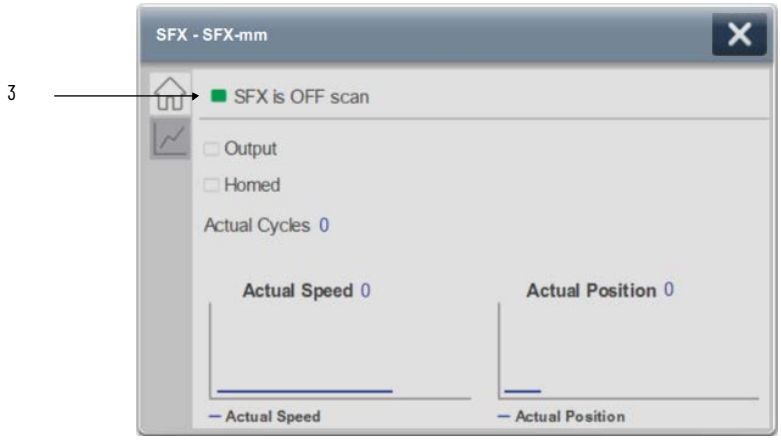


2

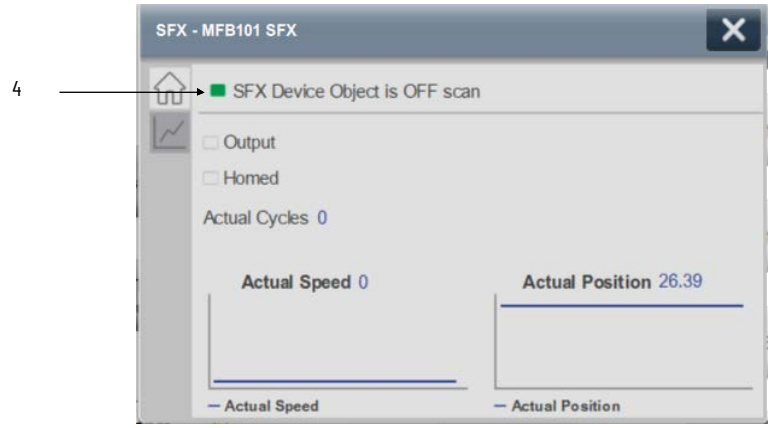


Item	Description
1	Banner- Executing State (Ready)
2	Banner- Not Ready Status

SFX is OFF Scan refers to the situation when the execution of the rung containing the SFX instruction is taken off scan or when the rung execution is disabled.



SFX Device Object is OFF Scan refers to the situation when the execution of the rung containing the SFX AOI is taken off scan or when the rung execution is disabled

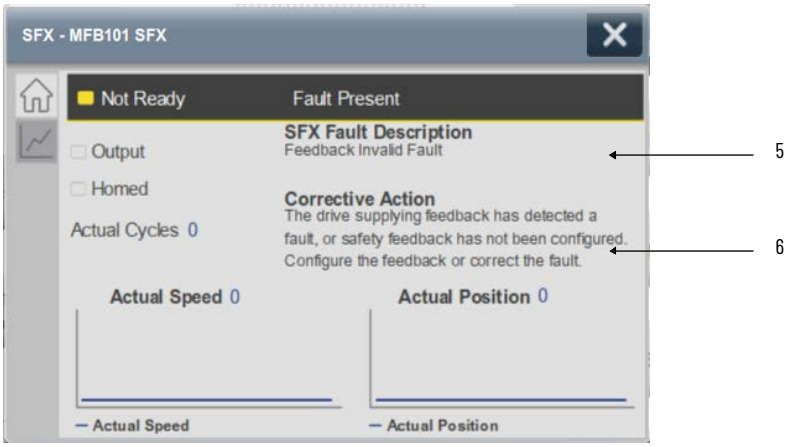


Item	Description
3	Banner- SFX is OFF Scan
4	Banner- SFX Device Object is OFF Scan

Home tab screen also provides fault and diagnostic information with required

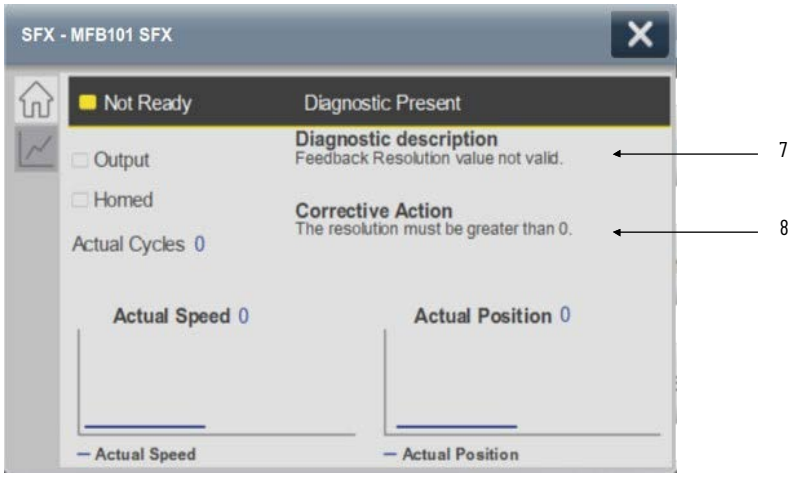
corrective actions on the right plane of the faceplate

When there is fault present, banner with black background is displayed on top of home screen with **Not Ready** and **Fault Present** text.



Item	Description
5	Fault Present - Fault Description
6	Fault Present - Fault Corrective Action

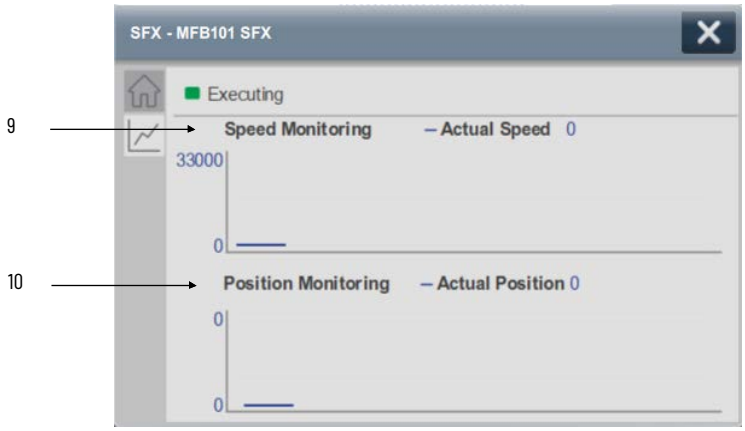
When there is diagnostic present, banner with black background is displayed on top of home screen with **Not Ready** and **Diagnostic Present** text.



Item	Description
7	Diagnostic Present - Diagnostic Short Description
8	Diagnostic Present - Diagnostic Corrective Action

Trend

The Trend tab is the Second tab of the faceplate. It is basically a data monitoring tab or the tab with graphical representation of operational data within SFX Instruction. Totally two graphs or trends are plotted on the trend tab **Speed Monitoring & Position Monitoring** .



Item	Description
9	Speed Monitoring Trend (Plotting Trend for Actual Speed)
10	Position Monitoring Trend (Plotting Trend for Actual Position)

Application Code Manager

Safety Device Library objects can be set-up and configured using Studio 5000® Application Code Manager.

Refer to the section Using Application Code Manager for complete details.

Definition Object: raC_Dvc_SFX

This object contains the AOI definition and used as linked library to implement object. This gives flexibility to choose to instantiate only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Implementation Object: raC_LD_Dvc_SFX

Parameter Name	Default Value	Instance Name	Definition	Description
RoutineName	{ObjectName}	{RoutineName}	Routine	Enter Routine name. Routine will be created and Object implement rung(s) inserted. A JSR will be inserted in MainRoutine. If routine name already exists, then object will be inserted into existing routine. By default, parameter is set to Object Name.
TagName	{ObjectName}	{TagName}	Backing Tag	Enter the backing tag of the main AOI. This will serve as the base tag name for other tags in this object that are derived from the base.
TagDescription	{ObjectDescription}	{TagDescription}		Tag Description of the main AOI backing tag
DisplayName	[ObjectName]	[DisplayName]		Faceplate title/group event list description. Please avoid underscore ("_") at the end.
TimeUnit	Seconds	{TimeUnit}		Time Unit Seconds or Minutes
PositionScaling	512.0	{PositionScaling}		User Entered Value for Position Scaling
FeedbackResolution	512	{FeedbackResolution}		User Entered Value for Feedback Resolution
Unwind	0	{Unwind}		User Entered Value for Unwind
HomePosition	0.0	{HomePosition}		User Entered Value for Home Position

Note that if the tag names are manually entered or not linked to input channel tags a red 'X' will be shown beside the parameter. This is acceptable and the program can still be generated.

Linked Libraries

Link Name	Catalog Number	Revision	Solution	Category
raC_Dvc_SFX	raC_Dvc_SFX	1	(RA-LIB) Device	Safety

Configured HMI Content

HMI Content	Instance Name	Description
Launch Button	{ObjectName}_GO_LaunchFP	Global Object configured callout instance

Attachments

Name	Description	File Name	Extraction Path
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols SE	(raC-1-SE) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewSE\Global Objects
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols ME	(raC-1-ME) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewME\Global Objects
V1_raC_Dvc_SFX	Faceplate SE	(raC-1_xx-SE) raC_Dvc_SFX-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
V1_raC_Dvc_SFX	Faceplate ME	(raC-1_xx-ME) raC_Dvc_SFX-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
V1_raC_Dvc_SFX	View Designer	(raC-1_xx-VD) raC_Dvc_Safety.vpd	{ProjectName}\Visualization\ViewDesigner
V1_RM_raC_DvcSafety	Reference Manual	DEVICE-RM500C-EN-P.pdf	{ProjectName}\Documentation
V1_Safety_Images	HMI Image Set	Safety_Images.zip	{ProjectName}\Visualization\Images

SS1 - Safe Stop 1(raC_Dvc_SS1)

Overview

The raC_Dvc_SS1 (Safe Stop 1) device object features a faceplate that provides an overview of the instruction's operational status. It displays information such as Output, Request, Stop Monitor Delay Active, Standstill and STO Request. Additionally, the faceplate includes trends and fault details related to the Safe Stop 1 device.

The instruction is considered ready when the following conditions are met:

- There are no faults present.
- No diagnostics issues are detected.
- The correct reset actions have been executed.
- SS1 request is not active.
- SS1 Output is in the ON state

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following videos for this section:
"Operational_Overview_of_Safety_Device_Object_Faceplates"
"Operational_Overview_of_Safety_Application_Example_in_LogixDesigner"

Functional Description

The Safe Stop 1 pre-configured Device Objects:

- Analyze and Process Axis Data using SS1 instruction and Application Logic
- Provide Device Status & Diagnostics Faceplates for Machine Safety and Operations.
- Include Common HMI Faceplate Components, Device States, and Control Interfaces providing Application Development and Operation Efficiency

All these features provide quick feedback, shorten recovery time, and simplify implementation.

Required Files

Device Objects include Add-On Instructions (AOIs) and HMI faceplates. The revision number (e.g. 1.01) used in filenames can change as new revisions are created.

Controller Files

Add-On Instructions are reusable code objects that contain encapsulated logic that can streamline implementing your system. This lets you create your own instruction set for programming logic as a supplement to the instruction set

provided natively in the ControlLogix® firmware. An Add-On Instruction is defined once in each controller project, and can be instantiated multiple times in your application code as needed.

The Add-On Instruction must be imported into the controller project to be used in the controller configuration. These can be imported as Add-On Instruction files, or as part of the ROUTINE Import or Import Library Objects wizard.

All Add-On Instruction and ROUTINE Import files can be found in the */Studio 5000 Logix Designer® Files - L5X/* folder in the library.

Device/Item	Add-On Instruction	ROUTINE Import
SS1	raC_Dvc_SS1_1.04_AOI.L5X	raC_Dvc_SS1_1.04_ROUTINE.L5X

FactoryTalk® View HMI Files

FactoryTalk® View ME or SE applications require importing the desired device faceplates in addition to all Global Object (ggfx) files and all images located in the */HMI FactoryTalk View Images - png/* folder of the library. FactoryTalk® View ME files are stored in the */HMI - FactoryTalk View ME/* library folder and FactoryTalk® View SE files are stored in the */HMI - FactoryTalk View SE/* library folder.

Device/Item	Type	FactoryTalk® View ME Faceplate	FactoryTalk® View SE Faceplate
SS1	Display	(raC-1_04-ME) raC_Dvc_SS1-Faceplate.gfx	(raC-1_04-SE) raC_Dvc_SS1-Faceplate.gfx
SS1 - Trend	Display	(raC-1_04-ME) raC_Dvc_SS1-Trend.gfx	(raC-1_04-SE) raC_Dvc_SS1-Trend.gfx
Graphic Symbols	Global Object	(raC-1-ME) Graphic Symbols - Safety Device	(raC-1-SE) Graphic Symbols - Safety Device.ggfx
Toolbox	Global Object	(raC-1-ME) Toolbox - Safety Device.ggfx	(raC-1-SE) Toolbox - Safety Device.ggfx

Studio 5000 View Designer® HMI Files

All Studio 5000 View Designer® Files can be found in the */HMI - ViewDesigner - vpd/* folder of the library.

Device/Item	Studio 5000 View Designer® Faceplate
SS1	(raC-1_04-VD) raC_Dvc_Safety.vpd

Studio 5000® Application Code Manager Files

Studio 5000® Application Code Manager (ACM) can be optionally used if it is installed. All devices can be easily registered in the ACM repositories by running the *setup.cmd* file located in the root folder of the library.

Individual HSL4 files are provided as an alternative to running the setup.cmd to allow users to manually register specific implementation objects. Each object has two files - an Asset Control file and a Device file. The Asset Control files include attachments of all required files for that object. The Device files are used to actually add that device into a Studio 5000 project and these reference the Asset Control files.

All Studio 5000® Application Code Manager files can be found in the /*ApplicationCodeManagerLibraries*/ folder of the library. The files included are as follows:

Implementation Object	Asset Control File (.HSL4)	Device File (.HSL4)
SS1	(RA-LIB)_Device_Asset-Control_Safety_raC_Dvc_SS1_(1.4)	(RA-LIB)_Device_Device_Safety_raC_LD_SS1_(1.4)

Operations

The Safety motion monitoring objects provides only physical operation mode. There is no virtual device mode offered.

Execution

The following table explains the handling of instruction execution conditions.

Condition	Description
EnableIn False (false rung)	Processing for EnableIn False (false rung) is handled the same as if the Execution of the rung which contains SS1 Instruction and AOI were removed by Command or Logical Condition, the Instruction and AOI outputs are de-energized and the object is shown as SS1 is OFF Scan on the HMI. All the Status bits "_bsts.xx" are also cleared to zero.
Powerup (prescan, first scan)	On prescan, any commands that are received before first scan are discarded. The device object is de-energized. On first scan, the Object Common Services, Internal control Enable in true and Internal control Enable in False bits are de-energized. All the Status bits "_bsts.xx" and ONS bits are cleared to zero.
Postscan	No SFC Postscan logic is provided.

Add-On Instruction I/O Data

InOut Data

InOut	Function / Description	DataType
Inp_SS1	Input Interface - SS1 Instruction	SAFE_STOP_1
Inp_SFX	Input Interface - SFX Instruction	SAFETY_FEEDBACK_INTERFACE
Out_Ctrl_Inf	Information Interface	raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	Status Interface	raC_UDT_Itf_SafetyDevice_Sts

Input Data

Input	Function/Description	DataType
EnableIn	Enable Input - System Defined Parameter	BOOL
Inp_SS1_Active	Active Status of SS1	BOOL
Cfg_TrendReset_SpdPercent	Set the percentage of run speed to reset the trend Y-axis. Set to a level that speed variations while stopped do not reset the Y-axis of trend. Default 1%.	REAL
Inp_ResetTrend	1 = Reset Decel Trend's Y-Axis scale to zero.	BOOL

Output Data

Output	Function/Descritpion	DataType
raC_Dvc_SS1	Unique Parameter Name for auto - discovery	BOOL
EnableOut	Enable Output - System Defined Parameter	BOOL
Sts_ResetRequired	Reset is required	BOOL
Sts_DiagnosticPresent	Diagnostic is present	BOOL
Sts_FaultPresent	Fault is present	BOOL
Sts_Request	Request is ON	BOOL
Sts_Active	Executing and Request State	BOOL
Sts_Standstill	Standstill output Is ON	BOOL
Sts_STORequest	STO Request output is ON	BOOL
Sts_Executing	Executing state of Instruction	BOOL
Sts_Output	Output is ON (Ready)	BOOL
Sts_StopMonitorDelayActive	Stop Monitor Delay is Active	BOOL
Sts_eReady	0: Ready 1: Safety Demand 2: Reset Required 3: Suspended (Status 1 & 3 are not applicable for SS1)	SINT
Sts_eNotReady	0: Ready 1: Fault Present 2: Diagnostic Present 9: Reset Required	SINT
Val_ActualSpeed	Actual Speed Value from SFX Instruction	REAL
Val_ActualSpeedMax	Calculated Value to Plot Trend	REAL
Val_DiagnosticCode	Diagnostic codes for different diagnostic condition	DINT
Val_FaultType	SS1 fault codes	DINT
Val_Scaled_ActualSpeed	Calculated Value to Plot Trend	REAL
Val_SpeedLimit	Speed Limit value from SS1	REAL
Val_SS1StandstillSpeed	Standstill speed value from SS1	REAL
EnableOut	Enable Output - System Defined Parameter	BOOL

Data Types

The following Safety Common Control Interface tags are the primary device program tags to read and write to when interfacing to safety devices. The value of using these tags in your specific application code is that you may use a number of different safety Instructions such as SS2, SS1, etc without having to update your application device interface tags.

Refer to the Interfaces section for detailed information on interfaces.

raC_UDT_Itf_SafetyDevice_Inf

Member	Description	Data Type
Type	Provides device object type information e.g. raC_Dvc_SafetyGroupMonitor, raC_Dvc_SS1, raC_Dvc_SS1TL, etc.	STR0032
DisplayName	Provides object display name	STR0024
eClass	Provide instruction type information eClass ID: Instruction Type 0: DCS 1: DCSTL 2: DCSTM 3: CROUT 5: SafetyGroupMonitor 6: TSAM 7: TSSM 8: FSBM 9: THRSe 10: SMAT 11:SS2 12: SS1 13:SFX 14:STO	DINT

raC_UDT_Itf_SafetyDevice_Sts

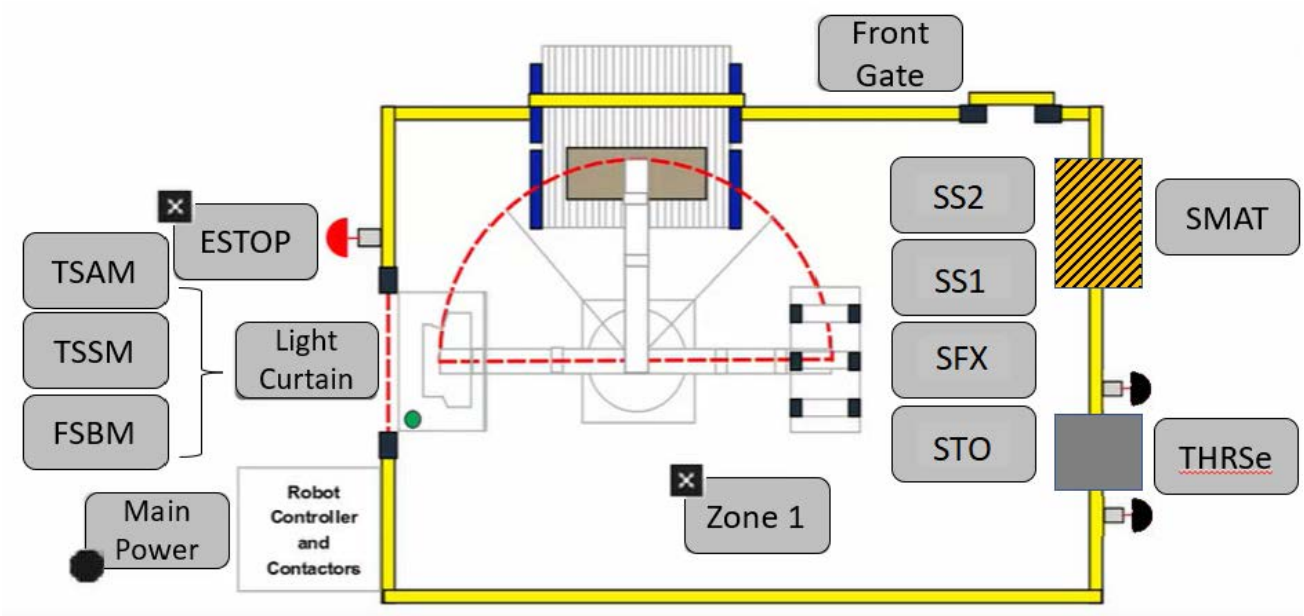
This is the Safety Device Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data. The table below shows member names, descriptions, and tag data types.

