



Mettler Toledo Device Library

Release v1.00



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.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc., is prohibited.

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.

	Table of Contents	3
	Preface	7
	Device Object Libraries Overview	7
	Application Code Manager	7
	Other Application Code Libraries	8
	Software and Firmware Upgrades	9
	Rockwell Automation® Services and Support	9
	Chapter 1	
Rockwell Automation® Mettler Toledo Device Library	Compatibility	11
	Compatible Software	11
	Compatible Hardware	11
	Summary of Changes	12
	Footprint	12
	Additional Resources	12
	Chapter 2	
Library Components	Mettler Toledo Device Instructions	13
	Library Folders and Files	13
	Visualization Files	15
	Studio 5000 View Designer® Visualization Files	15
	Basic Faceplate Attributes	16
	Common Status Banner	16
	Faceplate Navigation	17
	Faceplate Revision Notes	17
	Launch Buttons	17
	Library Versions	19
	State Model	20
	Interfaces	20
	Application Code Manager	21
	Architectural Overview	21
	Chapter 3	
Using the Library	Install the Library	23
	Download the Library	23
	Download & Install Studio 5000® Application Code Manager	23
	Register Libraries in Studio 5000® Application Code Manager ...	24
	Importing Logic into Studio 5000® Projects	26
	Import Library Objects Wizard	26
	Import Rung Logic	28
	Using Studio 5000 View Designer®	31
	Using View Designer Project Files	31
	Configuring View Designer Objects	32
	Using FactoryTalk® View Studio	32

Import FactoryTalk® View Visualization Files	32
Configuring FactoryTalk® View Objects.....	33
Library Upgrades	34
Add-On Instruction Upgrades	34
FactoryTalk® View Upgrades.....	40
Studio 5000 View Designer® Upgrades	40

Chapter 4

Using Application Code Manager

Overview of Application Code Manager	41
Creating a New Project	41
Adding & Configuring Device Objects.....	44
Adding Mettler Toledo Device Objects	44
Configuring Displays.....	47
Generating Displays.....	49
Importing Displays into FactoryTalk® View Studio	49
Generating Controller Files	50
Exporting Attachments.....	51

Chapter 5

Using the Mettler Toledo Device Library with Other Application Code Libraries

Application Code Libraries.....	53
Using Mettler Toledo Device Objects with PlantPax® Process Objects Library.....	53
Using Mettler Toledo Device Objects with Machine Builder Library ..	53

Chapter 6

Mettler Toledo IND360 (raC_Dvc_MT_IND360)

Overview.....	55
Functional Description	55
Required Files	55
Controller Files	55
FactoryTalk® View HMI Files	56
Studio 5000 View Designer® HMI Files	56
Studio 5000® Application Code Manager Files	56
Device Definition	57
HARDWARE Definition.....	57
Operations	58
Faults & Warnings.....	58
Execution.....	59
Add-On Instruction I/O Data.....	59
Add-On Instruction Ladder Implementation	59
InOut Data.....	60
Input Data	60
Output Data.....	62
Data Types	65
raC_UDT_ItfAD_MettlerToledo_Set.....	65
raC_UDT_ItfAD_MettlerToledo_Cmd.....	66
raC_UDT_ItfAD_MettlerToledo_Sts	66
raC_UDT_Event	67
raC_UDT_LookupMember_STRoo82	67

Programming Example	68
Graphic Symbols	68
FactoryTalk® View ME/SE Graphic Symbols	68
Studio 5000 View Designer® Graphic Symbols	69
Faceplates	69
Home Tab	70
Trend Tab	72
Configure Tab	72
Diagnostics Tab	74
Fault Warning Tab	75
Application Code Manager	76
Definition Object: raC_Dvc_MT_IND360	76
Implementation Object: raC_LD_Dvc_MT_IND360	77
Linked Libraries	77
Configured HMI Content	78
Attachments	78

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 life-cycle 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 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: <ul style="list-style-type: none"> • 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 PowerMonitor™ devices
Safety Device Library	Provides safety objects to interface with safety I/O
Condition Monitoring Device Library	Provides Dynamix™ -1444 module and machinery Condition Monitoring applications such as motors and pumps. This includes FactoryTalk View® SE HMI faceplates and Studio 5000® Application Code Manager implementations.
Electrical Protection Device Library	Provides a standard to represent protection devices within your electrical distribution system
Mettler Toledo Device Library	Provides Mettler Toledo Weighing Terminal Objects including pre-configured status and diagnostic faceplates

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 TechConnectSM 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® Mettler Toledo Device Library

The Mettler Toledo 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 Mettler Toledo® Weighing Terminal devices. The Mettler Toledo Device Objects may be used with Machine Builder, Process, and Packaged Libraries or as standalone components. Mettler Toledo Device Library add-on instructions objects collect, process, and deliver data between hardware devices and application logic.

The Mettler Toledo Device Library includes Add-On Instructions (AOIs) and HMI Faceplates for Weighing Terminal Devices.



The Mettler Toledo Device Library may be downloaded from the [Product Compatibility and Download Center](#). Search for Mettler Toledo Device Library.

Compatibility

Compatible Software

- Studio 5000 Logix Designer® v31.02 or later for PAC Application Development
- Studio 5000® Application Code Manager v4.01 and later for bulk code configuration
- Studio 5000 View Designer® v8.00 and later for PanelView™ 5000 Application Development
- FactoryTalk® View Studio v10 and later for PanelView™ Plus or FactoryTalk® View SE Application Development

Compatible Hardware

- PanelView™ 5500 with v8 or later firmware
- PanelView™ Plus with v10 or later firmware
- ControlLogix® 5570/5580 controller or CompactLogix™ 5370/5380 Controller with v3.01 or later firmware
- IND360 with v1.001 or later firmware

IMPORTANT FactoryTalk View HMI faceplates are not compatible with FactoryTalk® ViewPoint

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	all

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 (L85):** Estimated execution time / scan footprint evaluated in 1756-L85 PAC

Device Object Footprint

Device Object	Definition (kB)	Instance (kB)	Execution (µs)
raC_Dvc_MT_IND360	81.668	4.264	96

Additional Resources

These documents contain additional information concerning related products from Rockwell Automation.

Resource	Description
Rockwell Automation Library of Process Objects Reference Manual PROCES-RM200	Describes the Add-On Instructions, PlantPAx instructions, and associated faceplates that are available to develop applications.
Application Code Manager User Manual LOGIX-UM003	Studio 5000® Application Code Manager user manual.
EtherNet/IP Network Devices User Manual, publication ENET-UM006	Describes how to configure and use EtherNet/IP devices to communicate on the EtherNet/IP network.
Mettler Toledo IND360 User Manual IND360-WeighingIndicators	Provides installation instructions, wiring diagram, configuration, and specifications.