Input	Description	Data Type
bSts	Bit overlay (Visible) covering all subsequent Boolean members.	DINT
DeviceSuspended	Device suspended status (Inp_Suspend = 1) Note:- For SS1 this value is always zero.	BOOL
SafetyDemand	Safety demand status ON (1): Safety demand is present on the device object OFF (0): No safety demand Note:- For SS1 this value is always zero.	BOOL
ResetRequired	Reset Required status ON (1): Device requires reset signal OFF (0): No reset required	BOOL

Input	Description	Data Type
DiagnosticPresent	Diagnostic status ON (1): A diagnostic is present in the device object OFF (0): Device is operating normally	BOOL
FaultPresent	Fault Status ON (1): A fault is present in the device object OFF (0): Device is operating normally	BOOL
Muted	Device Muted status (Available for TSAMTM) ON (1): Muted status is present on one or more safety instructions within a specific safety group OFF (0): None of the instructions within group is muted Note:- For SS1 this value is always zero.	BOOL

Programming Example

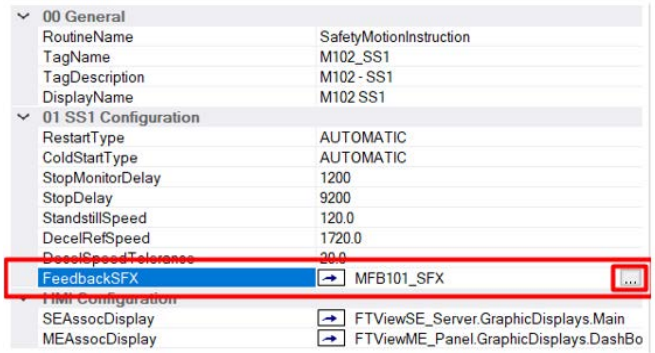
Fully configured device routine is provided below for reference. This example is taken from the ***Safety Program - SafetyMotionInstruction* routine** in the *SafetytApplication.ACD* example application provided in the library.

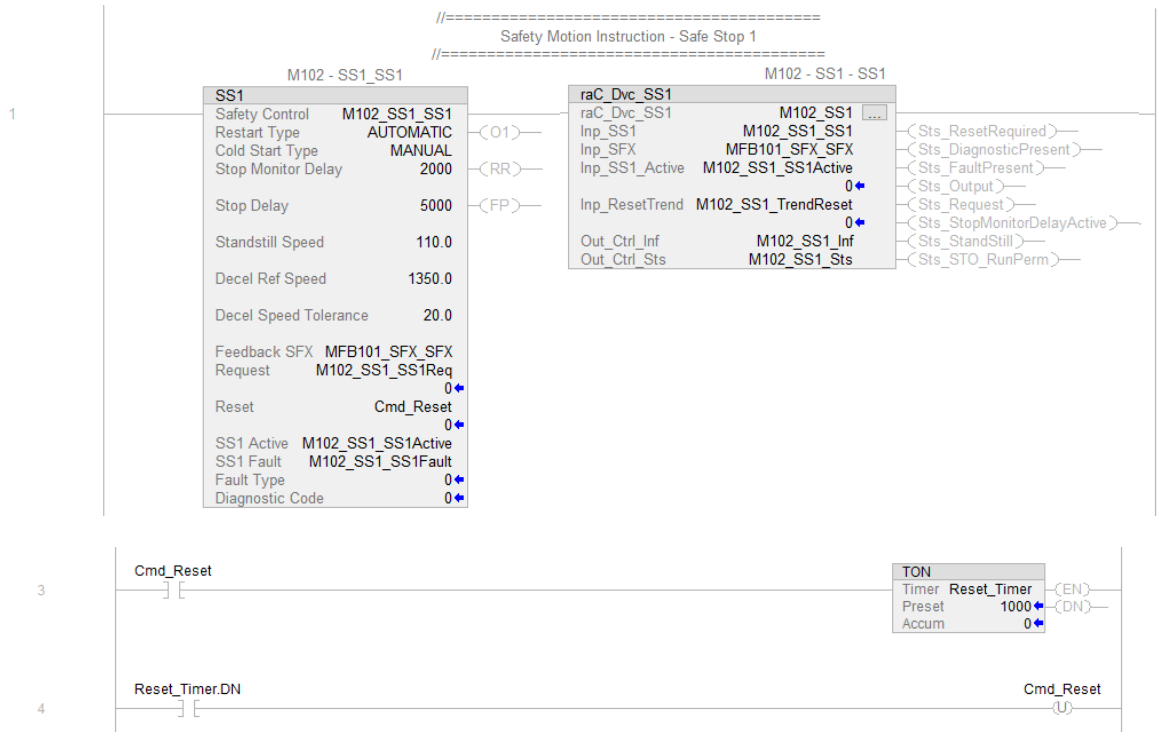


This example uses the SS1 instruction to generate or process the standstill output and STO Request Output irrespective of SS1 Request. In the example application this system is referred to as Robot_Area. The SS1 instruction is applicable in terms of stopping method without SOS monitoring System. The change in Speed is captured and plotted on trend faceplate with parameters like Actual Speed, Speed Limit and Standstill Speed. Rung descriptions are as follows:

- Rung 1: Configure SS1 and raC_Dvc_SS1 instructions
- Rung 3: Common Cmd_Reset Rung for SFX, SS2 and SS1
- Rung 4: Common Unlatch Cmd_Reset for SFX, SS2 and SS1

*Note: - when importing the supplied ROUTINE.L5X files for SS1 user can get to see that **Feedback SFX** tag is undefined tag. User can assign Feedback SFX tag from existing SFX or can initiate new SFX and assign that to SS1. when using Application Code Manager the provision is given to select the SFX tag, user can browse for SFX and assign. as shown in image below.*





Note that this programming example is the same code that is imported when either importing the supplied ROUTINE.L5X files or when using Application Code Manager or the Studio 5000 Import Library Objects wizard plug-in.

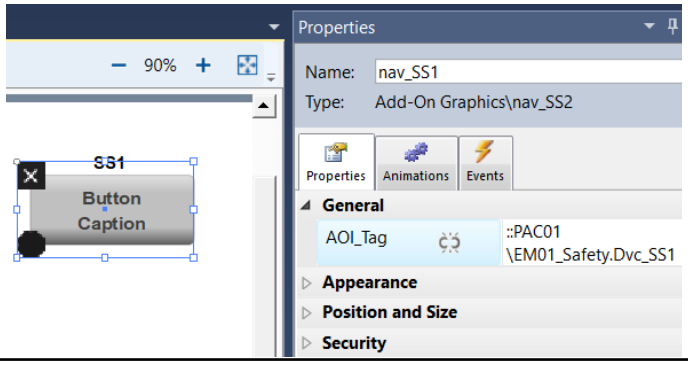
Graphic Symbols

Graphic Symbols are used as launch buttons within HMI applications to open up faceplate displays. See Basic Launch Button Attributes section for details on configuration and indicators

FactoryTalk® View ME/SE Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Global Object Parameter Values
btn_nav_SS1_SS2		Faceplate navigation button with string tag label. This launch button graphic object allows the user to navigate to the device object faceplate. The text on the button face is set to the tag's .Inf_DisplayName.	#101: AOI Instance (e.g. {::[PAC]Program::SafetyProgram._InstanceName})

Studio 5000 View Designer® Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Property Configuration
Launch		The supplied launch button in View Designer is used to navigate to the faceplate in a user application.	

Faceplates

There are basic faceplate attributes that are common across all instructions. See Basic Faceplate Attributes on page 27.

Home

The Home tab is the main tab of the faceplate. It provides the status of the Safe Stop 1 Object in the Ready banner at the top of the faceplate. The Ready banner has six states of Condition **SS1 is OFF Scan**, **Executing & Request**, **Not Ready** (Faulted), **Executing (Ready)** and **Reset Required, SS1 Device Object is OFF Scan.**

1

2

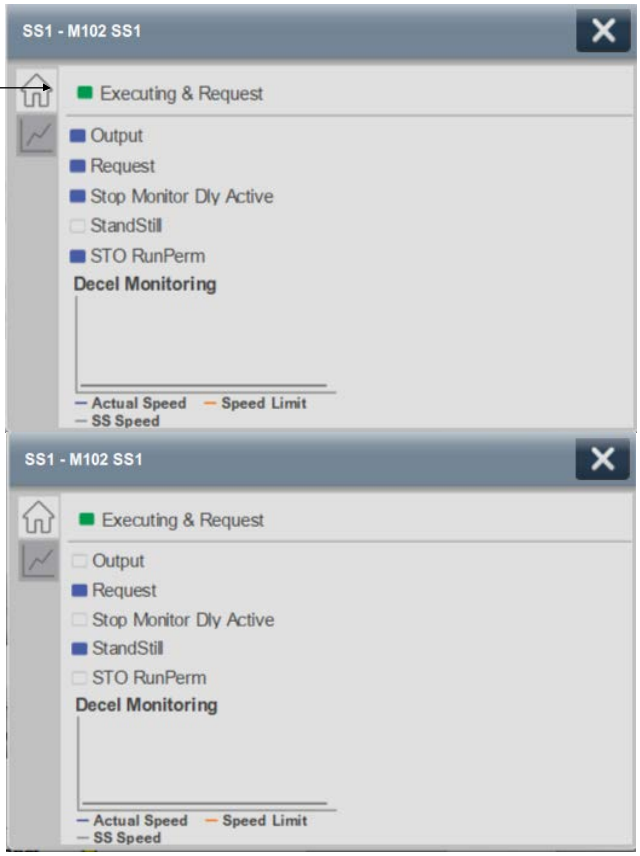
Item	Description
1	Banner- Executing State (Ready)
2	Banner- Not Ready Status

SS1 is OFF Scan refers to the situation when the execution of the rung containing the SS1 instruction is taken off scan or when the rung execution is disabled.

3

Executing & Request is the State where the SS1 request is present and it is processing. As soon as SS1.O1 (Output) is OFF, the faceplate displays Standstill indicator is ON and STO RunPerm is OFF.

4

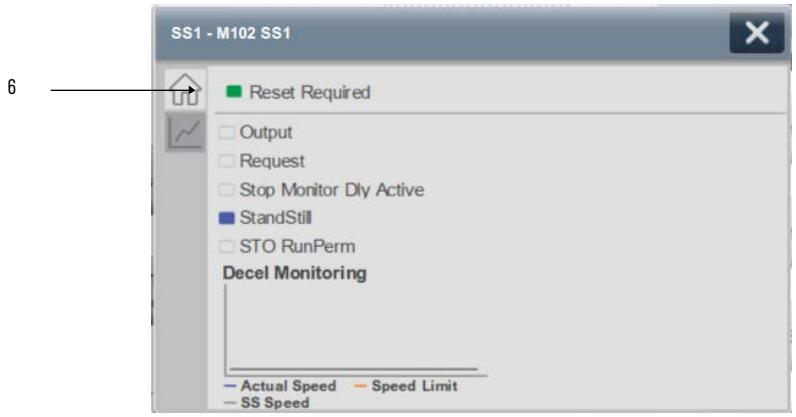


Item	Description
3	Banner- SS1 is OFF Scan
4	Banner- Executing & Request Status

SS1 Device Object is OFF Scan refers to the situation when the execution of the rung containing the SS1 AOI is taken off scan or when the rung execution is disabled.

Item	Description
5	Banner- SFX Device Object is OFF Scan

Reset Required is the state when SS1 request is not present, SS1 Active is OFF, Faults are cleared, Output of SS1 is also OFF and *M102_SS1_SS1.RR* is ON. This condition is observed when the controller is switched from program to run mode after downloading the program or if the instruction is idle for very long period of time.



Item	Description
6	Banner- Reset Required

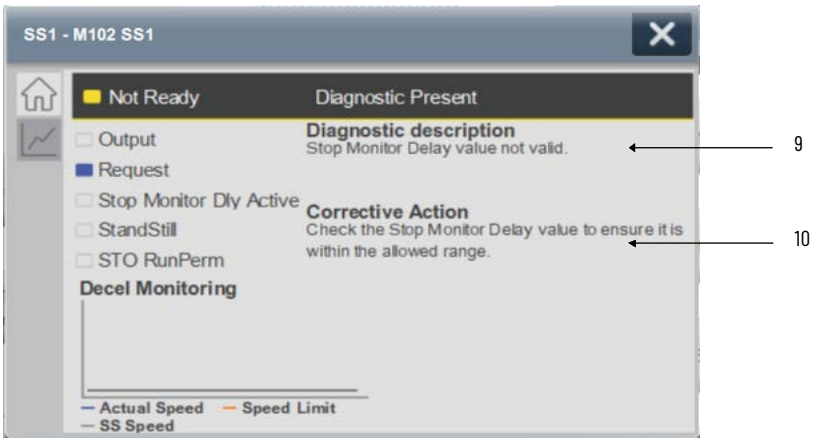
Home tab screen also provides fault and diagnostic information with required

corrective actions on the right plane of the faceplate

When there is fault present, banner with black background is displayed on top of home screen with **Not Ready** and **Fault Present** text.

Item	Description
7	Fault Present - Fault Description
8	Fault Present - Fault Corrective Action

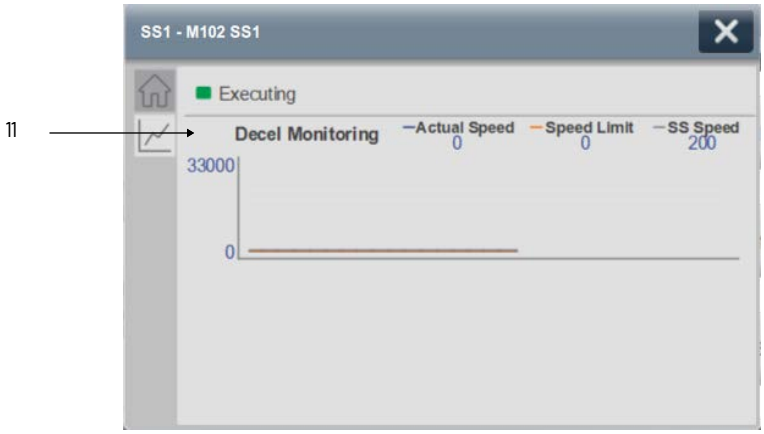
When there is diagnostic present, banner with black background is displayed on top of home screen with **Not Ready** and **Diagnostic Present** text.



Item	Description
9	Diagnostic Present - Diagnostic Short Description
10	Diagnostic Present - Diagnostic Corrective Action

Trend

The Trend tab is the Second tab of the faceplate. It is basically a data monitoring or the graphical representation of operational data within SS1 Instruction. The **Decel Monitoring** trend is plotted on the trend tab.



Item	Description
11	Decel Monitoring Trend (Plotting Trend for Actual Speed, Speed Limit and Standstill Speed)

Application Code Manager

Refer to the section Using Application Code Manager for complete details.

Definition Object: raC_Dvc_SS1

This object contains the AOI definition and used as linked library to implement object. This gives flexibility to choose to instantiate only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Implementation Object: raC_LD_Dvc_SS1

Parameter Name	Default Value	Instance Name	Definition	Description
RoutineName	{ObjectName}	{RoutineName}	Routine	Enter Routine name. Routine will be created and Object implement rung(s) inserted. A JSR will be inserted in MainRoutine. If routine name already exists, then object will be inserted into existing routine. By default, parameter is set to Object Name.
TagName	{ObjectName}	{TagName}	Backing Tag	Enter the backing tag of the main AOI. This will serve as the base tag name for other tags in this object that are derived from the base.

Parameter Name	Default Value	Instance Name	Definition	Description
TagDescription	{ObjectDescription}	{TagDescription}		Tag Description of the main AOI backing tag
DisplayName	[ObjectName]	[DisplayName]		Faceplate title/group event list description. Please avoid underscore ("_") at the end.
RestartType	AUTOMATIC	{RestartType}		Automatic or Manual Selection for RestartType
ColdStartType	AUTOMATIC	{ColdStartType}		Automatic or Manual Selection for ColdStartType
StopMonitorDelay	1000	{StopMonitorDelay}		User Entered Value for Stop Monitor Delay (msec)
StopDelay	9000	{StopDelay}		User Entered Value for Stop Delay (msec)
StandstillSpeed	100.0	{StandstillSpeed}		User Entered Value Standstill Speed
DecelRefSpeed	1710.0	{DecelRefSpeed}		User Entered Value Decel Ref Speed
DecelSpeedTolerance	10.0	{DecelSpeedTolerance}		User Entered Value Decel Ref Tolerance
FeedbackSFX		{FeedbackSFX}		Safety Feedback Interface Data

Note that if the tag names are manually entered or not linked to input channel tags a red 'X' will be shown beside the parameter. This is acceptable and the program can still be generated.

Linked Libraries

Link Name	Catalog Number	Revision	Solution	Category
raC_Dvc_SS1	raC_Dvc_SS1	1	(RA-LIB) Device	Safety

Configured HMI Content

HMI Content	Instance Name	Description
Launch Button	{ObjectName}_GO_LaunchFP	Global Object configured callout instance

Attachments

Name	Description	File Name	Extraction Path
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols SE	(raC-1-SE) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewSE\Global Objects
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols ME	(raC-1-ME) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewME\Global Objects
V1_raC_Dvc_SS1	Faceplate SE	(raC-1_xx-SE) raC_Dvc_SS1-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
V1_raC_Dvc_SS1	Faceplate ME	(raC-1_xx-ME) raC_Dvc_SS1-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
V1_raC_Dvc_SS1	View Designer	(raC-1_xx-VD) raC_Dvc_Safety.vpd	{ProjectName}\Visualization\ViewDesigner
V1_RM_raC_DvcSafety	Reference Manual	DEVICE-RM500C-EN-P.pdf	{ProjectName}\Documentation
V1_Safety_Images	HMI Image Set	Safety_Images.zip	{ProjectName}\Visualization\Images

SS2 - Safe Stop 2 (raC_Dvc_SS2)

Overview

The raC_Dvc_SS2 (Safe Stop 2) device object features a faceplate that provides an overview of the instruction's operational status. It displays information such as Output, Request, Stop Monitor Delay Active, Standstill, Stop Request, and more. Additionally, the faceplate includes trends and fault details related to the Safe Stop 2 device.

The instruction is considered ready when the following conditions are met:

- There are no SS2 (Safe Stop 2) or SOS (Safe Operating Stop) faults present.
- No diagnostics issues are detected.
- The correct reset actions have been executed.
- SS2 request is not active.
- SS2 Output is in the ON state.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following videos for this section:
"Operational_Overview_of_Safety_Device_Object_Faceplates"
"Operational_Overview_of_Safety_Application_Example_in_LogixDesigner"

Functional Description

The Safe Stop 2 pre-configured Device Objects:

- Analyze and Process Axis Data using SS2 instruction and Application Logic
- Provide Device Status & Diagnostics Faceplates for Machine Safety and Operations.
- Include Common HMI Faceplate Components, Device States, and Control Interfaces providing Application Development and Operation Efficiency

All these features provide quick feedback, shorten recovery time, and simplify implementation.

Required Files

Device Objects include Add-On Instructions (AOIs) and HMI faceplates. The revision number (e.g. 1.01) used in filenames can change as new revisions are created.

Controller Files

Add-On Instructions are reusable code objects that contain encapsulated logic that can streamline implementing your system. This lets you create your own

instruction set for programming logic as a supplement to the instruction set provided natively in the ControlLogix® firmware. An Add-On Instruction is defined once in each controller project, and can be instantiated multiple times in your application code as needed.

The Add-On Instruction must be imported into the controller project to be used in the controller configuration. These can be imported as Add-On Instruction files, or as part of the ROUTINE Import or Import Library Objects wizard.

All Add-On Instruction and ROUTINE Import files can be found in the */Studio 5000 Logix Designer® Files - L5X/* folder in the library.

Device/Item	Add-On Instruction	ROUTINE Import
SS2	raC_Dvc_SS2_1.04_AOI.L5X	raC_Dvc_SS2_1.04_ROUTINE.L5X

FactoryTalk® View HMI Files

FactoryTalk® View ME or SE applications require importing the desired device faceplates in addition to all Global Object (ggfx) files and all images located in the */HMI FactoryTalk View Images - png/* folder of the library. FactoryTalk® View ME files are stored in the */HMI - FactoryTalk View ME/* library folder and FactoryTalk® View SE files are stored in the */HMI - FactoryTalk View SE/* library folder.

Device/Item	Type	FactoryTalk® View ME Faceplate	FactoryTalk® View SE Faceplate
SS2	Display	(raC-1_04-ME) raC_Dvc_SS2-Faceplate.gfx	(raC-1_04-SE) raC_Dvc_SS2-Faceplate.gfx
SS2 - Trend	Display	(raC-1_04-ME) raC_Dvc_SS2-Trend.gfx	(raC-1_04-SE) raC_Dvc_SS2-Trend.gfx
Graphic Symbols	Global Object	(raC-1-ME) Graphic Symbols - Safety Device	(raC-1-SE) Graphic Symbols - Safety Device.ggfx
Toolbox	Global Object	(raC-1-ME) Toolbox - Safety Device.ggfx	(raC-1-SE) Toolbox - Safety Device.ggfx

Studio 5000 View Designer® HMI Files

All Studio 5000 View Designer® Files can be found in the */HMI - ViewDesigner - vpd/* folder of the library.

Device/Item	Studio 5000 View Designer® Faceplate
SS2	(raC-1_04-VD) raC_Dvc_Safety.vpd

Studio 5000® Application Code Manager Files

Studio 5000® Application Code Manager (ACM) can be optionally used if it is installed. All devices can be easily registered in the ACM repositories by running the *setup.cmd* file located in the root folder of the library.

Individual HSL4 files are provided as an alternative to running the setup.cmd to allow users to manually register specific implementation objects. Each object has two files - an Asset Control file and a Device file. The Asset Control files include attachments of all required files for that object. The Device files are used to actually add that device into a Studio 5000 project and these reference the Asset Control files.

All Studio 5000® Application Code Manager files can be found in the /*ApplicationCodeManagerLibraries*/ folder of the library. The files included are as follows:

Implementation Object	Asset Control File (.HSL4)	Device File (.HSL4)
SS2	(RA-LIB)_Device_Asset-Control_Safety_raC_Dvc_SS2_(1.4)	(RA-LIB)_Device_Device_Safety_raC_LD_SS2_(1.4)

Operations

The Safety motion monitoring objects provides only physical operation mode. There is no virtual device mode offered.

Execution

The following table explains the handling of instruction execution conditions.

Condition	Description
EnableIn False (false rung)	Processing for EnableIn False (false rung) is handled the same as if the Execution of the rung which contains SS2 Instruction and AOI were removed by Command or Logical Condition, the Instruction and AOI outputs are de-energized and the object is shown as SS2 is OFF Scan on the HMI. All the Status bits "_bsts.xx" are also cleared to zero.
Powerup (prescan, first scan)	On prescan, any commands that are received before first scan are discarded. The device object is de-energized. On first scan, the Object Common Services, Internal control Enable in true and Internal control Enable in False bits are de-energized. All the Status bits "_bsts.xx" and ONS bits are cleared to zero.
Postscan	No SFC Postscan logic is provided.

Add-On Instruction I/O Data

InOut Data

InOut	Function / Description	DataType
Inp_SS2	Input Interface - SS2 Instruction	SAFE_STOP_2
Inp_SFX	Input Interface - SFX Instruction	SAFETY_FEEDBACK_INTERFACE
Out_Ctrl_Inf	Information Interface	raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	Status Interface	raC_UDT_Itf_SafetyDevice_Sts

Input Data

Input	Function/Description	DataType
EnableIn	Enable Input - System Defined Parameter	BOOL
Inp_SS2_Active	Active Status of SS2	BOOL
Inp_SOS_Active	Active Status of SOS	BOOL
Inp_SOS_Standstill	Standstill Condition Status	BOOL
Cfg_TrendReset_SpdPercent	Set the percentage of run speed to reset the trend Y-axis. Set to a level that speed variations while stopped do not reset the Y-axis of trend. Default 1%.	REAL
Inp_ResetTrend	1 = Reset Decel Trend's Y-Axis scale to zero.	BOOL

Output Data

Output	Function/Descritpion	DataType
raC_Dvc_SS2	Unique Parameter Name for auto - discovery	BOOL
Sts_ResetRequired	Reset is required	BOOL
Sts_DiagnosticPresent	Diagnostic is present	BOOL
Sts_FaultPresent	Fault is present	BOOL
Sts_Request	SS2 Request is ON	BOOL
Sts_Active	Executing & Request State	BOOL
Sts_eReady	0: Ready 1: Safety Demand 2: Reset Required 3: Suspended (Status 1 & 3 are not applicable for SS2)	SINT
Sts_Executing	Executing state of Instruction	BOOL
Sts_eNotReady	0: Ready 1: Fault Present 2: Diagnostic Present 9: Reset Required	SINT
Sts_Standstill	Standstill output is ON	BOOL
Sts_StopRequest	Stop Request output is ON	BOOL
Sts_Output	Output is ON	BOOL
Sts_StopMonitorDelayActive	Stop Monitor Delay is Active	BOOL
Sts_SS2_Fault	SS2 Faults are Present	BOOL
Sts_SOS_Fault	SOS Faults are Present	BOOL
Val_ActualPosition	Actual Position Value from SFX Instruction	REAL
Val_ActualSpeed	Actual Speed Value from SFX Instruction	REAL
Val_ActualSpeedMax	Calculated Value to Plot Trend	REAL
Val_DiagnosticCode	Diagnostic codes for different diagnostic condition	SINT
Val_Mid_Pos	Calculated Value for Y-axis Scale on Trend	REAL
Val_Mode	Modes :- 1= Position Check, 2= Speed Check	DINT
Val_Scaled_ActualPosition	Calculated Value to Plot Trend	REAL
Val_Scaled_ActualSpeed	Calculated Value to Plot Trend	REAL
Val_Scaled_StandStillDeadband	Calculated Value to Plot Trend	REAL
Val_SOS_SS_Speed_PLimit	Calculated Value to Plot Trend	REAL
Val_SOSFaultType	SOS fault codes	SINT
Val_SOSStandStillSpeed_NLimit	Calculated Value to Plot Trend	REAL
Val_SOSStandStillSpeed_PLimit	Calculated Value to Plot Trend	REAL
Val_SpeedLimit	Speed Limit value from SS2	REAL

Output	Function/Descritpion	DataType
Val_SS2FaultType	SS2 fault codes	SINT
Val_SS2StandstillSpeed	Standstill speed value from SS2	REAL
EnableOut	Enable Output - System Defined Parameter	BOOL

Data Types

The following Safety Common Control Interface tags are the primary device program tags to read and write to when interfacing to safety devices. The value of using these tags in your specific application code is that you may use a number of different safety Instructions such as SS2, SS1, etc without having to update your application device interface tags.

Refer to the Interfaces section for detailed information on interfaces.

raC_UDT_Itf_SafetyDevice_Inf

Member	Description	Data Type
Type	Provides device object type information e.g. raC_Dvc_SafetyGroupMonitor, raC_Dvc_SS2, raC_Dvc_SS2TL, etc.	STR0032
DisplayName	Provides object display name	STR0024
eClass	Provide instruction type information eClass ID: Instruction Type 0: DCS 1: DCSTL 2: DCSTM 3: CROUT 5: SafetyGroupMonitor 6: TSAM 7: TSSM 8: FSBM 9: THRSe 10: SMAT 11:SS2 12: SS1 13:SFX 14:STO	DINT

raC_UDT_Itf_SafetyDevice_Sts

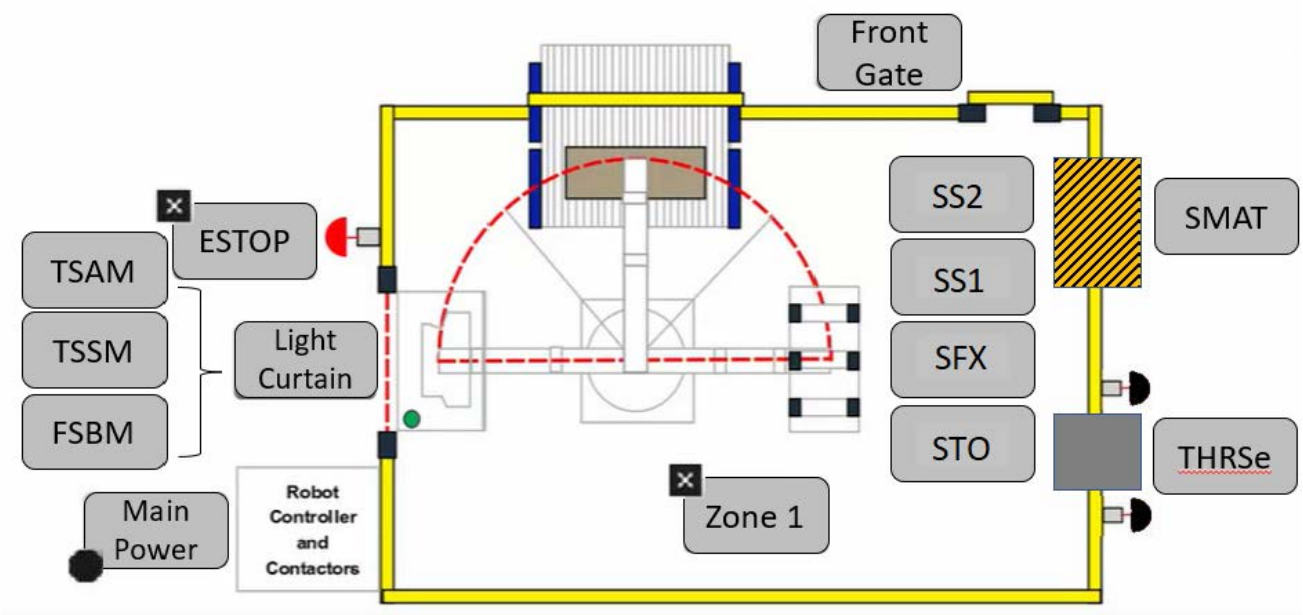
This is the Safety Device Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data.

Input	Description	Data Type
bSts	Bit overlay (Visible) covering all subsequent Boolean members.	DINT
DeviceSuspended	Device suspended status (Inp_Suspend = 1) Note:- For SS2 this value is always zero.	BOOL
SafetyDemand	Safety demand status ON (1): Safety demand is present on the device object OFF (0): No safety demand Note:- For SS2 this value is always zero.	BOOL
ResetRequired	Reset Required status ON (1): Device requires reset signal OFF (0): No reset required	BOOL

Input	Description	Data Type
DiagnosticPresent	Diagnostic status ON (1): A diagnostic is present in the device object OFF (0): Device is operating normally	BOOL
FaultPresent	Fault Status ON (1): A fault is present in the device object OFF (0): Device is operating normally	BOOL
Muted	Device Muted status (Available for TSAMTM) ON (1): Muted status is present on one or more safety instructions within a specific safety group OFF (0): None of the instructions within group is muted Note:- For SS2 this value is always zero.	BOOL

Programming Example

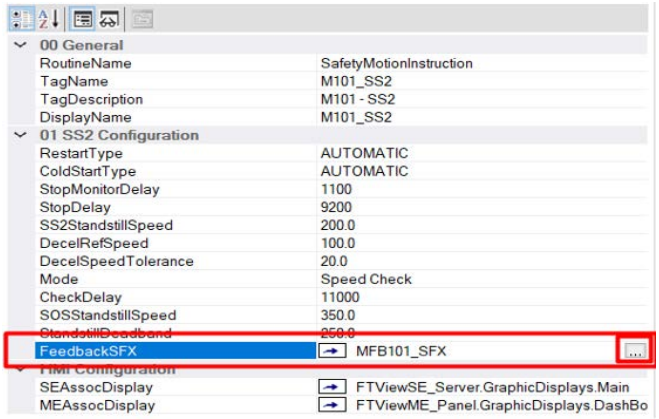
Fully configured device routine is provided below for reference. This example is taken from the ***Safety Program - SafetyMotionInstruction* routine** in the *SafetytApplication.ACD* example application provided in the library.

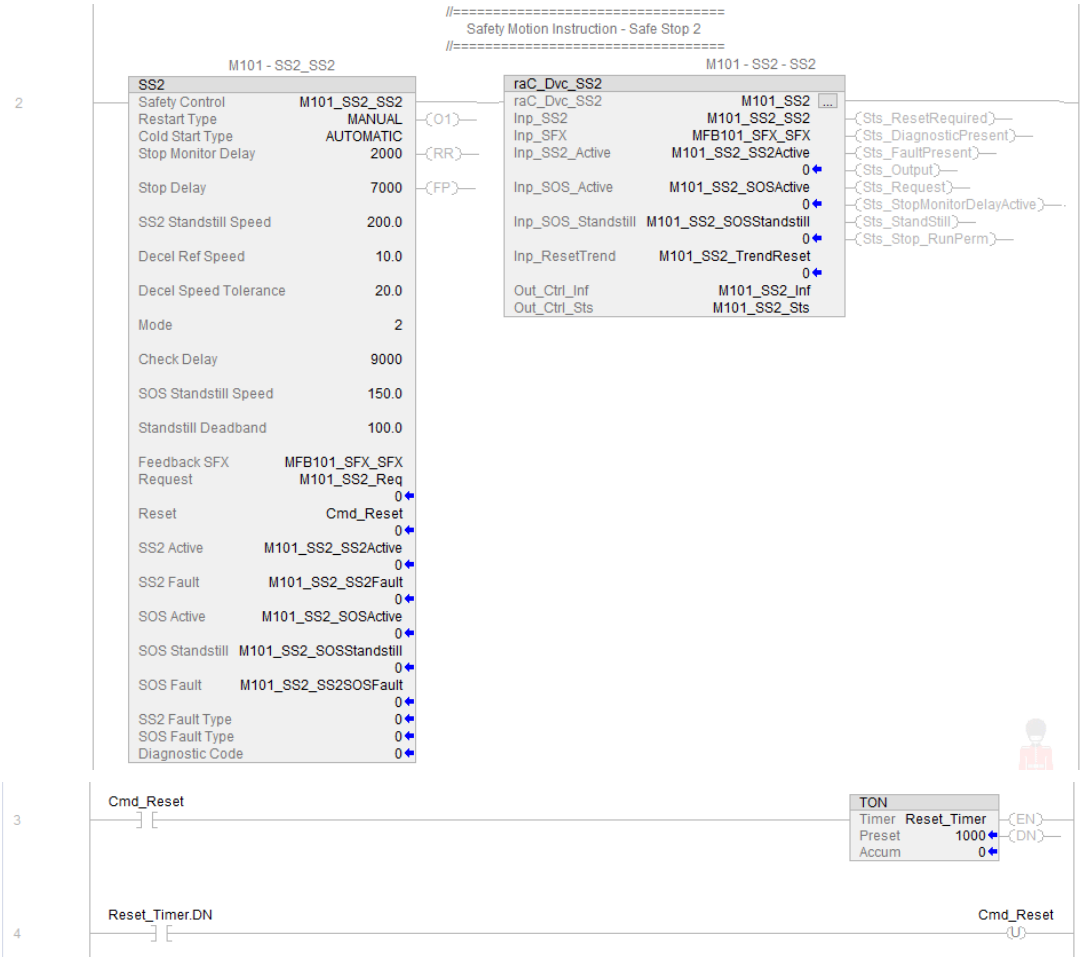


This example uses the SS2 instruction to generate or process the standstill output and Stop Request Output with respect to SS2 Request and SOS Monitoring. In the example application this system is referred to as Robot_Area. The SS2 instruction applicable in terms of stopping methods with SOS monitoring System. Actual Position trend and the difference between standstill setpoint and deadband is plotted on the trend Faceplate. Rung descriptions are as follows:

- Rung 2: Configure SS2 and raC_Dvc_SS2 instructions
- Rung 3: Common Cmd_Reset Rung for SFX, SS2 and SS1
- Rung 4: Common Unlatch Cmd_Reset for SFX, SS2 and SS1

Note: - when importing the supplied ROUTINE.L5X files for SS2 user can get to see that **Feedback SFX** tag is undefined tag. User can assign Feedback SFX tag from existing SFX or can initiate new SFX and assign that to SS2. When utilizing the Application Code Manager, you have the option to choose the SFX tag. Users can browse for the SFX tag and assign it, as illustrated in the image below.





Note that this programming example is the same code that is imported when either importing the supplied ROUTINE.L5X files or when using Application Code Manager or the Studio 5000 Import Library Objects wizard plug-in.

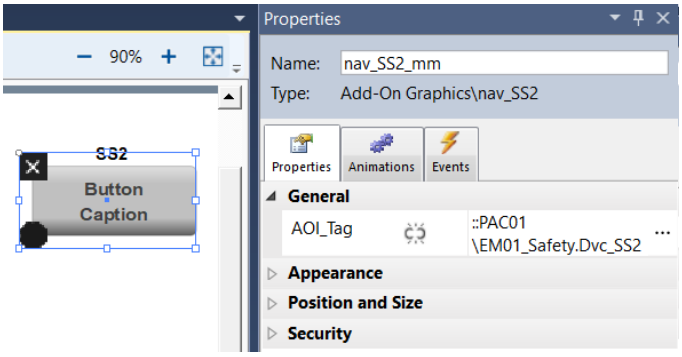
Graphic Symbols

Graphic Symbols are used as launch buttons within HMI applications to open up faceplate displays. See Basic Launch Button Attributes section for details on configuration and indicators

FactoryTalk® View ME/SE Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Global Object Parameter Values
btn_nav_SS1_SS2		Faceplate navigation button with string tag label. This launch button graphic object allows the user to navigate to the device object faceplate. The text on the button face is set to the tag's .Inf_DisplayName.	#101: AOI Instance (e.g. {::[PAC]Program::SafetyProgram._InstanceName})

Studio 5000 View Designer® Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Property Configuration
Launch		The supplied launch button in View Designer is used to navigate to the faceplate in a user application.	