Library Components

The Mettler Toledo 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 Mettler Toledo® Weighing Terminal device. The Mettler Toledo Device Objects may be used with Machine Builder, Process, and Packaged Libraries or as standalone components. Mettler Toledo Device Library add-on instructions objects collect, process, and deliver data between hardware devices and application logic.

Mettler Toledo Device Instructions

The Mettler Toledo Device Library includes instructions to interface with Weighing Terminal Modules. This single instruction is used for different weighing technologies like Analog, POWERCELL or Precision type modules. Weighing Terminals will provide information such as weight, status and condition monitoring, to your automation system rapidly and reliably.

The instruction included is as follows:

- [IND360](#) Weighing Terminal

There is one type of instruction in this library:

- Device (Dvc): instruction used for devices (e.g. IND360® Module).

Mettler Toledo Device Instructions

Instruction	Version	Category	Instruction Description
raC_Dvc_MT_IND360	1.00	Weighing Terminal	Mettler Toledo IND360 Analog, POWERCELL, and Precision.

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. IND360). 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
	MettlerToledoApplication_ACM_v1.00.xlsx		XLSX	Application Code Manager Project
	MettlerToledoApplication_v1.00.ACD		ACD	Logix Designer Example Project
	MettlerToledoApplication_SE_v1.00.apa		APA	FT View SE Project Archive
	MettlerToledoApplication_ME_v1.00.apa		APA	FT View ME Project Archive
	MettlerToledoApplication_VD_1.00.vpd		VPD	View Designer Project File

ApplicationCodeManagerLibraries			Folder	Application Code Manager files
	Attachments (.HZ1 and .txt files)		Folder	ACM Object Attachments
	(RA-LIB)_Device_Asset-Control_GROUP_raC_Dvc_OBJECT_(X.Y).HSL4		HSL4	ACM Asset-Control Object
	(RA-LIB)_Device_Device_GROUP_raC_Dvc_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 - MettlerToledo.ggfx	GGFX	Graphic Symbol/Launch Button global objects
		(raC-X-ME) Toolbox - MettlerToledo.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 - MettlerToledo.ggfx	GGFX	Graphic Symbol/Launch Button global objects
		(raC-X-SE) Toolbox - MettlerToledo.ggfx	GGFX	Toolbox global objects
HMI - ViewDesigner - vpd			Folder	Studio 5000 View Designer® Files
	(raC-X_YY-VD) raC_Dvc_MettlerToledo.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-RM905A-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_AOI.L5X	L5X	Object AOI import
Videos			Folder	How-to and Operational Overview Videos
	How_To_Import_and_Configure_TYPE_Objects_in_FTViewME_SE.mp4		MP4	How-to Video
	How_To_Import_and_Configure_TYPE_Objects_in_LogixDesigner.mp4		MP4	How-to Video
	How_To_Configure_TYPE_Objects_in_ViewDesigner.mp4		MP4	How-to Video
	How_To_Import_and_Configure_TYPE_Objects_in_ACM.mp4		MP4	How-to Video
	Operational_Overview_of_OBJECT_Faceplate.mp4		MP4	Operational Overview video
MettlerToledoDeviceLibrary_ReleaseNotes_vX.YY.pdf			PDF	Release Notes
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 Mettler Toledo 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 an HMI projects for FactoryTalk® View SE Local Station, FactoryTalk® View SE Local Station, and Studio 5000 View Designer®.

Visualization Files

Each Add-On Instruction has associated visualization files that provide a common user interface. The Mettler Toledo 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 <i>HMI FactoryTalk® View Images - png</i> 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 - MettlerToledo Device.ggfx	(raC-1-ME) Graphic Symbols - MettlerToledo Device.ggfx	Graphic symbols or launch buttons used to open faceplate displays from other displays.
	(raC-1-SE) Toolbox - MettlerToledo Device.ggfx	(raC-1-ME) Toolbox - MettlerToledo Device.ggfx	Common objects used across multiple device faceplates.
Standard displays (.gfx)	(raC-1_00-SE) precedes name of the display.	(raC-1_00-ME) precedes name of the display.	e.g. (raC-1_00-SE) raC_Dvc_MT_IND360-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

Studio 5000 View Designer® project files are supplied that contain faceplates and launch buttons for the Mettler Toledo Device Library. The devices are distributed over multiple Studio 5000 View Designer® Project files. These files are found in the *HMI - ViewDesigner - vpd* folder. Inside of the VPD file you will find a the required display files 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_MT_IND360_FP

Basic Faceplate Attributes

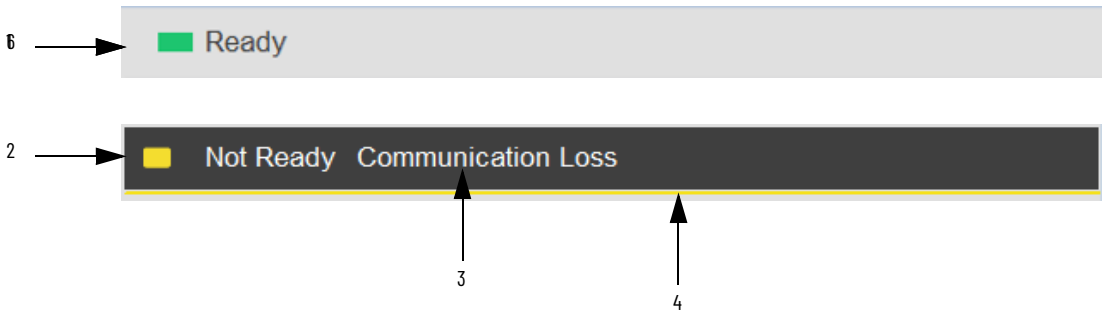
Faceplates consist of tabs, and each tab consists of one or more pages. The Home tab is displayed when the faceplate is initially opened. The faceplate provides the means for operators, maintenance personnel, engineers, and others to interact with a device or instruction instance, which includes a view of its status and values. Faceplates may also manipulate an instruction through its commands and settings. 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)
- Communication Loss

A fault status may be reset using the Reset button on the right side of the banner if the condition has been cleared.



Item	Description
1	Ready state displays green LED icon and grey background.
2	Faulted state shows yellow LED icon and Not Ready status.
3	Fault message for latest fault present. Will also display "Virtual" if virtual mode is enabled.
4	Faulted state shows yellow border around banner.

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.



Home Tab



Trend Tab



Configuration Tab



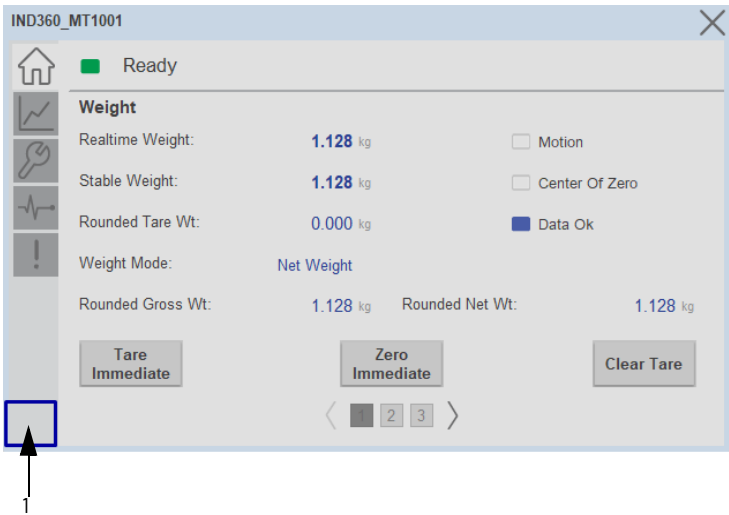
Diagnostics Tab



Fault Tab

Faceplate Revision Notes

By clicking on the open space near the bottom left corner of the faceplate you can momentarily view revision notes and details of the active faceplate. This may be useful in troubleshooting or when communicating with Rockwell Automation® Tech Support.



2
3


MT_IND360
Revision 1.00
(raC-1_00-ME)
raC_Dvc_MT_IND360-Faceplate
Copyright © Rockwell Automation, Inc. All Rights Reserved

Item	Description
1	Click near the bottom right corner to temporarily open up the revision notes dialogue
2	Revision number
3	Faceplate display name

Launch Buttons





Launch buttons are provided in Global Display (GGFX) files for FactoryTalk View® ME/SE as well as in Studio 5000 View Designer® projects. These are

used to open HMI faceplate displays or pop-ups. Two types of launch buttons are provided:

Launch Button Style	Image Examples	Usage
Basic Text Button	 A rectangular button with a light gray background and a thin black border. The text "S...S" is centered in the button. Four diagnostic icons are shown around the button: a square with an 'x' in the top-left corner, a circle with an 'x' in the bottom-left corner, a square with an 'x' in the top-right corner, and a solid black circle in the bottom-right corner.	Simple launch button with diagnostic information.

Diagnostic Icons

Diagnostic icons may be displayed on the graphic buttons for compatible modules. Safety modules are designated with a small guard icon.

Icon	Image	Visible Condition
Communications Failure		Connection Faulted
Fault		Any device fault active (module hardware issue)
Warning		Any device warning active (maintenance required)
Not Ready		Device Not Ready

Library Versions

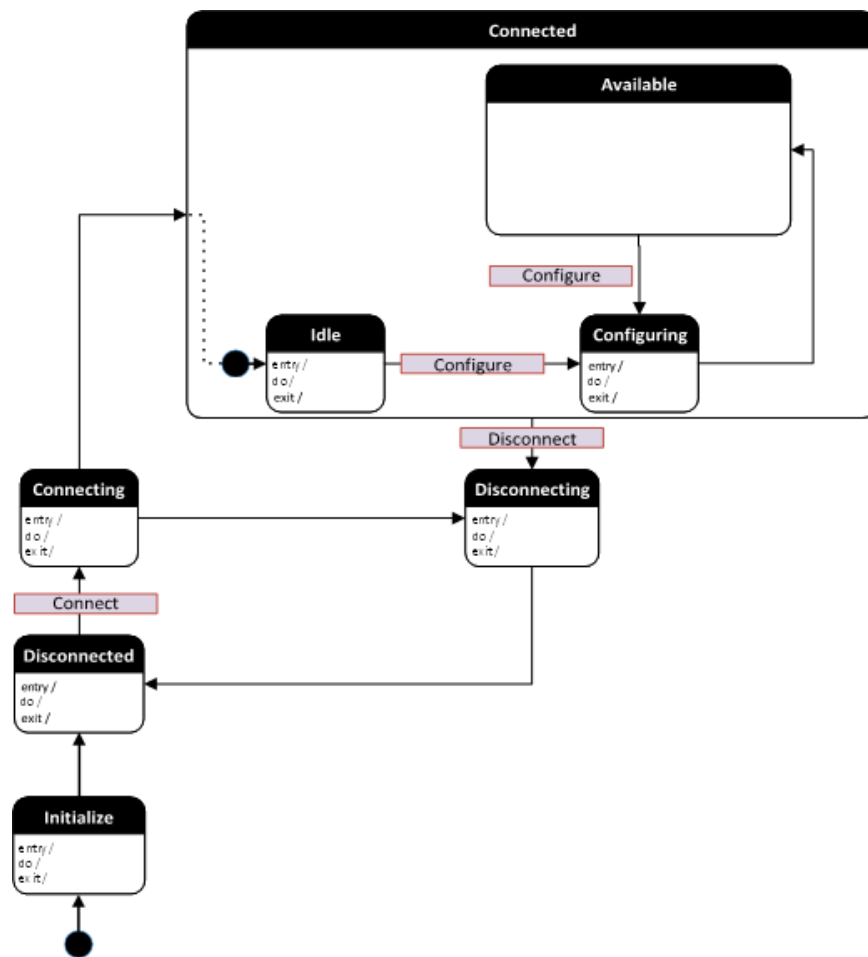
Each library object has a revision x.yy where: x is the Major Revision number and yy is the Minor Revision number. Each release of the library comes with release notes that describe the changes that were made since the last release. You can find the revision number of the object in a number of locations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