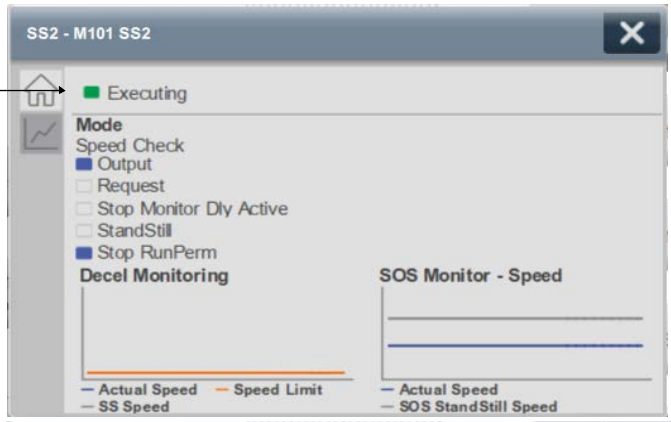
Faceplates

There are basic faceplate attributes that are common across all instructions. See Basic Faceplate Attributes on page 27.

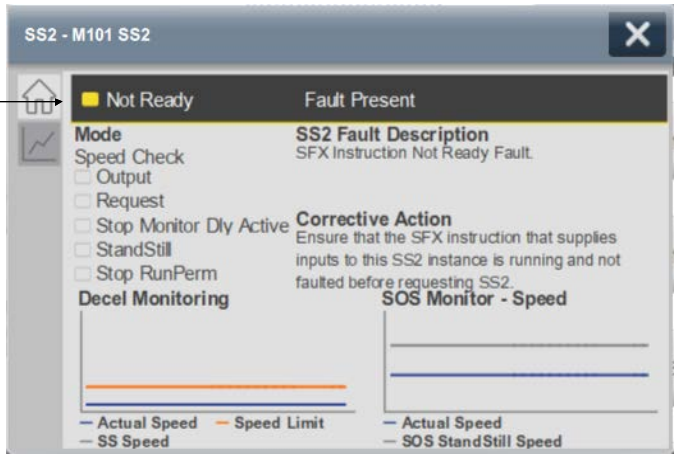
Home

The Home tab is the main tab of the faceplate. It provides the status of the Safe Stop 2 object in the Ready banner at the top of the Faceplate. The Ready banner has six states of Condition **SS2 is OFF Scan**, **Executing & Request**, **Not Ready** (Faulted), **Executing** (Ready), **Reset Required, SS2 Device Object is OFF Scan**.

1



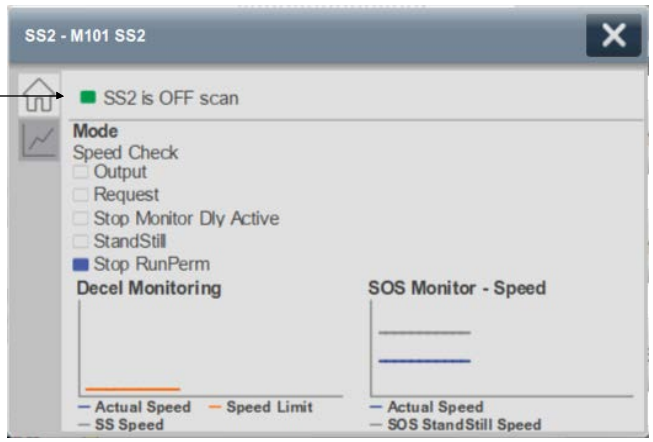
2



Item	Description
1	Banner- Executing State (Ready)
2	Banner- Not Ready Status

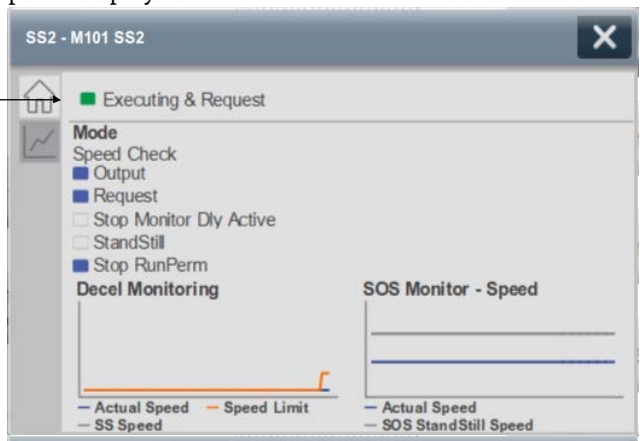
SS2 is OFF Scan refers to the situation when the execution of the rung containing the SS2 instruction is taken off scan or when the rung execution is disabled

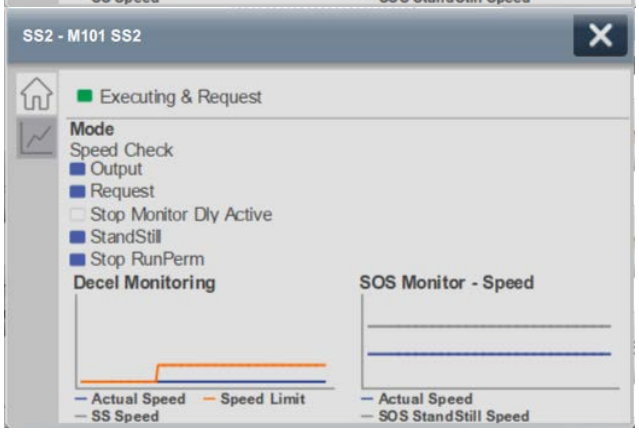
3



Executing & Request is the State where the SS2 request is present and it is processing. As soon as speed reaches SOS standstill the Standstill Output gets ON and Faceplate displays Standstill indicator is ON.

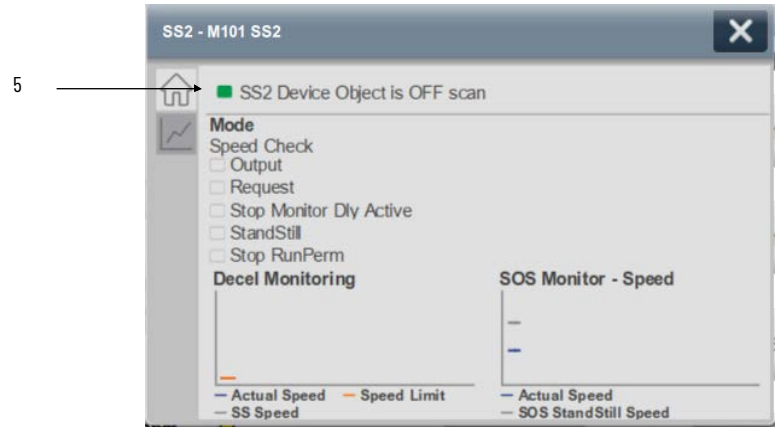
4





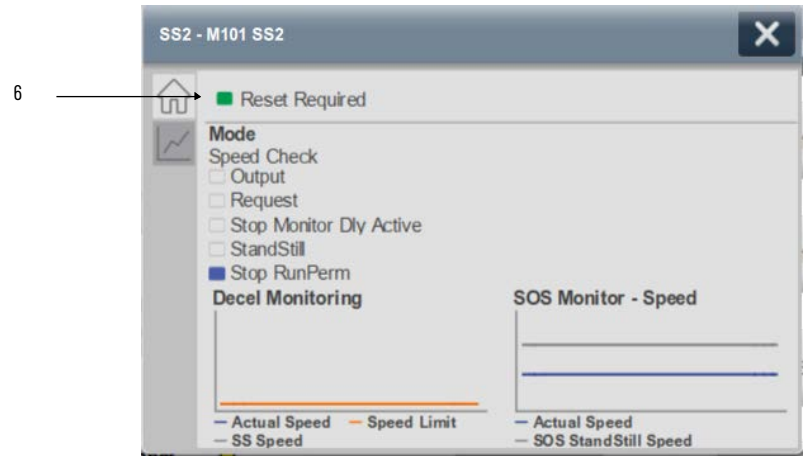
Item	Description
3	Banner- SS2 is OFF Scan
4	Banner- Executing & Request Status

SS2 Device Object is OFF Scan refers to the situation when the execution of the rung containing the SS2 AOI is taken off scan or when the rung execution is disabled



Item	Description
5	Banner- SS2 Device Object is OFF Scan

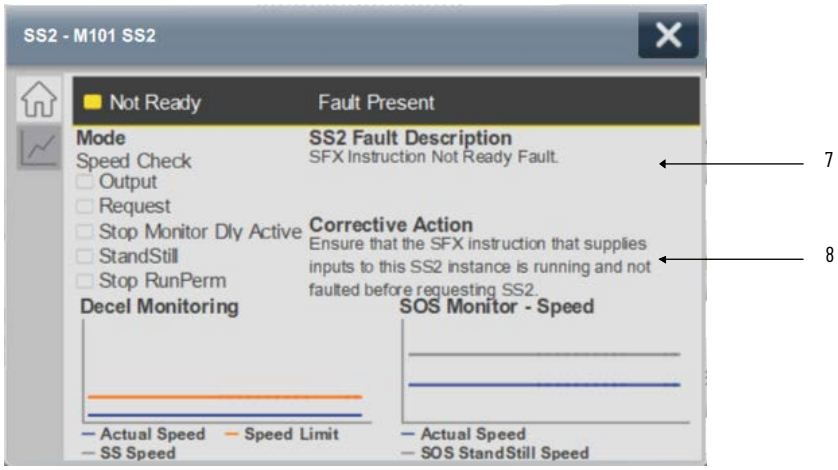
Reset Required is the state when SS2 request is not present, SS2/SOS Active is OFF, Faults are cleared, Output of SS2 is also OFF and *M101_SS2_SS2.RR* is ON. This condition is observed when the controller is switched from program to run mode after downloading the program or if the instruction is idle for very long period of time.



Item	Description
6	Banner- Reset Required

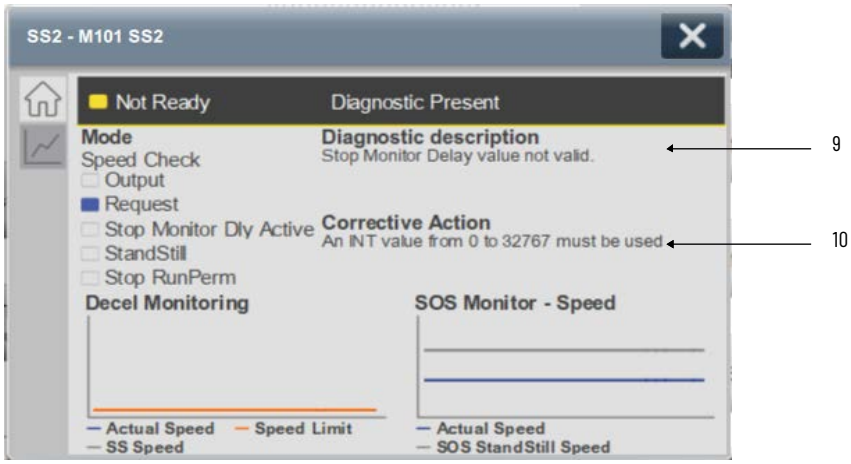
Home tab screen also provides fault and diagnostic information with required corrective actions on the right plane of the faceplate

When there is fault present, banner with black background is displayed on top of home screen with **Not Ready** and **Fault Present** text.



Item	Description
7	Fault Present - Fault Description
8	Fault Present - Fault Corrective Action

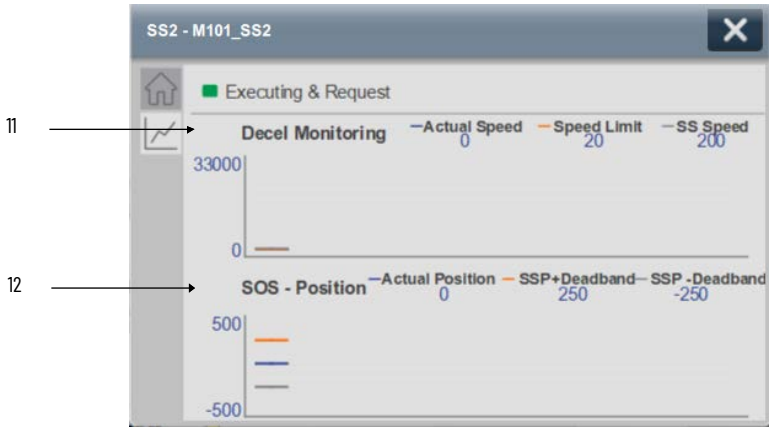
When there is diagnostic present, banner with black background is displayed on top of home screen with **Not Ready** and **Diagnostic Present** text.



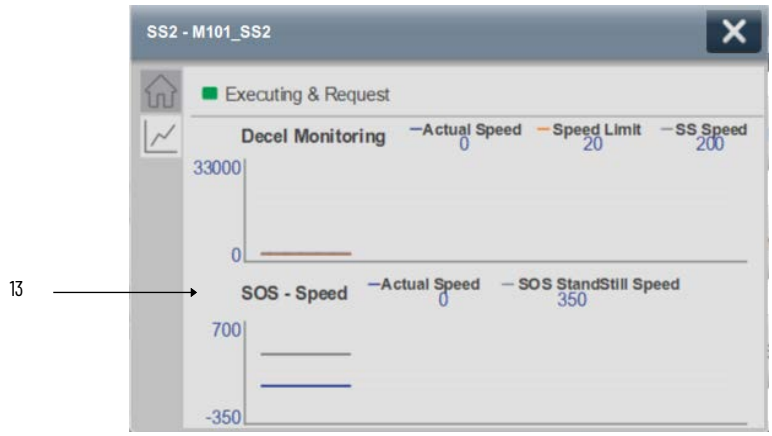
Item	Description
9	Diagnostic Present - Diagnostic Short Description
10	Diagnostic Present - Diagnostic Corrective Action

Trend

The Trend tab is the Second tab of the faceplate. It is basically a data monitoring or the graphical representation of operational data within SS2 Instruction. Totally three graphs or trends are plotted on the trend tab **Decel Monitoring**, **SOS Monitoring - Position** and **SOS Monitoring - Speed**.



Item	Description
11	Decel Monitoring Trend (Plotting Trend for Actual Speed, Speed Limit and Standstill Speed)
12	SOS - Position Monitoring Trend (Plotting Trend for Actual Position and Standstill Setpoint ± Deadband)



Item	Description
13	SOS - Speed Monitoring Trend (Plotting Trend for Actual Speed and SOS Standstill Speed)

Application Code Manager

Safety Device Library objects can be set-up and configured using Studio 5000® Application Code Manager.

Refer to the section Using Application Code Manager for complete details.

Definition Object: raC_Dvc_SS2

This object contains the AOI definition and used as linked library to implement object. This gives flexibility to choose to instantiate only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Implementation Object: raC_LD_Dvc_SS2

Parameter Name	Default Value	Instance Name	Definition	Description
RoutineName	{ObjectName}	{RoutineName}	Routine	Enter Routine name. Routine will be created and Object implement rung(s) inserted. A JSR will be inserted in MainRoutine. If routine name already exists, then object will be inserted into existing routine. By default, parameter is set to Object Name.
TagName	{ObjectName}	{TagName}	Backing Tag	Enter the backing tag of the main AOI. This will serve as the base tag name for other tags in this object that are derived from the base.
TagDescription	{ObjectDescription}	{TagDescription}		Tag Description of the main AOI backing tag
DisplayName	[ObjectName]	[DisplayName]		Faceplate title/group event list description. Please avoid underscore ("_") at the end.
RestartType	AUTOMATIC	{RestartType}		Automatic or Manual Selection for RestartType
ColdStartType	AUTOMATIC	{ColdStartType}		Automatic or Manual Selection for ColdStartType
StopMonitorDelay	1000	{StopMonitorDelay}		User Entered Value for Stop Monitor Delay (msec)
StopDelay	9000	{StopDelay}		User Entered Value for Stop Delay (msec)
SS2StandstillSpeed	100.0	{SS2StandstillSpeed}		User Entered Value for SS2 Standstill Speed
DecelRefSpeed	0.0	{DecelRefSpeed}		User Entered Value for Decel Ref Speed
DecelSpeedTolerance	10.0	{DecelSpeedTolerance}		User Entered Value for Decel Speed Tolerance

Parameter Name	Default Value	Instance Name	Definition	Description
Mode	Position Check	{Mode}		User Selection for Position Check or Speed Check
CheckDelay	9000	{CheckDelay}		User Entered Value for Check Delay (msec)
SOSStandstillSpeed	250.0	{SOSStandstillSpeed}		User Entered Value for SOS Standstill Speed
StandStillDeadband	150.0	{StandStillDeadband}		User Entered Value for Standstill Deadband
FeedbackSFX		{FeedbackSFX}		Safety Feedback Interface Data

Note that if the tag names are manually entered or not linked to input channel tags a red 'X' will be shown beside the parameter. This is acceptable and the program can still be generated.

Linked Libraries

Link Name	Catalog Number	Revision	Solution	Category
raC_Dvc_SS2	raC_Dvc_SS2	1	(RA-LIB) Device	Safety

Configured HMI Content

HMI Content	Instance Name	Description
Launch Button	{ObjectName}_GO_LaunchFP	Global Object configured callout instance

Attachments

Name	Description	File Name	Extraction Path
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols SE	(raC-1-SE) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewSE\Global Objects
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols ME	(raC-1-ME) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewME\Global Objects
V1_raC_Dvc_SS2	Faceplate SE	(raC-1_xx-SE) raC_Dvc_SS2-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
V1_raC_Dvc_SS2	Faceplate ME	(raC-1_xx-ME) raC_Dvc_SS2-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
V1_raC_Dvc_SS2	View Designer	(raC-1_xx-VD) raC_Dvc_Safety.vpd	{ProjectName}\Visualization\ViewDesigner
V1_RM_raC_DvcSafety	Reference Manual	DEVICE-RM500C-EN-P.pdf	{ProjectName}\Documentation
V1_Safety_Images	HMI Image Set	Safety_Images.zip	{ProjectName}\Visualization\Images

STO - Safe Torque OFF (raC_Dvc_STO)

Overview

The Safe Torque OFF device object (raC_Dvc_STO) includes a faceplate which displays Input and Output status of connected hardware. The Input includes the status of STO Active, Torque Disabled, Safety Fault, Reset Required and Run Permissive and the output includes the status of STO Output and Reset Request. The faceplate also displays faults and different states of Safe Torque OFF device object.

The instruction is considered ready when the following conditions are met:

- There are no faults present.
- The correct reset actions have been executed.
- STO Output is in the ON State

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following videos for this section:
"Operational_Overview_of_Safety_Device_Object_Faceplates"
"Operational_Overview_of_Safety_Application_Example_in_LogixDesigner"

Functional Description

The Safe Torque OFF pre-configured Device Objects:

- Read Input and Output Data from connected Safety Devices and generate the safe state STO output from Application Logic.
- Provide Device Status & Diagnostics Faceplates for Machine Safety and Operations.
- Include Common HMI Faceplate Components, Device States, and Control Interfaces providing Application Development and Operation Efficiency

All these features provide quick feedback, shorten recovery time, and simplify implementation.

Required Files

Device Objects include Add-On Instructions (AOIs) and HMI faceplates. The revision number (e.g. 1.01) used in filenames can change as new revisions are created.

Controller Files

Add-On Instructions are reusable code objects that contain encapsulated logic that can streamline implementing your system. This lets you create your own instruction set for programming logic as a supplement to the instruction set provided natively in the ControlLogix® firmware. An Add-On Instruction is

defined once in each controller project, and can be instantiated multiple times in your application code as needed.

The Add-On Instruction must be imported into the controller project to be used in the controller configuration. These can be imported as Add-On Instruction files, or as part of the ROUTINE Import or Import Library Objects wizard.

All Add-On Instruction and ROUTINE Import files can be found in the */Studio 5000 Logix Designer® Files - L5X/* folder in the library.

Note:- There are two separate ROUTINE files provided for STO. One for Single Axis and other is for Dual Axis. The ROUTINE can be import as per selection of number of axes. The ROUTINE files are named based on the number of axes. Refer table given below.