Component	Example																										
The Add-On Instruction in Logix Designer application has revision information visible when the instruction is selected in the Controller Organizer.	<table border="1"> <tr><td>Description</td><td>Mettler Toledo IND360 Analog, PowerCell and Precision</td></tr> <tr><td>Revision</td><td>v1.0 .00</td></tr> <tr><td>Revision Note</td><td>Released</td></tr> <tr><td>Vendor</td><td>Rockwell Automation</td></tr> <tr><td>Data Type Size</td><td>984 bytes</td></tr> <tr><td>Created</td><td>4/24/2017 7:11:25 PM</td></tr> <tr><td>Created By</td><td>Not Available</td></tr> <tr><td>Edited</td><td>7/27/2023 6:28:48 PM</td></tr> <tr><td>Edited By</td><td>Not Available</td></tr> <tr><td>Signature ID</td><td><none></td></tr> <tr><td>Protection Type</td><td>Source Key</td></tr> <tr><td>Protection Name</td><td>v1raC_Dvc_MettlerToledo*,*</td></tr> <tr><td>Protection Permissions</td><td>Protect, Edit, Copy, Export, +View, Use</td></tr> </table>	Description	Mettler Toledo IND360 Analog, PowerCell and Precision	Revision	v1.0 .00	Revision Note	Released	Vendor	Rockwell Automation	Data Type Size	984 bytes	Created	4/24/2017 7:11:25 PM	Created By	Not Available	Edited	7/27/2023 6:28:48 PM	Edited By	Not Available	Signature ID	<none>	Protection Type	Source Key	Protection Name	v1raC_Dvc_MettlerToledo*,*	Protection Permissions	Protect, Edit, Copy, Export, +View, Use
Description	Mettler Toledo IND360 Analog, PowerCell and Precision																										
Revision	v1.0 .00																										
Revision Note	Released																										
Vendor	Rockwell Automation																										
Data Type Size	984 bytes																										
Created	4/24/2017 7:11:25 PM																										
Created By	Not Available																										
Edited	7/27/2023 6:28:48 PM																										
Edited By	Not Available																										
Signature ID	<none>																										
Protection Type	Source Key																										
Protection Name	v1raC_Dvc_MettlerToledo*,*																										
Protection Permissions	Protect, Edit, Copy, Export, +View, Use																										
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 faceplate in FactoryTalk® View software has revision information visible when the pointer is clicked just inside the lower left corner of the faceplate.																											
The revision number is shown in the file names for GFX, VPD, ACM.HSL4, AOI.L5X, and RUNG.L5X files.	<ul style="list-style-type: none"> <input type="checkbox"/> (raC-1_00-ME) raC_Dvc_MT_IND360-Faceplate.gfx <input type="checkbox"/> (raC-1_00-SE) raC_Dvc_MT_IND360-Faceplate.gfx <input checked="" type="checkbox"/> (raC-1_00-VD) raC_Dvc_MettlerToledo <input type="checkbox"/> (RA-LIB)_Device_Asset-Control_MettlerToledo_raC_Dvc_MT_IND360_(1.0).HSL4 <input type="checkbox"/> (RA-LIB)_Device_Device_MettlerToledo_raC_LD_Dvc_MT_IND360_(1.0).HSL4 <input type="checkbox"/> raC_Dvc_MT_IND360_1.00_AOI.L5X <input type="checkbox"/> raC_Dvc_MT_IND360_1.00_RUNG.L5X 																										

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).

These interfaces are commonly used when interfacing with other Rockwell Automation® application code libraries such as the PlantPAx® Process Objects Library or the Machine Builder Library.

For detailed information on specific interfaces, please refer to the appropriate section in this manual. A list of interface UDTs used in this library follows. Note that OBJECT used in the Inp interfaces is replaced with the specific Mettler Toledo device object (e.g. IND360).

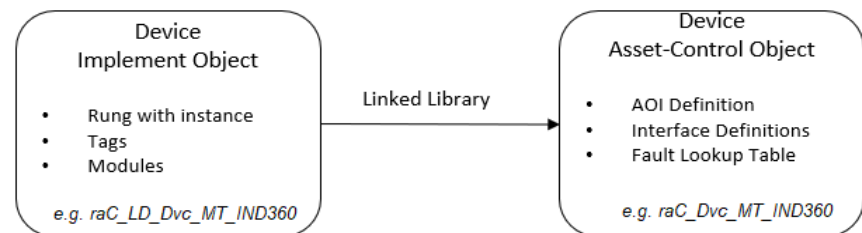
Interface Class	Object Class	Object Sub-Class	Interface Type	Interface Name (UDT)
Control	Mettler Toledo	MT_IND360	Command	raC_UDT_ItfAD_MettlerToledo_Cmd
			Setting	raC_UDT_ItfAD_MettlerToledo_Set
			Status	raC_UDT_ItfAD_MettlerToledo_Sts

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.



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.

FIND DOWNLOADS ?

Device Library	All Categories	All Families	Q	COMPARE	DOWNLOADS
IO Device Library	Tested, documented and life-cycle managed library objects for Rockwell Automation 1756, 1769, 1734, 1794, 1738, 1732E, 1719, 509 (Application Content/Engineering Libraries)			Power Device Library 3.01.00	↓
IO-Link Device Library	Tested, documented and life-cycle managed IO-Link Master and Sensor Library Objects. (Application Content/Engineering Libraries)			IO Device Library 5.00	↓
Network Device Library	Tested, documented and life-cycle managed library objects for Stratix Switch and Device Level Ring DLR network objects. (Application Content/Engineering Libraries)			IO-Link Device Library 2.2.00	↓
Power Device Library	Tested, documented and life-cycle managed Power Device Library Objects for E300, ArmorStart, SMC50, PowerFlex, and Kinetix. (Application Content/Engineering Libraries)			Network Device Library 1.04.00	↓
Safety Device Library	Tested, documented and life-cycle managed library objects for Rockwell Automation Safety Instructions. (Application Content/Engineering Libraries)			Safety Device Library 1.01.00	↓

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.

FIND DOWNLOADS ?

The screenshot shows the 'FIND DOWNLOADS' interface. On the left, a search bar contains 'Application Code Manager'. Below it, a list of results is shown under the heading '(Application Content/Engineering Libraries)'. The results include 'Independent Cart Technology Libraries', 'Machine Builder Libraries', 'Process Library', and 'Studio 5000 Application Code Manager'. The 'Studio 5000 Application Code Manager' entry is highlighted with a red box. Below the list, it says '5 items found' and a 'MOVE SELECTIONS' button. On the right, a sidebar shows the selected item 'Studio 5000 Application Code Manager' with version '4.02.00'. A download icon is highlighted with a red box. Below the sidebar, it says '1 selection' and buttons for 'COMPARE' and 'DOWNLOADS'.

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.



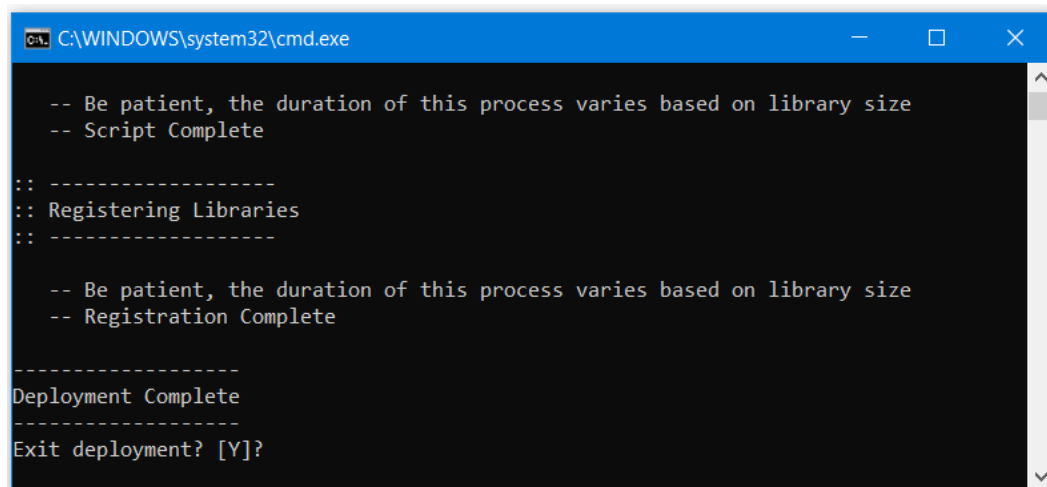
Using Studio 5000® Application Code Manager is not mandatory although it is highly recommended to reduce the likelihood of configuration errors and simplify the workflow. Alternatively, you can import the RUNG.L5X files directly into a Studio 5000® project.



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.



```

C:\WINDOWS\system32\cmd.exe

-- Be patient, the duration of this process varies based on library size
-- Script Complete

:: -----
:: Registering Libraries
:: -----

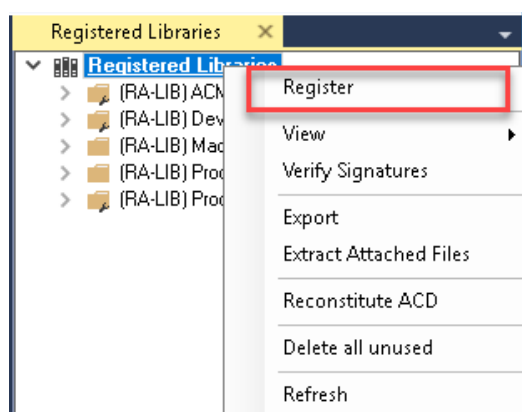
-- Be patient, the duration of this process varies based on library size
-- Registration Complete

-----
Deployment Complete
-----
Exit deployment? [Y]?

```

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 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 videos for this section: “How_To_Import_and_Configure_MettlerToledo_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.

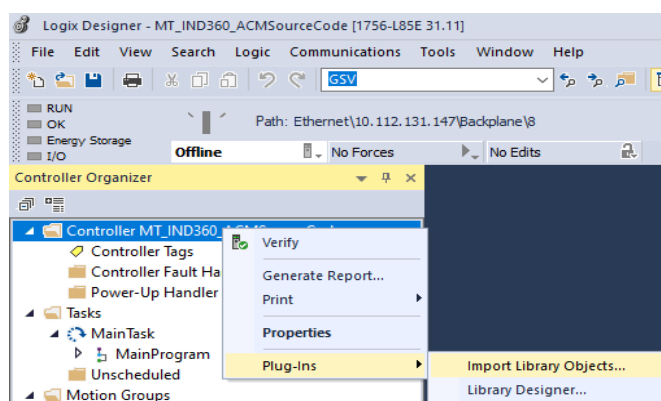
Below is a table to capture recommendations on when to use which tool or workflow when importing and configuring device objects.

Tool/Workflow	Description of when to use	Software Requirements
Application Code Manager (full application)	Project is developed 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 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®

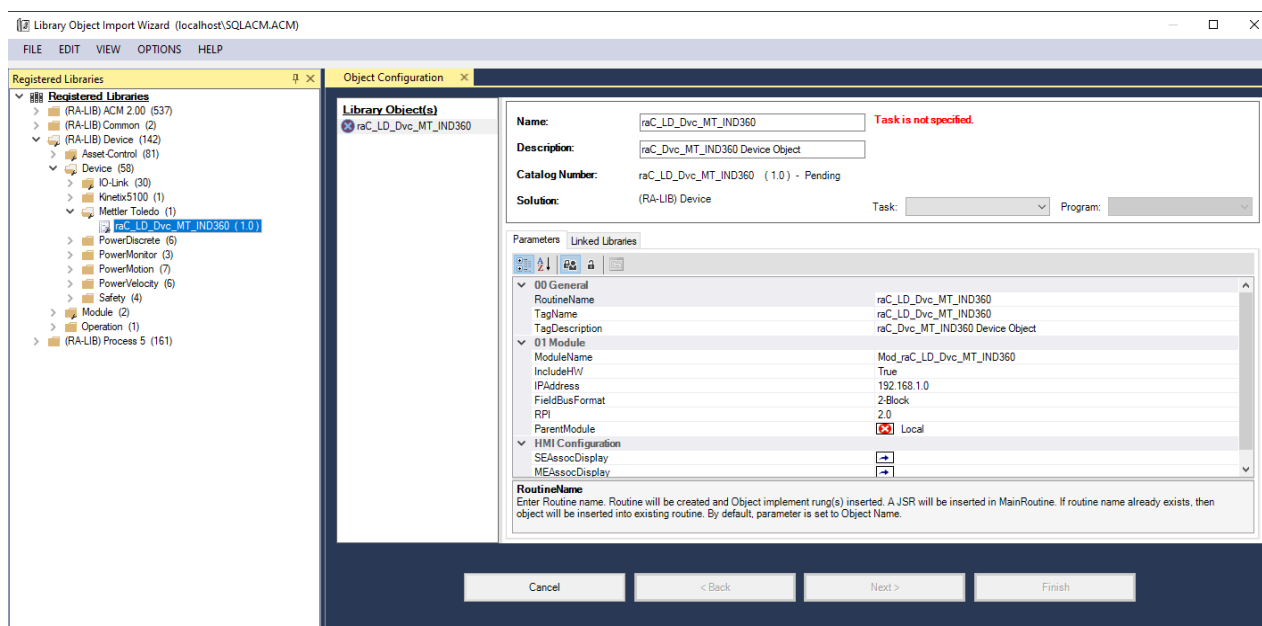
Import Library Objects Wizard

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 the **Module** (associated hardware e.g. IND360)
 - Set **IncludeHW** to True if you would like the wizard to add a new module (e.g. IND360) to your hardware tree. Set this to False if you already have the module pre-existing in your hardware tree.
 - Set the **ModuleName**. If IncludeHW is false, set this to the name of the existing module. If IncludeHW is true, set this to the desired name of the module that will be created.
 - Set the **IPAddress** to the IP Address of the device.