Device/Item	Add-On Instruction	ROUTINE Import
STO	raC_Dvc_STO_1.04_AOI.L5X	raC_Dvc_STO_SingleAxis_1.04_ROUTINE.L5X
STO	raC_Dvc_STO_1.04_AOI.L5X	raC_Dvc_STO_DualAxis_1.04_ROUTINE.L5X

FactoryTalk® View HMI Files

FactoryTalk® View ME or SE applications require importing the desired device faceplates in addition to all Global Object (ggfx) files and all images located in the */HMI FactoryTalk View Images - png/* folder of the library. FactoryTalk® View ME files are stored in the */HMI - FactoryTalk View ME/* library folder and FactoryTalk® View SE files are stored in the */HMI - FactoryTalk View SE/* library folder.

Device/Item	Type	FactoryTalk® View ME Faceplate	FactoryTalk® View SE Faceplate
STO	Display	(raC-1_04-ME) raC_Dvc_STO-Faceplate.gfx	(raC-1_04-SE) raC_Dvc_STO-Faceplate.gfx
Graphic Symbols	Global Object	(raC-1-ME) Graphic Symbols - Safety Device	(raC-1-SE) Graphic Symbols - Safety Device.ggfx
Toolbox	Global Object	(raC-1-ME) Toolbox - Safety Device.ggfx	(raC-1-SE) Toolbox - Safety Device.ggfx

Studio 5000 View Designer® HMI Files

All Studio 5000 View Designer® Files can be found in the */HMI - ViewDesigner - vpd/* folder of the library.

Device/Item	Studio 5000 View Designer® Faceplate
STO	(raC-1_04-VD) raC_Dvc_Safety.vpd

Studio 5000® Application Code Manager Files

Studio 5000® Application Code Manager (ACM) can be optionally used if it is installed. All devices can be easily registered in the ACM repositories by running the *setup.cmd* file located in the root folder of the library.

Individual HSL4 files are provided as an alternative to running the setup.cmd to allow users to manually register specific implementation objects. Each object has two files - an Asset Control file and a Device file. The Asset Control files include attachments of all required files for that object. The Device files are used to actually add that device into a Studio 5000 project and these reference the Asset Control files.

All Studio 5000® Application Code Manager files can be found in the /*ApplicationCodeManagerLibraries*/ folder of the library. The files included are as follows:

Implementation Object	Asset Control File (.HSL4)	Device File (.HSL4)
STO	(RA-LIB)_Device_Asset-Control_Safety_raC_Dvc_STO_(1.4)	(RA-LIB)_Device_Device_Safety_raC_LD_STO_(1.4)

Operations

The Safety motion monitoring objects provides only physical operation mode. There is no virtual device mode offered.

Execution

The following table explains the handling of instruction execution conditions.

Condition	Description
EnableIn False (false rung)	Processing for EnableIn False (false rung) is handled the same as if the Execution of the rung which contains STO AOI were removed by Command or Logical Condition, the AOI outputs are de-energized and the object is shown as STO is OFF Scan on the HMI. All the Status bits "_bsts.xx" are also cleared to zero. All the operational outputs and timers are set to zero.
Powerup (prescan, first scan)	On prescan, any commands that are received before first scan are discarded. The device object is de-energized. On first scan, the Object Common Services, Internal control Enable in true and Internal control Enable in False bits are de-energized. All the instruction error codes, Status bits "_bsts.xx" and ONS bits are cleared to zero.
Postscan	No SFC Postscan logic is provided.

Add-On Instruction I/O Data

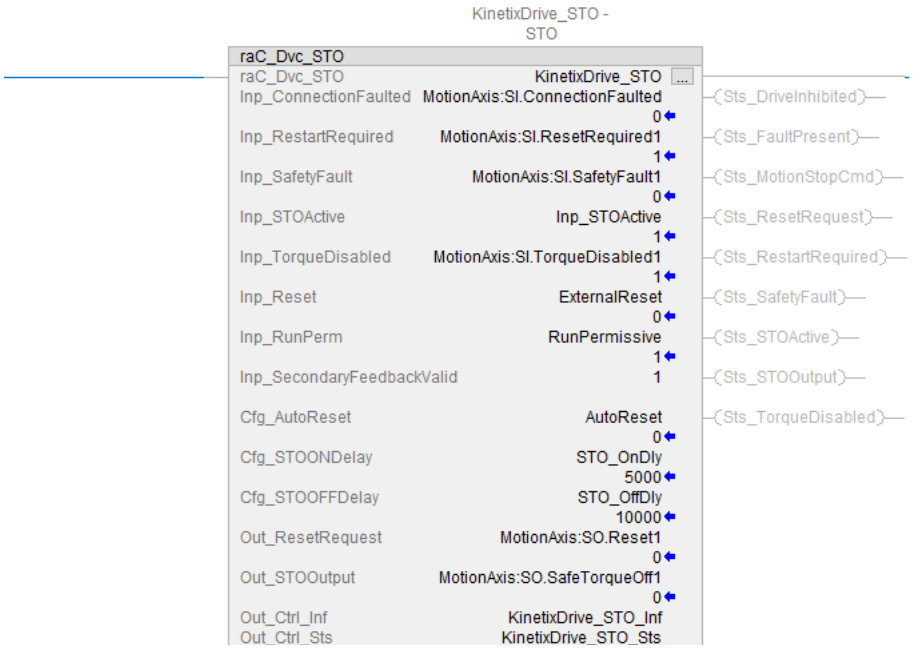
InOut Data

InOut	Function / Description	DataType
Out_Ctrl_Inf	Information Interface	raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	Status Interface	raC_UDT_Itf_SafetyDevice_Sts

Input Data

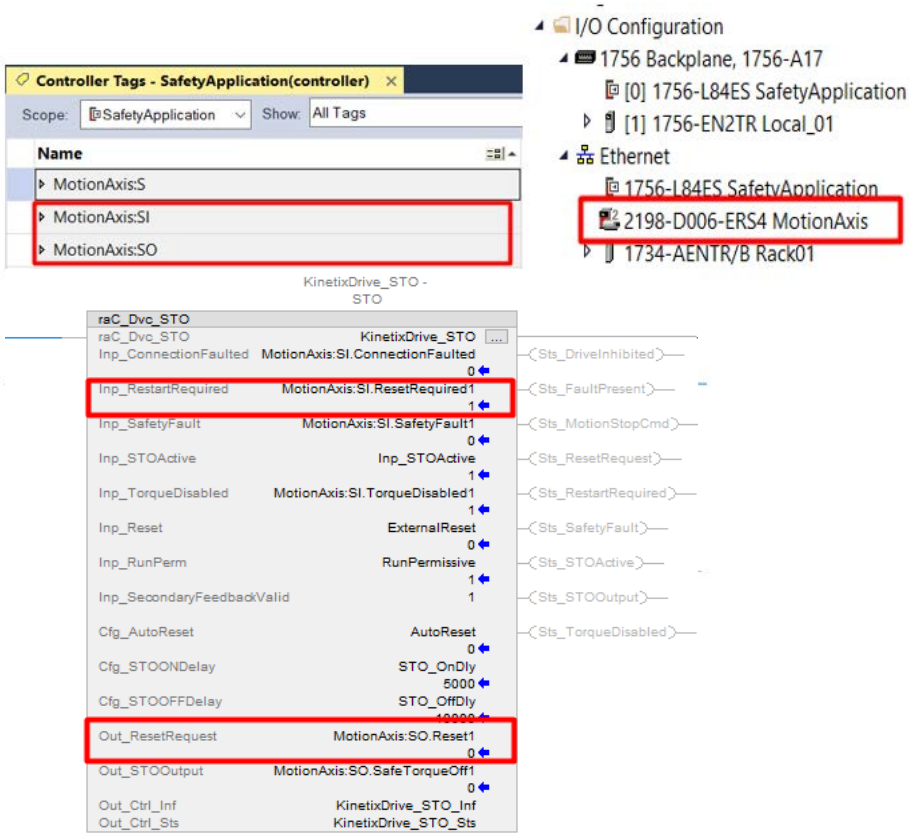
Input	Function/Description	DataType
EnableIn	Enable Input - System Defined Parameter	BOOL
Inp_STO_Active	Active Status of STO	BOOL
Inp_TorqueDisabled	Active Status of Torque Disabled	BOOL
Inp_SafetyFault	Active Status of Safety Faults	BOOL
Inp_RestartRequired	Active Status of Restart Required	BOOL
Inp_ConnectionFaulted	Active Status of Connection	BOOL
Inp_RunPerm	Run Permissive Input	BOOL
Inp_Reset	Reset Input	BOOL
Inp_SecondaryFeedback Valid	Input Secondary Feedback Valid Set to 1, if no Secondary Encoder Feedback	BOOL
Cfg_STOONDelay	STO ON Delay in msec	DINT
Cfg_STOOFFDelay	STO OFF Delay in msec	DINT
Cfg_AutoReset	1 = AutoReset is ON	BOOL

The above Input Data table is applicable for single axis device. In case of dual axis user needs import routine for dual axis and assign dual axis Input Data. Initially the Input Data values are assigned with a name **MotionAxis:SI** and **MotionAxis:SO** in the program as shown below.



User needs to add specific hardware with the name **MotionAxis** in **IO Configuration** or Add device with any name in IO Configuration and then assign module defined datatype tags of that specific device manually to the Input Data.

The Input data to the AOI will be hardware specific for that the user needs to select the hardware first in IO Configuration & then only user will able to assign the module defined tags to the AOI input data. Below is the image of STO Instance with assigned Input Data.



In Above image or example the STO is instantiated as a **KinetixDrive_STO** and the hardware kinetix drive is added in the IO configuration with the name **MotionAxis.**

After adding a hardware the module defined data types of that hardware are generated in the controller Tags and those are **MotionAxis:S**, **MotionAxis:SI** and **MotionAxis:SO.**

User needs to use **MotionAxis:SI** and **MotionAxis:SO** tags as a Input and Output data to the AOI, as shown in above image.

Note: - when using module defined tags (MotionAxis:SI) to configure STO AOI, the Secondary Feedback Valid input tag (MotionAxis:SI.SecondaryFeedbackValid) may not be available with each hardware. In that case always set **Inp_SecondaryFeedbackValid =1** to configure STO AOI correctly.

Output Data

Output	Function/Description	DataType
raC_Dvc_STO	Unique Parameter Name for auto - discovery	BOOL
Out_STOOutput	User needs to assign Module defined DataType tag to this output. 1 = STO Output is ON	BOOL
Out_Reset_Request	User needs to assign Module defined DataType tag to this output. 1 = Reset Required	BOOL
Sts_STOOutput	STO Output 1 = ON; 0 = OFF	BOOL
Sts_STOActive	1 = Safe Torque Off Active state	BOOL
Sts_ResetRequest	1 = Reset Request is Present	BOOL
Sts_SafetyFault	1 = Safety Faults Present	BOOL
Sts_TorqueDisabled	1 = Torque Disabled	BOOL
Sts_RestartRequired	1 = Restart Required	BOOL
Sts_FaultPresent	1 = Fault Present	BOOL
Sts_NotReady	1 = Not Ready	BOOL
Sts_Ready	1 = Ready	BOOL
Sts_MotionStop	1 = Stop a Motion	BOOL
Sts_eReady	0: Ready 1: Safety Demand 2: Reset Required 3: Suspended (Status 1 & 3 are not applicable for STO)	SINT
Sts_eNotReady	0: Ready 1: Fault Present 9: Reset Required	SINT
Sts_InvalidCfgSTODelay	Invalid Cfg_STODelay Error Bit (Sts_ERR = 1011)	BOOL
Sts_DriveInhibited	1 = Drive Inhibited	BOOL
Sts_ResetInProgress	1 = Safety Fault Reset is in Progress	BOOL

Data Types

The following Safety Common Control Interface tags are the primary device program tags to read and write to when interfacing to safety devices. The value of using these tags in your specific application code is that you may use a number of different safety Instructions such as SS2, STO, etc without having to update your application device interface tags.

Refer to the Interfaces section for detailed information on interfaces.

raC_UDT_Itf_SafetyDevice_Inf

Member	Description	Data Type
Type	Provides device object type information e.g. raC_Dvc_SafetyGroupMonitor, raC_Dvc_STO, etc.	STR0032
DisplayName	Provides object display name	STR0024
eClass	Provide instruction type information eClass ID: Instruction Type 0: DCS 1: DCSTL 2: DCSTM 3: CROUT 5: SafetyGroupMonitor 6: TSAM 7: TSSM 8: FSBM 9: THRSe 10: SMAT 11:SS2 12: SS1 13:SFX 14:STO	DINT

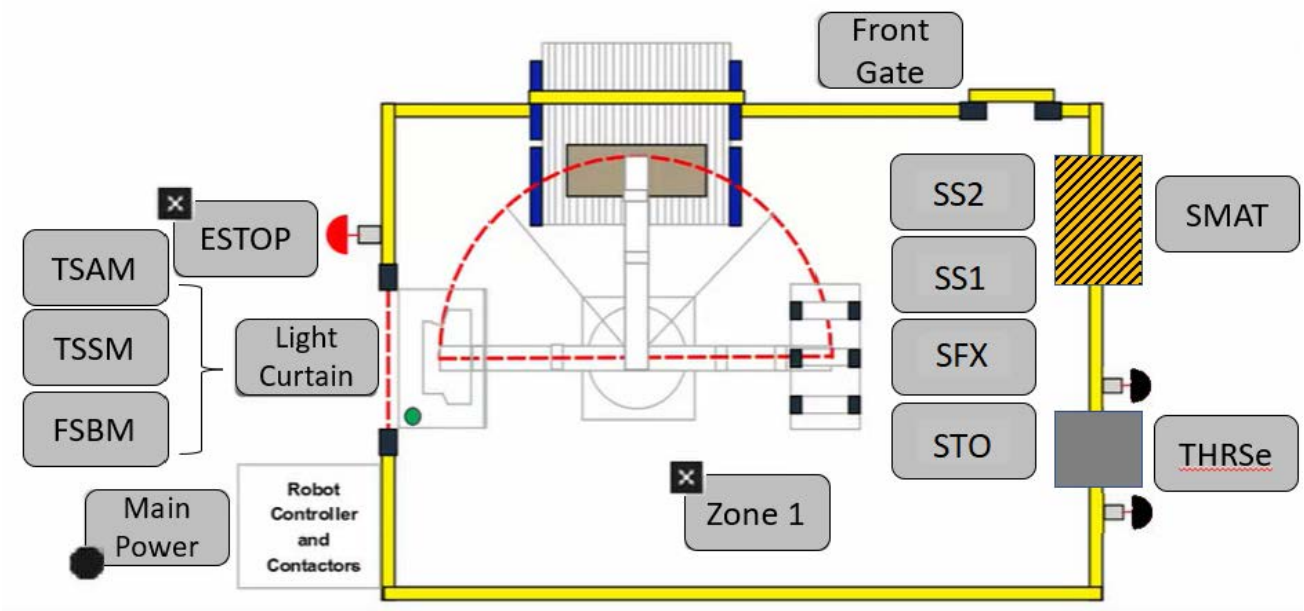
raC_UDT_Itf_SafetyDevice_Sts

This is the Safety Device Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data. The table below shows member names, descriptions, and tag data types.

Input	Description	Data Type
bSts	Bit overlay (Visible) covering all subsequent Boolean members.	DINT
DeviceSuspended	Device suspended status (Inp_Suspend = 1) Note:- For STO this value is always zero.	BOOL
SafetyDemand	Safety demand status ON (1): Safety demand is present on the device object OFF (0): No safety demand Note:- For STO this value is always zero.	BOOL
ResetRequired	Reset Required status ON (1): Device requires reset signal OFF (0): No reset required	BOOL
DiagnosticPresent	Diagnostic status ON (1): A diagnostic is present in the device object OFF (0): Device is operating normally Note:- For STO this value is always zero.	BOOL
FaultPresent	Fault Status ON (1): A fault is present in the device object OFF (0): Device is operating normally	BOOL
Muted	Device Muted status (Available for TSAMTM) ON (1): Muted status is present on one or more safety instructions within a specific safety group OFF (0): None of the instructions within group is muted Note:- For STO this value is always zero.	BOOL

Programming Example

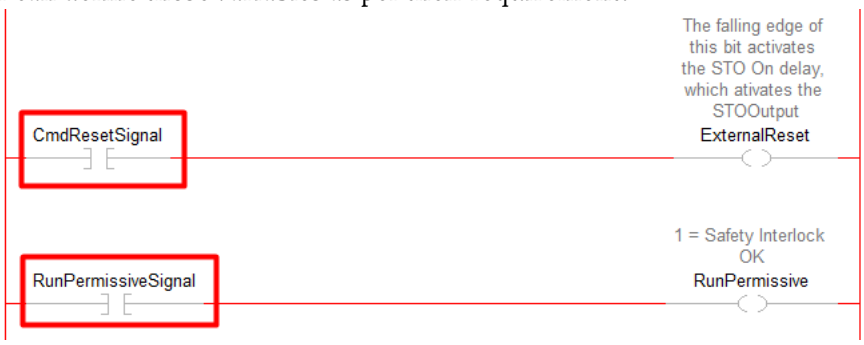
Fully configured device routine is provided below for reference. This example is taken from the ***Safety Program - SafetyMotionInstruction* routine** in the *SafetytApplication.ACD* example application provided in the library.



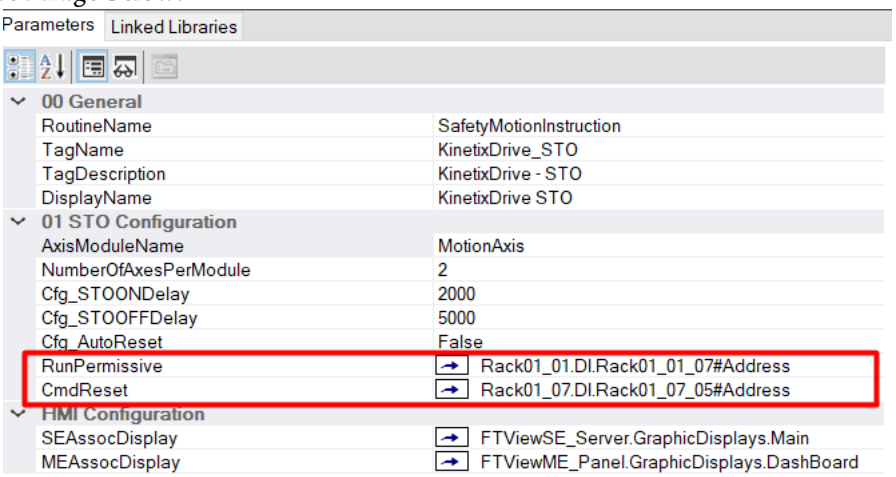
This example uses the STO device object to generate Safe Torque OFF output for the hardware as a safety function. In the example application this system is referred to as Robot_Area. The STO device object reads the input and output data of connected hardware and controls the hardware safety function by identifying the safety faults of the hardware. The faceplate displays the status like **STO is active**, **Torque disabled**, **Safety Fault**, **Reset Required**, **Run Permissive**, **Safe Toque OFF Output, Reset Request** etc. on the Home screen. Rung descriptions are as follows:

- Rung 5: External Input Reset Command
- Rung 6: External Input RunPerm (SafetyInterlocks)
- Rung 7: Configure raC_Dvc_STO instruction
- On Rung 7 User needs to assign the required input data to the raC_Dvc_STO instruction

When importing the supplied ROUTINE.L5X files, user will get to see that Rung 5 and Rung 6 is having two undefined variables. shown in image below. User can define those variables as per their requirement.



When using Application Code Manager to generate the STO device object. There is a provision to assign inputs to that undefined variables. As shown in the image below.



Note that this programming example is the same code that is imported when either importing the supplied ROUTINE.L5X files or when using Application Code Manager or the Studio 5000 Import Library Objects wizard plug-in.

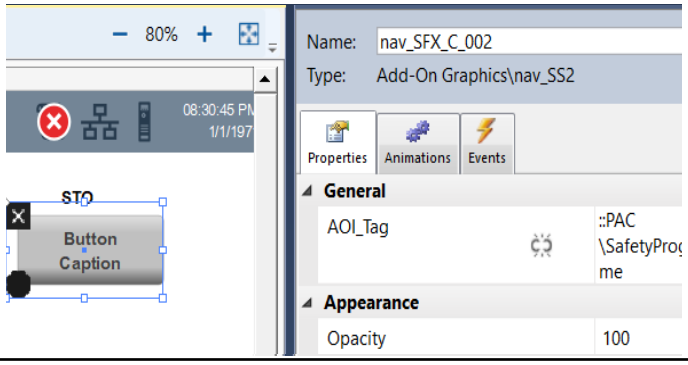
Graphic Symbols

Graphic Symbols are used as launch buttons within HMI applications to open up faceplate displays. See Basic Launch Button Attributes section for details on configuration and indicators

FactoryTalk® View ME/SE Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Global Object Parameter Values
btn_nav_STO		Faceplate navigation button with string tag label. This launch button graphic object allows the user to navigate to the device object faceplate. The text on the button face is set to the tag's .Inf_DisplayName.	#101: AOI Instance (e.g. {::[PAC]Program::SafetyProgram._InstanceName })

Studio 5000 View Designer® Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Property Configuration
Launch		The supplied launch button in View Designer is used to navigate to the faceplate in a user application.	

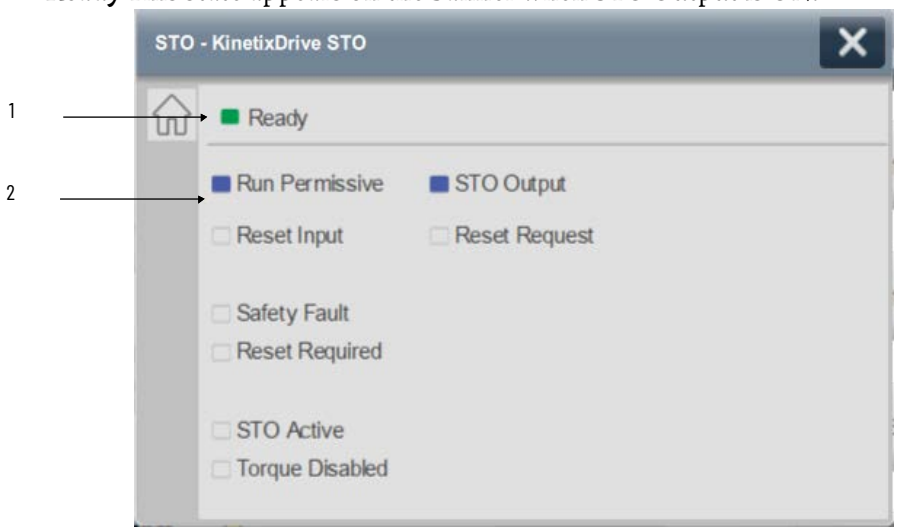
Faceplates

There are basic faceplate attributes that are common across all instructions. See Basic Faceplate Attributes on page 27.

Home

The Home tab is the main tab of the faceplate. It provides the status of the Safe Torque OFF object in the Ready banner at the top of the faceplate. The ready banner mainly has two states of Condition **Ready** and **Not Ready.**

Ready This state appears on the banner when STO Output is ON.



Item	Description
1	Banner- Ready Status
2	STO - SI and SO Status

Not Ready This State appears on the banner with nine different conditions those are listed below in the table followed by detailed explanation of each condition.

- **Fault Present - Safety Fault** faceplate displays *Fault Present - Safety Fault - Drive Inhibited* on the fault banner when Safety Fault input is ON and STO Output is OFF.

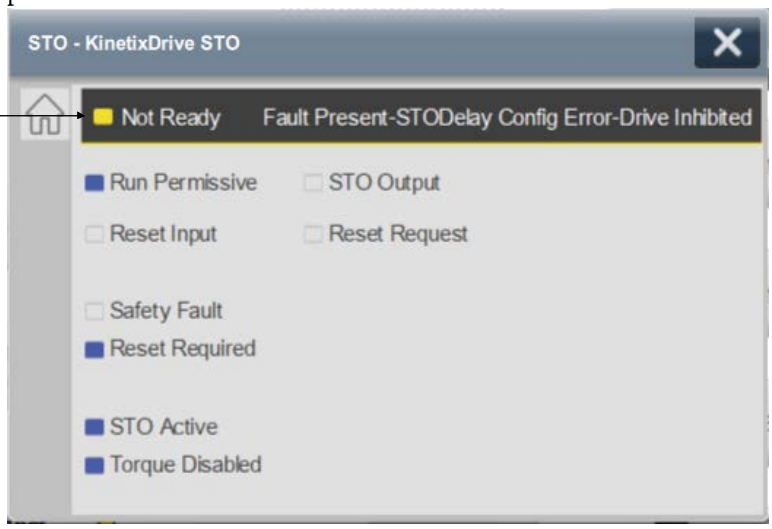
3

- **Fault Present - Connection Faulted** faceplate displays *Fault Present - Connection Faulted* on the fault banner when Connection Faulted input is ON and STO Output is OFF.

4

- **Fault Present - STO Delay Config Error** faceplate displays *Fault Present - STO Delay Config Error - Drive Inhibited* on the fault banner when user enters negative values in Cfg_STOONDelay or Cfg_STOOFFDelay parameters.

5

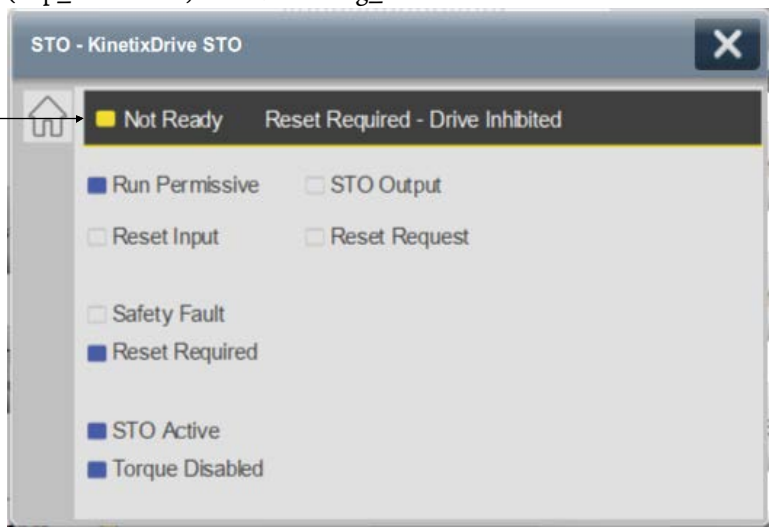


- **External STO RunPerm is OFF** faceplate displays *External STO RunPerm is OFF- Drive Inhibited* on the fault banner when faults are not present and RunPermissive input (Inp_RunPerm) is OFF.

6

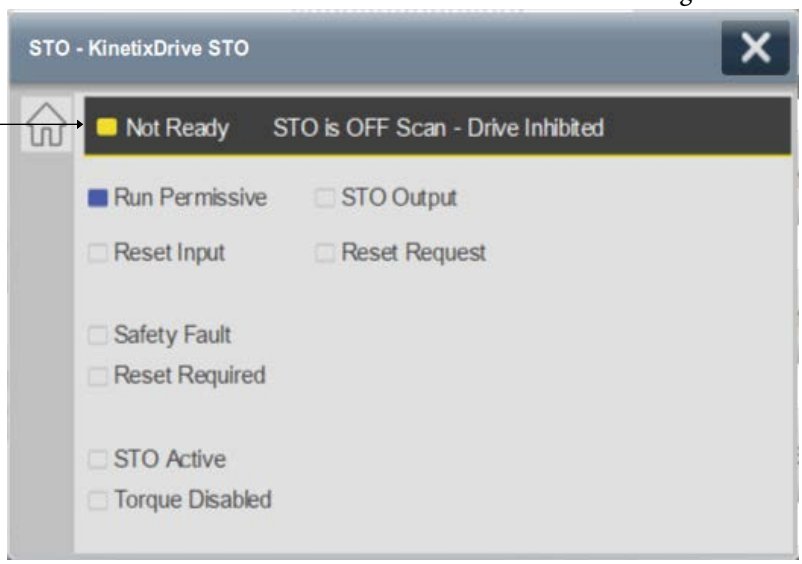
- **Reset Required** faceplate displays *Reset Required- Drive Inhibited* on the fault banner when faults are not present, RunPermissive input (Inp_RunPerm) is ON and Cfg_AutoReset = 0.

7



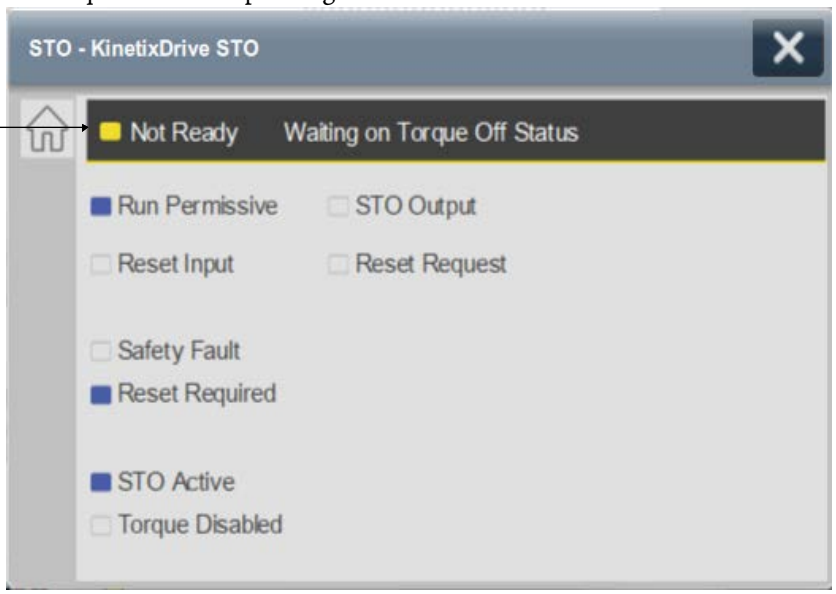
- **STO is OFF Scan** faceplate displays *STO is OFF Scan- Drive Inhibited* on the fault banner when the execution of the rung which contains STO add-on instruction is taken off scan or disabled the rung execution.

8



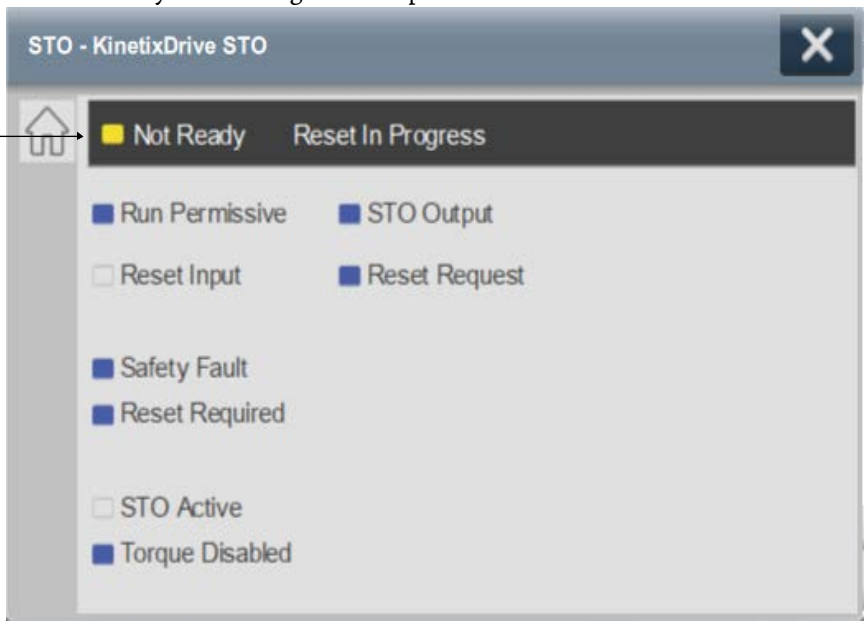
- **Waiting on Torque Off Status** faceplate displays *Waiting on Torque Off Status* on the fault banner when the *STO Active input is ON* and *Torque Disabled input is OFF*. This condition may observed for very small amount of time when the STO Active input is ON and waiting for Torque disable input to gets ON.

9

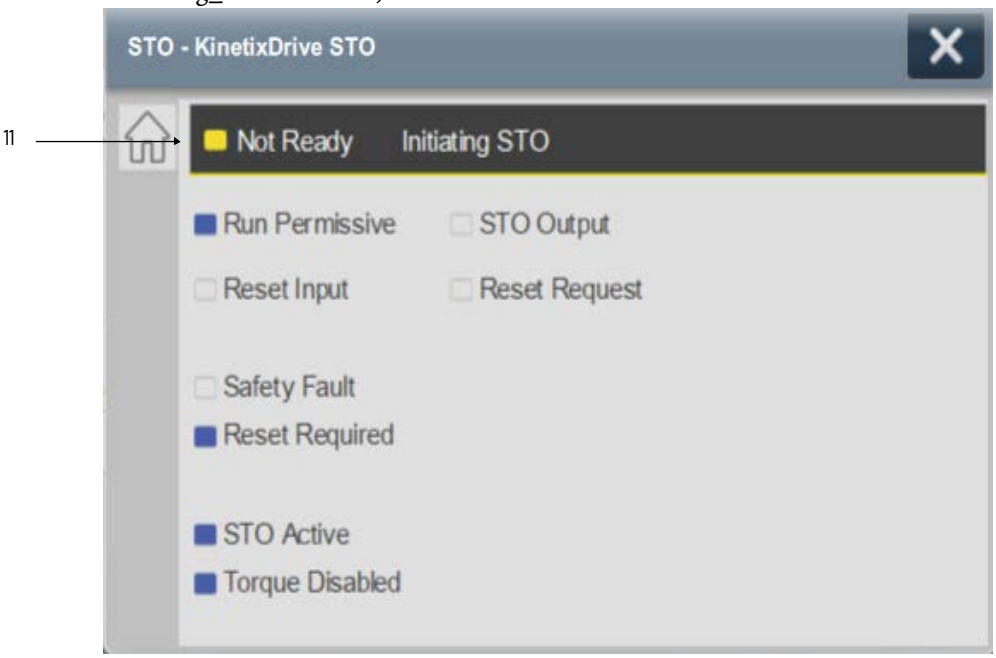


- **Reset In Progress** faceplate displays *Reset In Progress* on the fault banner when continuous safety fault is present on the hardware and resetting the safety fault using Reset Input.

10



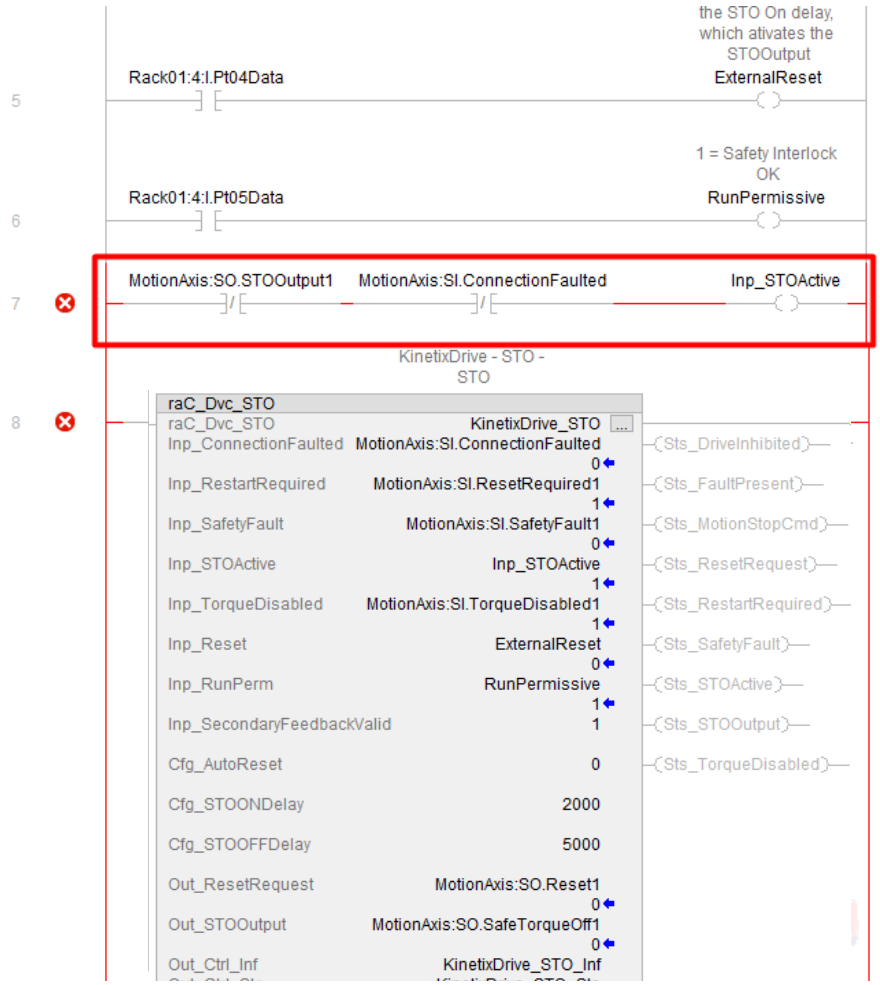
- **Initiating STO** faceplate displays *Initiating STO* on the fault banner on every Reset Required condition when "Cfg_AutoReset" is ON (i.e. Cfg_AutoReset =1)



Item	Description
3	Fault Present - Safety Fault - Drive Inhibited
4	Fault Present - Connection Faulted - Drive Inhibited
5	Fault Present - STO Delay Config. Error - Drive Inhibited
6	External STO RunPerm is OFF - Drive Inhibited
7	Reset Required - Drive Inhibited
8	STO is OFF Scan - Drive Inhibited
9	Waiting on Torque Off Status
10	Reset In Progress
11	Initiating STO

Note: - The STO_ Active Input may not be available with each Hardware (e.g.In PF527, module defined data-type input "_ModuleName:SI.STOActive" is not available). In that case, user needs to insert one rung between rung 6 & 7 to Create & configure "STOActive Input", in a routine where the STO object is instantiated.

Refer below image for How to create & configure "STOActive" Input, if it is not available with the connected hardware.



Application Code Manager

Safety Device Library objects can be set-up and configured using Studio 5000® Application Code Manager.

Refer to the section Using Application Code Manager for complete details.

Definition Object: raC_Dvc_STO

This object contains the AOI definition and used as linked library to implement object. This gives flexibility to choose to instantiate only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Implementation Object: raC_LD_Dvc_STO

Parameter Name	Default Value	Instance Name	Definition	Description
RoutineName	{ObjectName}	{RoutineName}	Routine	Enter Routine name. Routine will be created and Object implement rung(s) inserted. A JSR will be inserted in MainRoutine. If routine name already exists, then object will be inserted into existing routine. By default, parameter is set to Object Name.
TagName	{ObjectName}	{TagName}	Backing Tag	Enter the backing tag of the main AOI. This will serve as the base tag name for other tags in this object that are derived from the base.
TagDescription	{ObjectDescription}	{TagDescription}		Tag Description of the main AOI backing tag
DisplayName	[ObjectName]	[DisplayName]		Faceplate title/group event list description. Please avoid underscore ("_") at the end.
AxisModuleName	AxisModuleName	{AxisModuleName}		This Parameter is used to assign the Name of STO Axis Module.
NumberOfAxesPerModule	1	N A	Device Selection	This Parameter is used to select the device is Single Axis or Dual Axis. 1 = Single Axis, 2 = Dual Axis.
Cfg_STOONDelay	10	{Cfg_STOONDelay}		This Parameter is used to configure the STO Output ON Delay.
Cfg_STOOFFDelay	1000	{Cfg_STOOFFDelay}		This Parameter is used to configure the STO Output OFF Delay.
Cfg_AutoReset	False	{Cfg_AutoReset}		This Parameter is used to automatically Reset the error and STO Object.
RunPermissive	RunPermissiveSignal	{RunPermissiveSignal}		Enter the input address (data) tag for Runperm from safety digital input module. Note State 1 = Run permissive is ON.
CmdReset	CmdResetSignal	{CmdResetSignal}		Enter the input address (data) tag for CmdReset from safety digital input module. Note State 1 = Cmd Reset is ON.