- Select the **FieldBusFormat** and **RPI** Parameters. The *FieldBusFormat* will default to 2-Block and *RPI* will default to 2.0ms.
- Set the **ParentModule** to name of the network card that the device is connected to. If using the embedded Ethernet port of the processor module, leave as Local.
- The HMI Configuration options are not used in the Plug-In Wizard and can be ignored.

Name:	IND360_MT1002		
Description:	Weighing Terminal MT1002		
Catalog Number:	raC_LD_Dvc_MT_IND360 (1.0) - Pending		
Solution:	(RA-LIB) Device	Task: MainTask	Program: MainProgram

Parameters		Linked Libraries
<div> </div>		
00 General	RoutineName: IND360_MT1002 TagName: IND360_MT1002 TagDescription: Weighing Terminal MT1002	
01 Module	ModuleName: Mod_IND360_MT1002 IncludeHW: True IPAddress: 192.168.1.0 FieldBusFormat: 2-Block RPI: 2.0 ParentModule: Local	
HMI Configuration	SEAssocDisplay: MEAssocDisplay:	
RoutineName 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.		

- 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.

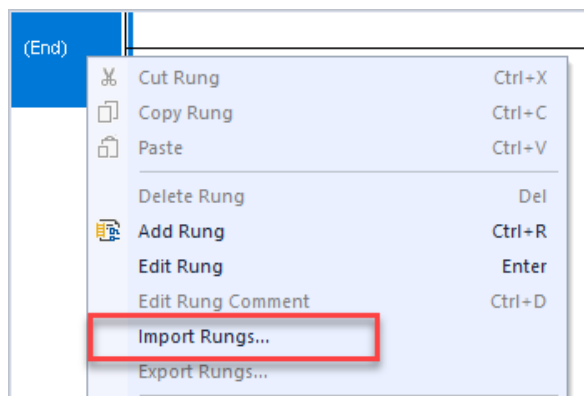
Import Rung Logic

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 RUNG.L5X file into a controller project.

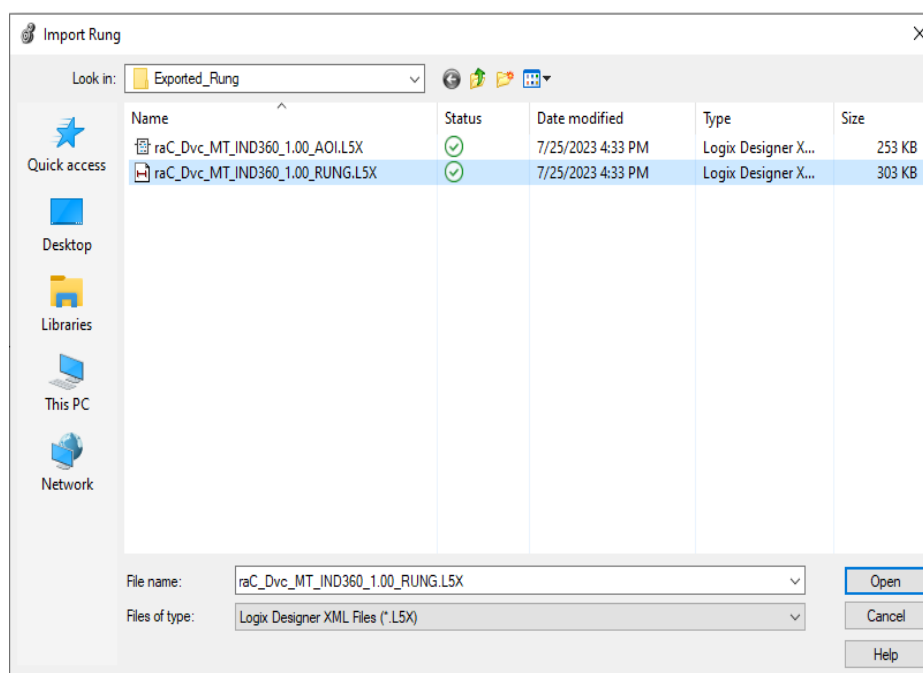
1. In the Studio 5000 Logix Designer® application, open a new or existing project.

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

2. Choose or create a new ladder routine to open. Right-click in the routine ladder and choose Import Rungs...



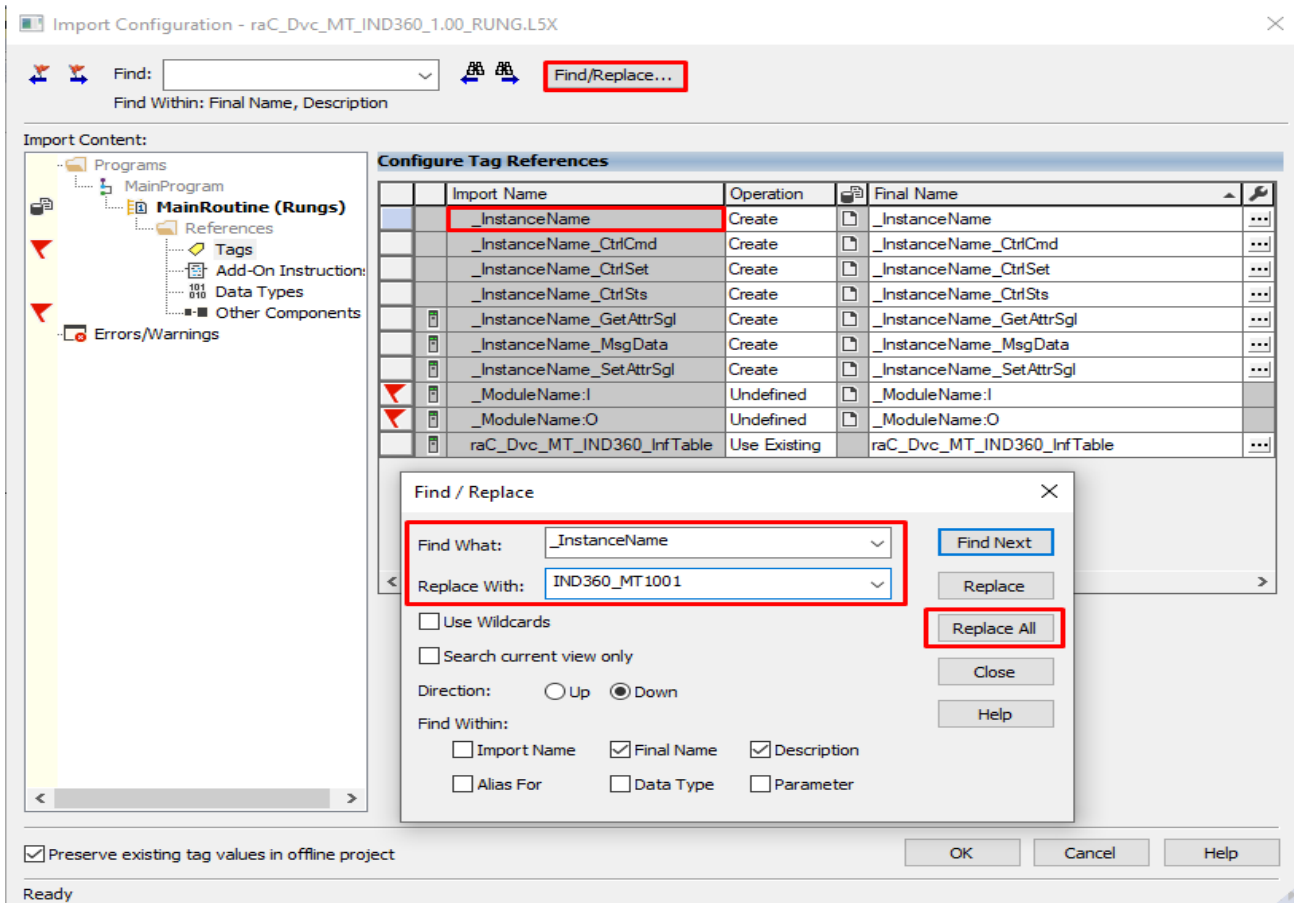
3. Select the desired RUNG and Select Import. The file will have a name like *raC_Dvc_Object_x.yy_RUNG.L5X*.



Both "RUNG" and "AOI" .L5X files are provided. Import the RUNG file 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...*

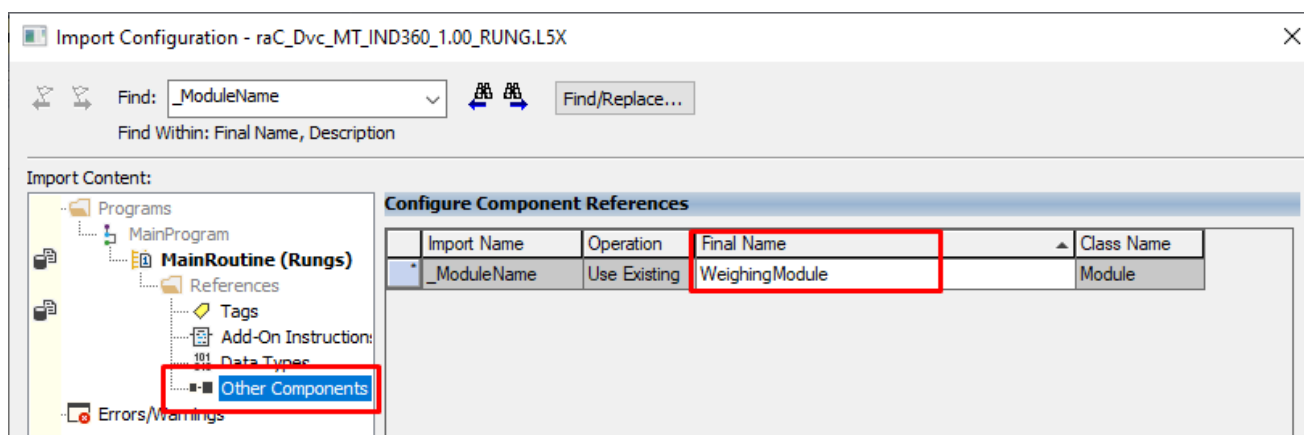
button and replace all instances of “_InstanceName” with your desired device name (e.g. “IND360_MT1001”).



- You will need to point the new object to the correct module in your project. You can type in or browse for the correct input (:I) and Output (:O) tags in your project. In this example our module is called Mod_Motor101.

Configure Tag References			
	Import Name	Operation	Final Name
	_InstanceName	Create	_InstanceName
	_InstanceName_CtrlCmd	Create	_InstanceName_CtrlCmd
	_InstanceName_CtrlSet	Create	_InstanceName_CtrlSet
	_InstanceName_CtrlSts	Create	_InstanceName_CtrlSts
	_InstanceName_GetAttrSgl	Create	_InstanceName_GetAttrSgl
	_InstanceName_MsgData	Create	_InstanceName_MsgData
	_InstanceName_SetAttrSgl	Create	_InstanceName_SetAttrSgl
	raC_Dvc_MT_IND360_InfTable	Use Existing	raC_Dvc_MT_IND360_InfTable
*	_ModuleName:I	Use Existing	WeighingModule:I
*	_ModuleName:O	Use Existing	WeighingModule:O

- Click on the *Other Components* section and type or browse to the name of the module. In this example our module is called WeighingModule.



7. The rung will now be imported into your ladder routine.

Using Studio 5000 View Designer®

Using View Designer Project Files

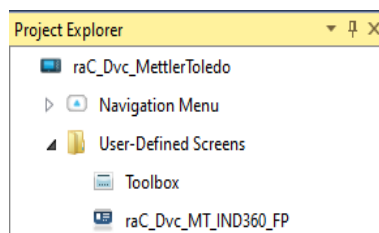
Studio 5000 View Designer® may be used for HMI development for PanelView™ 5000 applications. Open up your Studio 5000 View Designer® project alongside a second application instance running the required VPD file in the library folder *HMI - ViewDesigner - vpd*.



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 videos for this section:
 “How_To_Configure_MettlerToledo_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.