Note that if the tag names are manually entered or not linked to input channel tags a red 'X' will be shown beside the parameter. This is acceptable and the program can still be generated.

Linked Libraries

Link Name	Catalog Number	Revision	Solution	Category
raC_Dvc_STO	raC_Dvc_STO	1	(RA-LIB) Device	Safety

Configured HMI Content

HMI Content	Instance Name	Description
Launch Button	{ObjectName}_GO_LaunchFP	Global Object configured callout instance

Attachments

Name	Description	File Name	Extraction Path
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols SE	(raC-1-SE) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewSE\Global Objects
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols ME	(raC-1-ME) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewME\Global Objects
V1_raC_Dvc_STO	Faceplate SE	(raC-1_xx-SE) raC_Dvc_STO-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
V1_raC_Dvc_STO	Faceplate ME	(raC-1_xx-ME) raC_Dvc_STO-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
V1_raC_Dvc_STO	View Designer	(raC-1_xx-VD) raC_Dvc_Safety.vpd	{ProjectName}\Visualization\ViewDesigner
V1_RM_raC_DvcSafety	Reference Manual	DEVICE-RM500C-EN-P.pdf	{ProjectName}\Documentation
V1_Safety_Images	HMI Image Set	Safety_Images.zip	{ProjectName}\Visualization\Images

Safety Group Monitor and Safety Group Members (raC_Opr_SafetyGroupMonitor, raC_Opr_SafetyGroupMember)

Overview

The Safety Group Monitor and Safety Group Members device objects (raC_Opr_SafetyGroupMonitor and raC_Opr_SafetyGroupMember) allows grouping of up to 32 safety devices for convenient monitoring. The Safety Group Monitor faceplate provides a summary status of all the safety instructions that have been assigned to a specific group. The groups may be organized for your specific safety system needs. A group may be made up of safety input instructions, output instructions, other safety monitor groups, or a combination of any of these.

In the Library there is a folder named *Videos* which contains many How-To and Operational Overview Videos which walk step-by-step through each process. You can refer to the following videos for this section:
"Operational_Overview_of_Safety_Device_Object_Faceplates"
"Operational_Overview_of_Safety_Application_Example_in_LogixDesigner"

Functional Description

The Safety Group Monitor and Safety Group Members pre-configured Device Objects:

- Collect, Process and Deliver Data between Smart Devices and Application Logic
- Provide Device Status & Diagnostics Faceplates for Machine Startup, Commissioning, Operations, and Maintenance
- Include Common HMI Faceplate Components, Device States, and Control Interfaces providing Application Development and Operation Efficiency

All these features provide quick feedback, shorten recovery time, and simplify implementation.

Required Files

Device Objects include Add-On Instructions (AOIs) and HMI faceplates. The revision number (e.g. 1.01) used in filenames can change as new revisions are created.

Controller Files

Add-On Instructions are reusable code objects that contain encapsulated logic that can streamline implementing your system. This lets you create your own instruction set for programming logic as a supplement to the instruction set provided natively in the ControlLogix® firmware. An Add-On Instruction is

defined once in each controller project, and can be instantiated multiple times in your application code as needed.

The Add-On Instruction must be imported into the controller project to be used in the controller configuration. These can be imported as Add-On Instruction files, or as part of the Rung Import or Import Library Objects wizard.

All Add-On Instruction and Rung Import files can be found in the */Studio 5000 Logix Designer® Files - L5X/* folder in the library.

Device/Item	Add-On Instruction	Rung Import
SafetyGroupMonitor	raC_Opr_SafetyGroupMonitor_1.01_AOI.L5X	raC_Opr_SafetyGroupMonitor_1.01_RUNG.L5X
SafetyGroupMember	raC_Opr_SafetyGroupMember_1.01_AOI.L5X	raC_Opr_SafetyGroupMember_1.01_RUNG.L5X

FactoryTalk® View HMI Files

FactoryTalk® View ME or SE applications require importing the desired device faceplates in addition to all Global Object (ggfx) files and all images located in the */HMI FactoryTalk View Images - png/* folder of the library. FactoryTalk® View ME files are stored in the */HMI - FactoryTalk View ME/* library folder and FactoryTalk® View SE files are stored in the */HMI - FactoryTalk View SE/* library folder.

Device/Item	Type	FactoryTalk® View ME Faceplate	FactoryTalk® View SE Faceplate
SafetyGroupMonitor	Display	(raC-1_01-ME) raC_Opr_SafetyGroupMonitor-Faceplate.gfx	(raC-1_01-SE) raC_Opr_SafetyGroupMonitor-Faceplate.gfx
Graphic Symbols	Global Object	(raC-1-ME) Graphic Symbols - Safety Device	(raC-1-SE) Graphic Symbols - Safety Device.ggfx
Toolbox	Global Object	(raC-1-ME) Toolbox - Safety Device.ggfx	(raC-1-SE) Toolbox - Safety Device.ggfx

Studio 5000 View Designer® HMI Files

All Studio 5000 View Designer® Files can be found in the */HMI - ViewDesigner - vpd/* folder of the library.

Device/Item	Studio 5000 View Designer® Faceplate
SafetyGroupMonitor	(raC-1_02-VD) raC_Opr_Safety.vpd

Studio 5000® Application Code Manager Files

Studio 5000® Application Code Manager (ACM) can be optionally used if it is installed. All devices can be easily registered in the ACM repositories by running the *setup.cmd* file located in the root folder of the library.

Individual HSL4 files are provided as an alternative to running the setup.cmd to allow users to manually register specific implementation objects. Each object has two files - an Asset Control file and a Device file. The Asset Control files include attachments of all required files for that object. The Device files are used to actually add that device into a Studio 5000 project and these reference the Asset Control files.

All Studio 5000® Application Code Manager files can be found in the /*ApplicationCodeManagerLibraries*/ folder of the library. The files included are as follows:

Implementation Object	Asset Control File (.HSL4)	Device File (.HSL4)
SafetyGroupMonitor	(RA-LIB)_Device_Asset-Control_Safety_raC_Opr_SafetyGroupMonitor_(1.2)	(RA-LIB)_Device_Device_Safety_raC_LD_SafetyGroupMonitor_(1.2)
SafetyGroupMember	(RA-LIB)_Device_Asset-Control_Safety_raC_Opr_SafetyGroupMember_(1.2)	

Operations

The Safety Device objects provides only physical operation mode. There is no virtual device mode offered.

Execution

The following table explains the handling of instruction execution conditions.

Condition	Description
EnableIn False (false rung)	Processing for EnableIn False (false rung) is handled the same as if the device were taken out of service by Command. The device outputs are de-energized and the device is shown as Program Out of Service on the HMI. All alarms are cleared.
Powerup (prescan, first scan)	On prescan, any commands that are received before first scan are discarded. The device is de-energized. On first scan, the device is treated as if it were returning from Hand command source: the instruction state is set based on the position feedback that is received from the device. If the feedback is valid for one position, the device is set to that position. If the device does not have position feedback or the position feedback is invalid, the device is set to the 'unknown/powerup' state. The command source is set to its default, either Operator or Program (unlocked).
Postscan	No SFC Postscan logic is provided.

Add-On Instruction I/O Data

raC_Opr_SafetyGroupMonitor - InOut Data

InOut	Function / Description	DataType
Ref_Ctrl_SafetyGroup	Input Interface - SafetyGroupMonitor Instruction	raC_UDT_Itf_SafetyGroup
Out_Ctrl_Inf	Information Interface	raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	Status Interface	raC_UDT_Itf_SafetyDevice_Sts

raC_Opr_SafetyGroupMonitor - Input Data

Input	Function/Description	DataType
EnableIn	Enable Input - System Defined Parameter	BOOL
Cmd_ClearList	Clear list	BOOL

raC_Opr_SafetyGroupMonitor - Output Data

Output	Function/Descritpion	DataType
Sts_bDeviceSuspended	Device is suspended - bitwise per group member	DINT
Sts_bSafetyDemand	Safety demand is present - bitwise per group member	DINT
Sts_bResetRequired	Reset is required - bitwise per group member	DINT
Sts_bDiagnosticPresent	Diagnostic is present - bitwise per group member	DINT
Sts_bFaultPresent	Fault is present - bitwise per group member	DINT
Sts_bMuted	Muting is active - bitwise per group member	DINT
Sts_DeviceSuspended	Device is suspended	BOOL
Sts_SafetyDemand	Safety demand is present	BOOL
Sts_ResetRequired	Reset is required	BOOL
Sts_DiagnosticPresent	Diagnostic is present	BOOL
Sts_FaultPresent	Fault is present	BOOL
Sts_Muted	Muted status	BOOL
Sts_DuplicateDevice	There is a duplicate device associated with the safet group	BOOL
Sts_eDuplicateDevice	Enumerated duplicate device status	DINT
Sts_ERR	Instruction Error Code	DINT
raC_Opr_SafetyGroupMonitor	Unique Parameter Name for auto - discovery	BOOL
EnableOut	Enable Output - System Defined Parameter	BOOL

raC_Opr_SafetyGroupMember - InOut Data

InOut	Function / Description	DataType
Ref_Ctrl_SafetyGroup	Input Interface - SafetyGroupMonitor Instruction	raC_UDT_Itf_SafetyGroup
Out_Ctrl_Inf	Information Interface	raC_UDT_Itf_SafetyDevice_Inf
Out_Ctrl_Sts	Status Interface	raC_UDT_Itf_SafetyDevice_Sts

raC_Opr_SafetyGroupMember - Input Data

Input	Function/Description	DataType
EnableIn	Enable Input - System Defined Parameter	BOOL
Set_ID	Set group member ID #.	DINT

raC_Opr_SafetyGroupMember- Output Data

Output	Function/Descritpion	DataType
Sts_DeviceSuspended	Device is suspended	BOOL
Sts_SafetyDemand	Safety demand is present	BOOL
Sts_ResetRequired	Reset is required	BOOL
Sts_DiagnosticPresent	Diagnostic is present	BOOL
Sts_FaultPresent	Fault is present	BOOL
Sts_ERR	Instruction Error Code	DINT
raC_Opr_SafetyGroupMember	Unique Parameter Name for auto - discovery	BOOL
EnableOut	Enable Output - System Defined Parameter	BOOL

Data Types

The following Safety Common Control Interface tags are the primary device program tags to read and write to when interfacing to safety devices. The value of using these tags in your specific application code is that you may use a number of different safety devices such as DCS, DCSTL, etc without having to update your application device interface tags. The primary function of these interfaces within the Safety Device Library is to pass individual instruction information to the Safety Group Monitor instruction.

Refer to the Interfaces section for detailed information on interfaces.

raC_UDT_Itf_SafetyDevice_Inf

Member	Description	Data Type
Type	Provides device object type information e.g. raC_Opr_SafetyGroupMonitor, raC_Opr_SafetyGroupMonitor, raC_Opr_SafetyGroupMonitorTL, etc.	STR0032
DisplayName	Provides object display name	STR0024
eClass	Provide instruction type information eClass ID: Instruction Type 0: DCS 1: DCSTL 2: DCSTM 3: CROUT 5: SafetyGroupMonitor 6: TSAM 7: TSSM 8: FSBM 9: THRSe 10: SMAT 11:SS2 12: SS1 13:SFX 14:STO	DINT

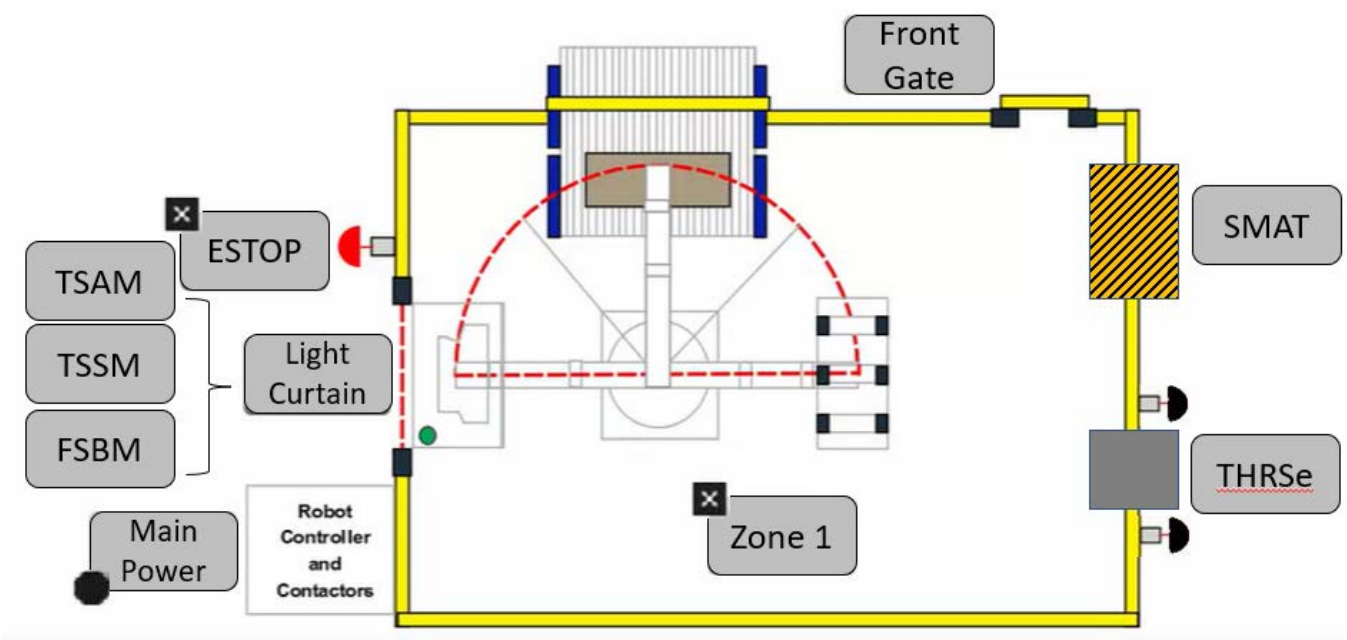
raC_UDT_Itf_SafetyDevice_Sts

This is the Safety Device Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data. The table below shows member names, descriptions, and tag data types.

Input	Description	Data Type
bSts	Bit overlay (Visible) covering all subsequent Boolean members. (Unused)	DINT
DeviceSuspended	Device suspended status (Inp_Suspend = 1)	BOOL
SafetyDemand	Safety demand status ON (1): Safety demand is present on the device object OFF (0): No safety demand	BOOL
ResetRequired	Reset Required status ON (1): Device requires reset signal OFF (0): No reset required	BOOL
DiagnosticPresent	Diagnostic status ON (1): A diagnostic is present in the device object OFF (0): Device is operating normally	BOOL
FaultPresent	Fault Status ON (1): A fault is present in the device object OFF (0): Device is operating normally	BOOL
Muted	Device Muted status (Available for SafetyGroupMonitorTM) ON (1): Muted status is present on one or more safety instructions within a specific safety group OFF (0): None of the instructions within group is muted	BOOL

Programming Example

Fully configured device routine is provided below for reference. This example is taken from the *Safety Program - Monitor* routine in the *SafetytApplication.ACD* example application provided in the library.



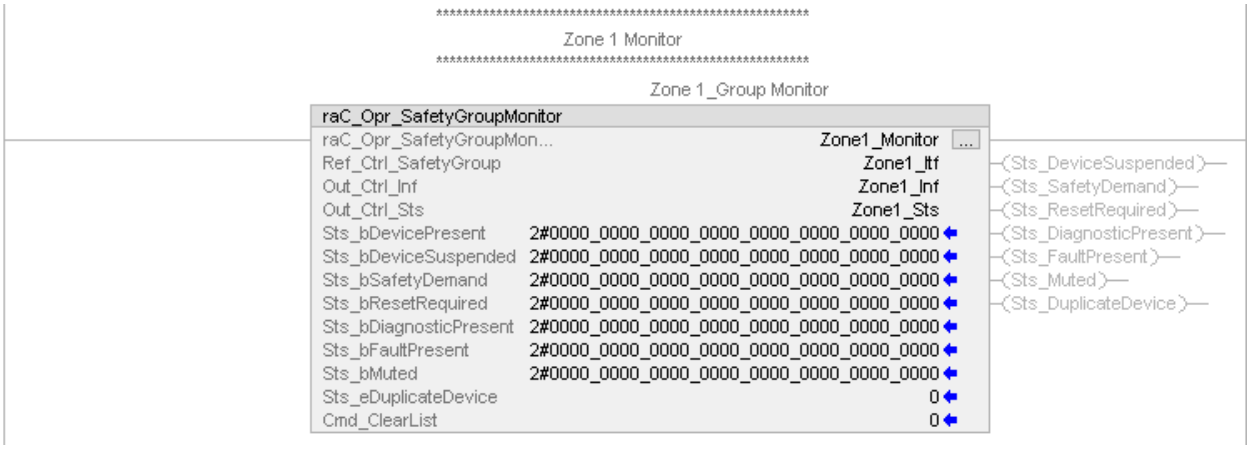
This example uses the SafetyGroupMonitor instruction to first group all inputs (DCS, DCSTL, DCSTM, SMAT, THRSe, TSSM, TSAM) and all outputs (CROUT) into groups Inputs and Outputs respectively. Then both the Inputs and Outputs groups are combined into a Zone1 group shown as *Zone 1* in the safe machine diagram. Note that the example code for each individual SafetyGroupMember is shown in the sections of this manual specific to each safety instruction.

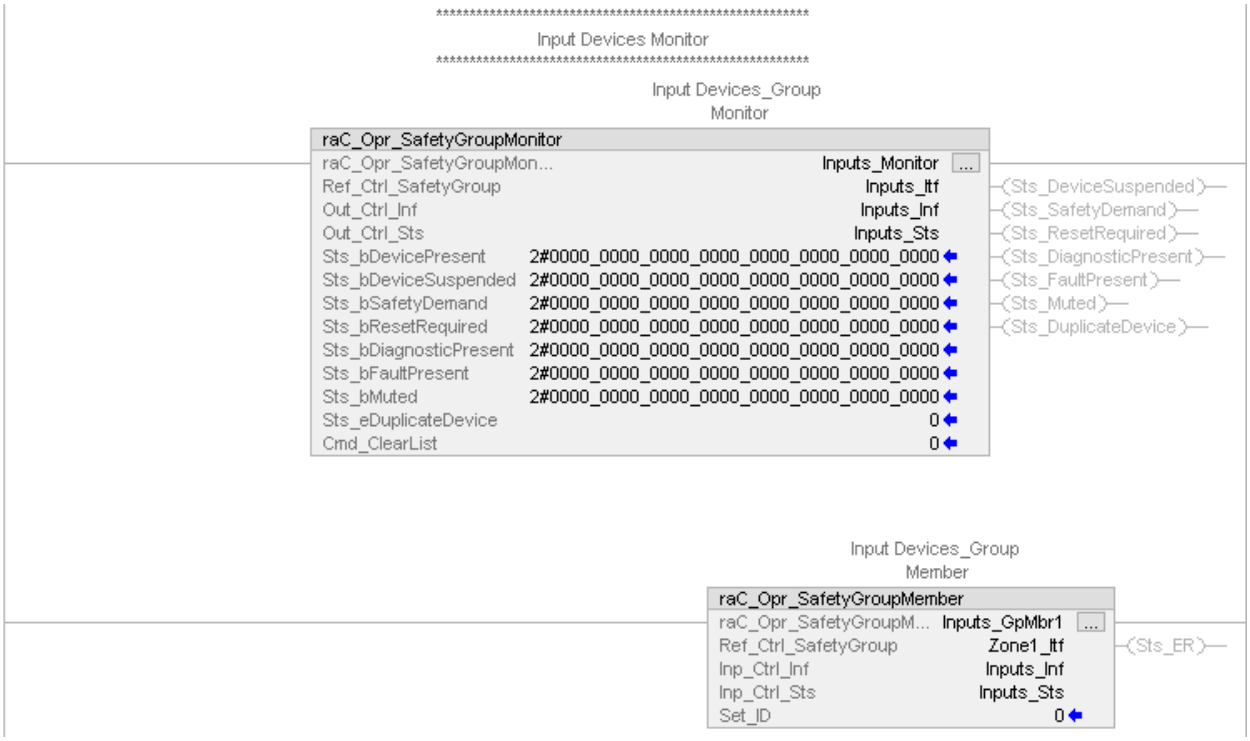
Safety Group	ID#	Members	Member Type
Inputs	0	MainEstop	DCS
	1	LightCurtain	DCSTM
	2	FrontGate	DCSTL
	4	Conveyor_101	TSAM
	5	Conveyor_102	TSSM
	6	Conveyor_103	FSBM
	7	Press_Machine	THRSe
	8	Robot_Area	SMAT
Outputs	0	MainPower	CROUT
Zone1	0	Inputs group	Safety Group
	1	Outputs group	Safety Group

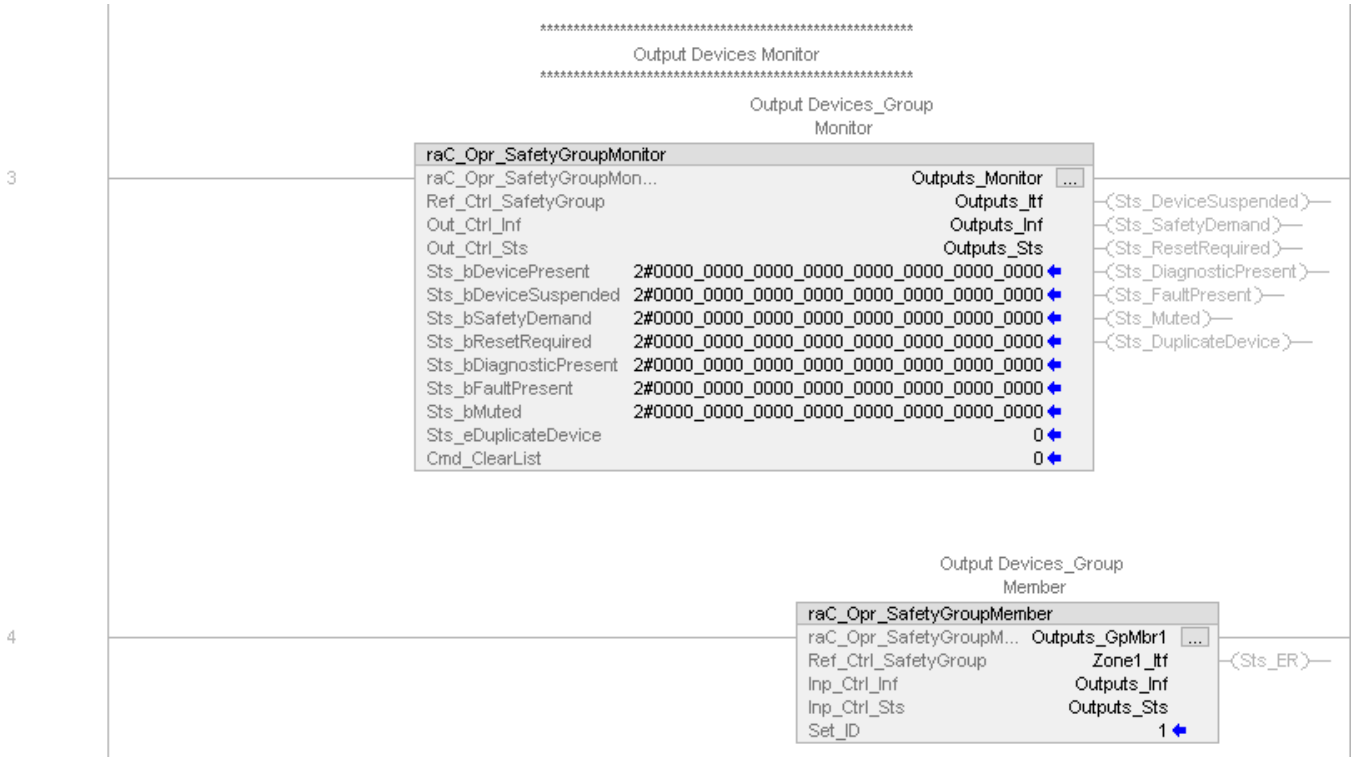
Rung descriptions are as follows:

- Rung 0: Assign Zone1 Safety Group Monitor
- Rung 1: Assign Inputs Safety Group Monitor
- Rung 2: raC_Opr_SafetyGroupMember assigns Inputs SafetyGroupMonitor to Zone1 Safety Group at ID #0.

- Rung 3: Assign Outputs Safety Group Monitor
- Rung 4: raC_Opr_SafetyGroupMember assigns Outputs SafetyGroupMonitor to Zone1 Safety Group at ID #1.







Note that this programming example is the same code that is imported when either importing the supplied rung .L5X files or when using Application Code Manager or the Studio 5000 Import Library Objects wizard plug-in.

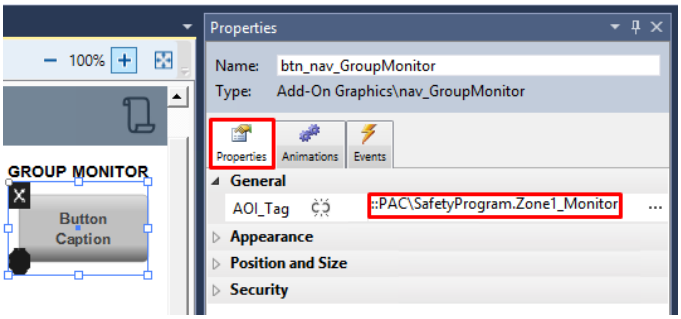
Graphic Symbols

Graphic Symbols are used as launch buttons within HMI applications to open up faceplate displays. See Basic Launch Button Attributes section for details on configuration and indicators

FactoryTalk® View ME/SE Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Global Object Parameter Values
btn_nav_safety		Faceplate navigation button with string tag label. This launch button graphic object allows the user to navigate to the device object faceplate. The text on the button face is set to the tag's .Inf_DisplayName.	#101: AOI Instance (e.g. {::[PAC]Program::SafetyProgram._InstanceName})

Studio 5000 View Designer® Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Property Configuration
Launch		The supplied launch button in View Designer is used to navigate to the faceplate in a user application.	

Faceplates

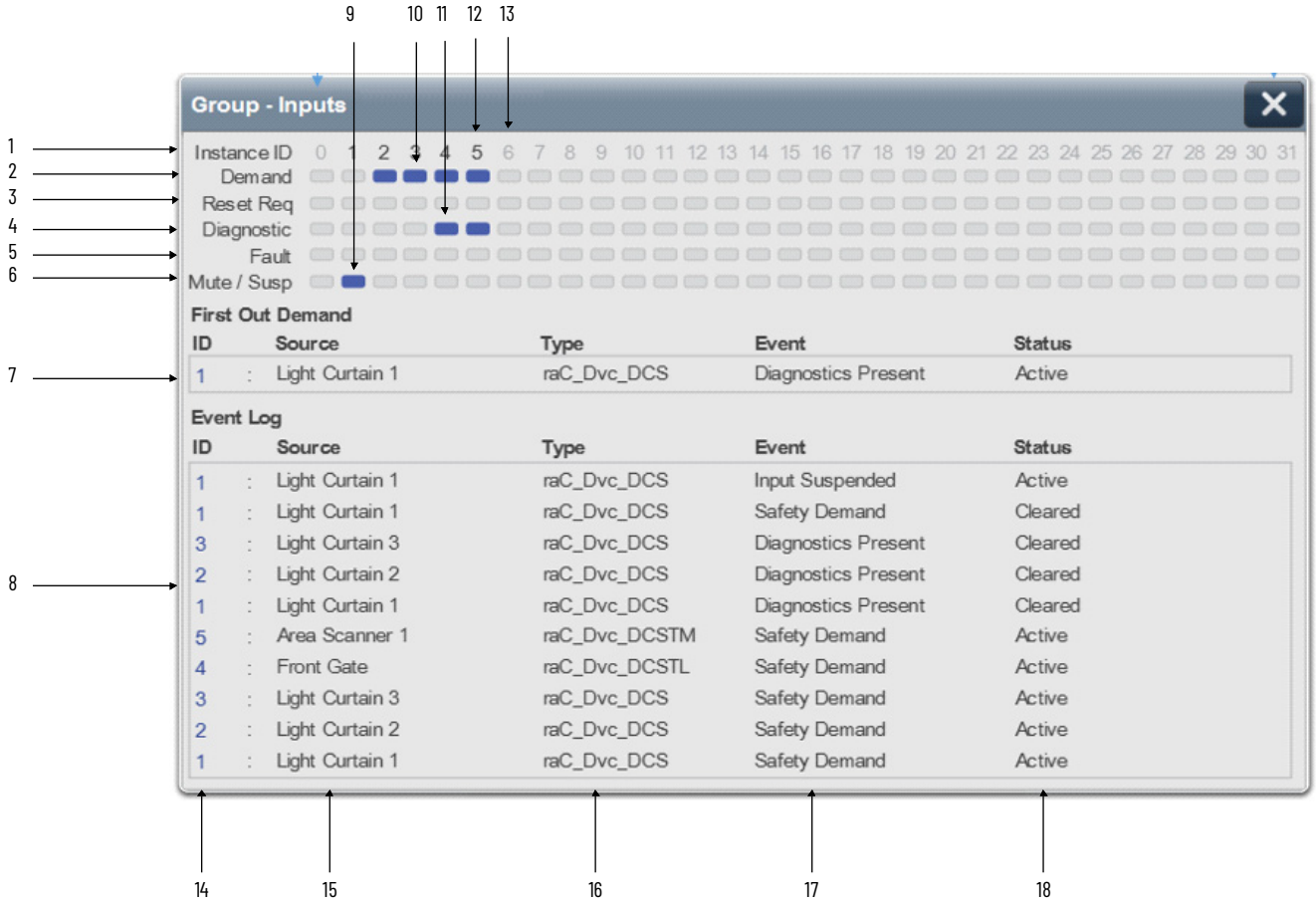
There are basic faceplate attributes that are common across all instructions. See Basic Faceplate Attributes on page 27.

Safety Group Monitor

Safety Group Monitor faceplate consist of a single screen which displays summary status of safety devices, it also display group name as faceplate title on top left corner. Faceplate title and launch button caption is similar and is linked to "Inf_DisplayName" tag coming from AOI instance and this AOI tag is user configurable.

The Safety Group Monitor faceplates display the ID number and individual indicators providing the demand, reset required, diagnostic, fault, and mute/suspend status of each safety device configured as the member of the associated Safety Group.

The Safety Group Monitor faceplates also display the 10 most recent member status events and the First Out Demand event to assist in troubleshooting the root cause of a number of safety demand conditions.



Item	Description
1	Status Summary - Instance ID Unique device IDs shown from 0 to 31 as assigned to the safety group members. Used/assigned ID numbers are shown in black text; unused/unassigned ID numbers are shown in gray text.
2	Status Summary - Safety Demand Present 32 binary indicators, each represents safety demand status of assigned device ID safety group member; Gray LED (OFF) = No safety demand. Blue LED (ON) = Safety demand present.
3	Status Summary - Reset Request 32 binary indicators, each represents reset required status of assigned device ID safety group member; Gray LED (OFF) = No reset required. Blue LED (ON) = reset required.
4	Status Summary - Diagnostic Present 32 binary indicators, each represents diagnostic status of assigned device ID safety group member; Gray LED (OFF) = no diagnostic present. Blue LED (ON) = diagnostic present
5	Status Summary - Fault Present 32 binary indicators, each represents fault status of assigned device ID safety group member; Gray LED (OFF) = no fault present. Blue LED (ON) = fault present
6	Status Summary - Mute/Suspended Active 32 binary indicators, each represents mute or input suspended status of assigned device ID safety group member; Gray LED (OFF) = mute/suspended inactive. Blue LED (ON) = mute/suspended active
7	First Out Demand Event Displays the first event occurred on respective safety group when previously there was no events.
8	Event Log Displays 10 most recent events related to the safety group.
9	Mute/Suspended Active Status; OFF (0) = Gray LED, ON (1) = Blue LED
10	Demand Active Status; OFF (0) = Gray LED, ON (1) = Blue LED
11	Diagnostic Active Status; OFF (0) = Gray LED, ON (1) = Blue LED
12	Used/assigned ID number shown in black text
13	Unused/unassigned ID number shown in gray text

Item	Description
14	Event/First Out - ID Unique device IDs shown from 0 to 31 as assigned to the safety group member.
15	Event/First Out - Source Device name of the related safety group member
16	Event/First Out - Instruction Type Instruction type of the related safety group member.
17	Event/First Out - Event Type Event type of the related safety group member. e.g. Fault Present, Diagnostics Present, Safety Demand, Reset Required, Mute, Input Suspended
18	Event/First Out - Status Event status active or cleared. Active and cleared events are logged separately.

There are three main sections of the faceplate:

- Status summary
- First Out Demand
- Event Log

Status Summary

When Group Monitor faceplate launch button is pressed, it will launch Group Monitor faceplate. On top section of faceplate status summary of safety devices is shown.

Safety group can consist up to 32 members (safety instructions) which can be safety input instructions, output instructions, or a combination of both.

This section of faceplate displays status of all these devices which are the part of this group.

Each member of group is identified by unique member ID's from 0 to 31.

There are six rows in the status section:

- Instance ID
- Demand
- Reset Req
- Diagnostic
- Fault
- Mute/Susp

First Out Demand

The First Out Demand section displays the first event occurred on respective safety group when previously there was no events.

The first out demand row displays events when the Group Monitor instruction receives an event of type Fault, Diagnostic, Safety Demand or Reset Required when previously there was no status for these categories.

The event will continue to be listed until Fault, Diagnostic, Safety Demand or Reset Required all return to cleared status. Once all statuses are cleared, the First out event information will be removed.

The following first out demand information is displayed:

Item	Description
ID	ID column displays device ID on which first out demand is present
Source	Source column shows device name on which first out demand is detected
Type	Type column shows type of device on which first out demand is detected
Event	Event column shows event name present of device on which first out demand is detected - Fault Present - Diagnostics Present - Safety Demand - Reset Required - Mute - Input Suspended
Status	Status column shows event status as Active or cleared

Event Log

Event log section displays the historical event history which is occurred on defined safety group. It will display last 10 event occurred on respective safety group.

If there is more than 10 events present then it will discard the events based on first in first out priority and displays the most recent 10 events.

The following event information is displayed:

Item	Description
ID	ID column displays the device ID on which corresponding event is triggered
Source	Source column shows device name on which corresponding event is triggered
Type	Type column shows type of device on which event is triggered
Event	Event column shows exact event name that was triggered or present on corresponding device Events are: - Fault Present - Diagnostics Present - Safety Demand - Reset Required - Mute - Input Suspended
Status	Status column shows event status as Active or cleared, As soon as event is triggered on a device it will be logged as an active event and once it is cleared it will be logged as Cleared event.

Application Code Manager

Safety Device Library objects can be set-up and configured using Studio 5000® Application Code Manager.

Refer to the section Using Application Code Manager for complete details on how to create Safety Groups and assign Safety Group Members.

Definition Object: raC_Opr_SafetyGroupMember

This object contains the Safety Group Member AOI definition and used as linked library to implement object. This gives flexibility to choose to instantiate

only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Definition Object: raC_Opr_SafetyGroupMonitor

This object contains the Safety Group Monitor AOI definition and used as linked library to implement object. This gives flexibility to choose to instantiate only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Implementation Object: raC_LD_Opr_SafetyGroupMonitor

Parameter Name	Default Value	Instance Name	Definition	Description
RoutineName	{ObjectName}	{RoutineName}	Routine	Enter Routine name. Routine will be created and Object implement rung(s) inserted. A JSR will be inserted in MainRoutine. If routine name already exists, then object will be inserted into existing routine. By default, parameter is set to Object Name.
TagName	{ObjectName}	{TagName}	Backing Tag	Enter the backing tag of the main AOI. This will serve as the base tag name for other tags in this object that are derived from the base.
TagDescription	{ObjectDescription}	{TagDescription}		Tag Description of the main AOI backing tag
DisplayName	[ObjectName]	[DisplayName]		Faceplate title/group event list description. Please avoid underscore ("_") at the end.

Linked Libraries

Link Name	Catalog Number	Revision	Solution	Category
raC_Opr_SafetyGroupMonitor	raC_Opr_SafetyGroupMonitor	1	(RA-LIB) Device	Safety
raC_Opr_SafetyGroupMember	raC_Opr_SafetyGroupMember	1	(RA-LIB) Device	Safety

Configured HMI Content

HMI Content	Instance Name	Description
Launch Button	{ObjectName}_GO_LaunchFP	Global Object configured callout instance

Attachments

Name	Description	File Name	Extraction Path
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols SE	(raC-1-SE) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewSE\Global Objects
V1_raC_GuardLogix_SafetyDevice_Global	Graphic Symbols ME	(raC-1-ME) Graphic Symbols - Safety Device.ggfx	{ProjectName}\Visualization\FTViewME\Global Objects
V1_raC_Opr_SafetyGroupMonitor	Faceplate SE	(raC-1_xx-SE) raC_Opr_SafetyGroupMonitor-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
V1_raC_Opr_SafetyGroupMonitor	Faceplate ME	(raC-1_xx-ME) raC_Opr_SafetyGroupMonitor-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
V1_raC_Opr_SafetyGroupMonitor	View Designer	(raC-1_xx-VD) raC_Dvc_Safety.vpd	{ProjectName}\Visualization\ViewDesigner
V1_RM_raC_OprSafety	Reference Manual	DEVICE-RM500C-EN-P.pdf	{ProjectName}\Documentation
V1_Safety_Images	HMI Image Set	Safety_Images.zip	{ProjectName}\Visualization\Images

Rockwell Automation Support

Use these resources to access support information.

Technical Support Center	Find help with how-to videos, FAQs, chat, user forums, and product notification updates.	rok.auto/support
Knowledgebase	Access Knowledgebase articles.	rok.auto/knowledgebase
Local Technical Support Phone Numbers	Locate the telephone number for your country.	rok.auto/phonesupport
Literature Library	Find installation instructions, manuals, brochures, and technical data publications.	rok.auto/literature
Product Compatibility and Download Center (PCDC)	Get help determining how products interact, check features and capabilities, and find associated firmware.	rok.auto/pcdc

Documentation Feedback

Your comments help us serve your documentation needs better. If you have any suggestions on how to improve our content, complete the form at rok.auto/docfeedback.

Waste Electrical and Electronic Equipment (WEEE)

At the end of life, this equipment should be collected separately from any unsorted municipal waste.

Rockwell Automation maintains current product environmental information on its website at rok.auto/pec.

Rockwell Otomasyon Ticaret A.Ş. Kar Plaza İş Merkezi E Blok Kat:6 34752, İçerenköy, İstanbul, Tel: +90 (216) 5698400 EEE YÖnetmeliğine Uygundur

Connect with us.

rockwellautomation.com — expanding **human possibility**™

AMERICAS: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
EUROPE/MIDDLE EAST/AFRICA: Rockwell Automation NV, Pegasus Park, De Kleetlaan 12a, 1831 Diegem, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
ASIA PACIFIC: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publication DEVICE-RM500C-EN-P - March 2024
Supersedes Publication DEVICE-RM500B-EN-P - July 2023