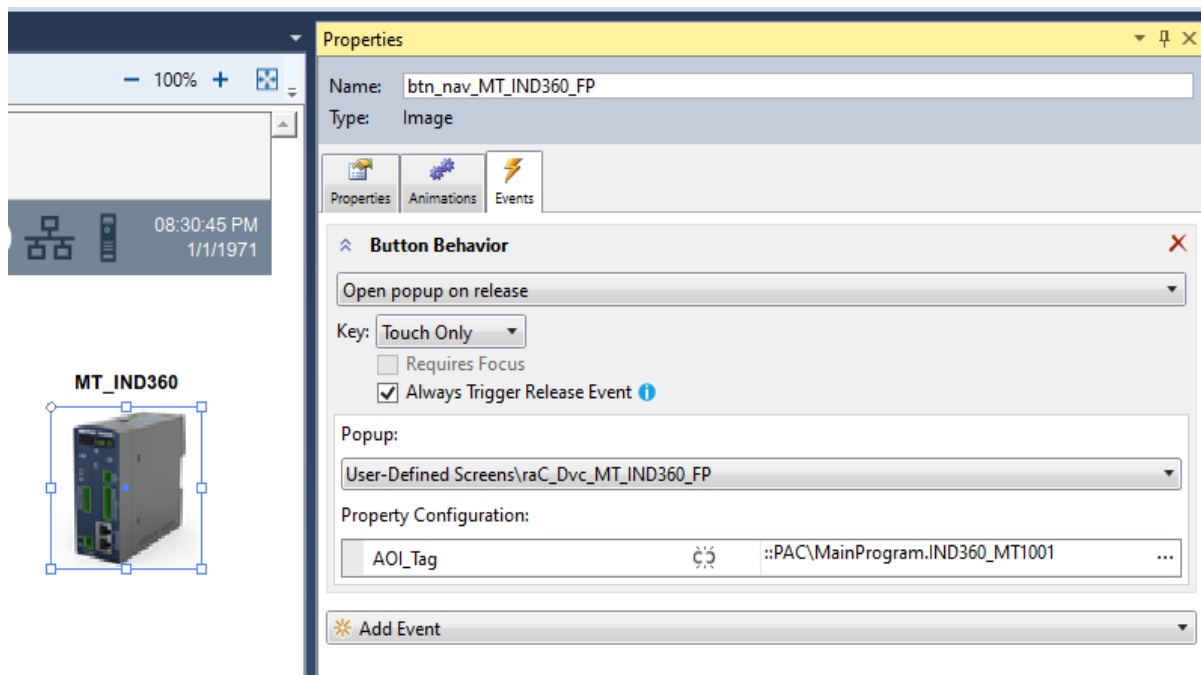
MT_IND360



Configuring View Designer Objects

To link the launch button to the faceplate, highlight the button and view the *Events* tab of within the *Properties* pane. Set an Event to *Open popup on release* with the following settings:

- Key: Touch Only
- Popup: Select desired faceplate screen
- AOI_Tag: Browse to AOI backing tag for the device object in your controller file



Using FactoryTalk® View Studio

Import FactoryTalk® View Visualization Files

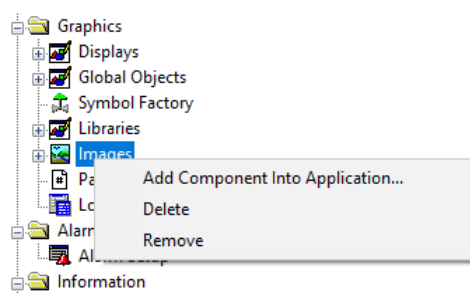
There are several components to import for the visualization files. You import files from the downloaded Rockwell Automation® library files via FactoryTalk® View SE/ME. The workflow is the same for both FactorTalk® View ME and SE.



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 videos for this section:

"How_To_Import_and_Configure_MettlerToledo_Objects_in_FTVViewME_SE"

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

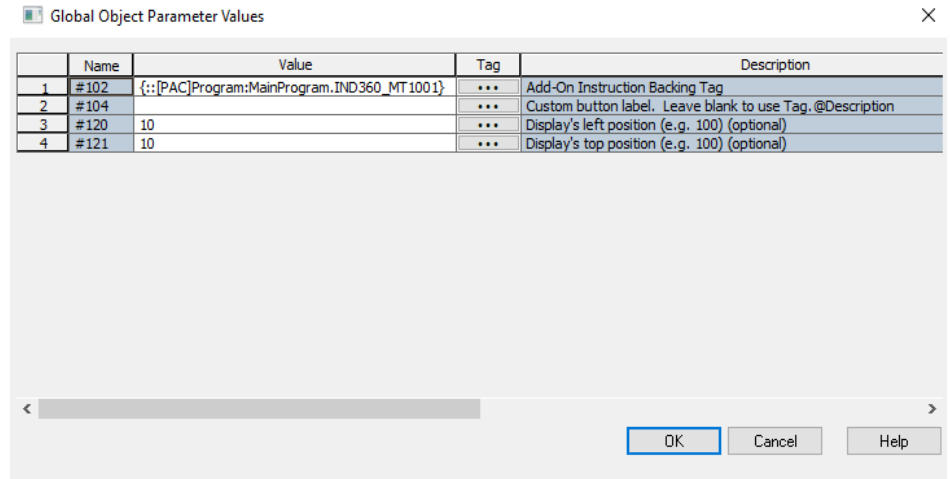
Select the faceplate (.gfx) files from the *\HMI - FactoryTalk® View ME\Displays - gfx* or *\HMI - FactoryTalk® View SE\Displays - gfx* folder

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 - Mettler Toledo Device* or *(raC-X-SE) Graphic Symbols - Mettlor Toledo 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 Backing Tag (#102) and Navigation Button Label (#104) are mandatory while the display position values (#120, #121) are optional. 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

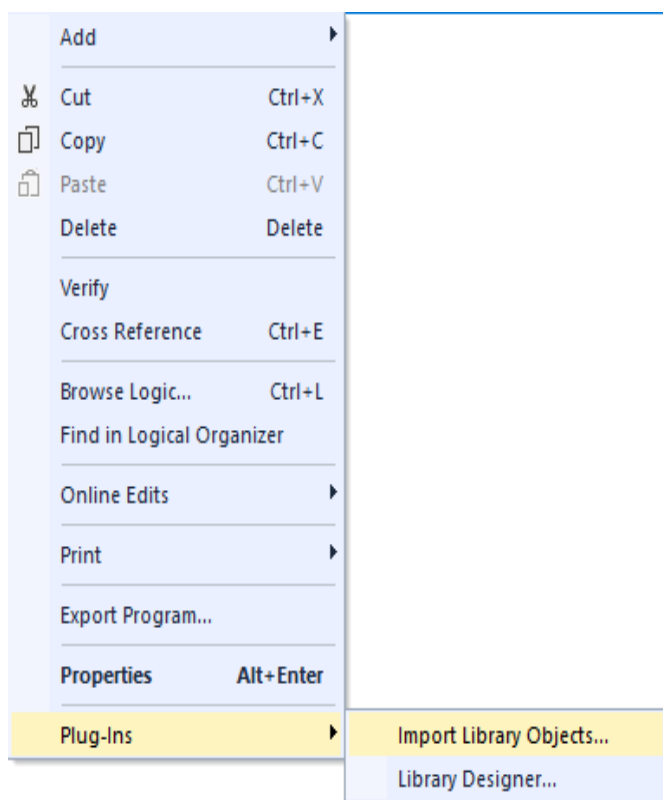
There are two methods to upgrading existing device object add-on instructions in a project. You can do this either by using the Studio 5000® Plug-In *Import Library Objects* Wizard or by importing individual add-on instruction AOI.L5X files. Using the Studio 5000® Plug-In *Import Library Objects* Wizard is the preferred method to reduce the risk of errors or compatibility issues. 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.

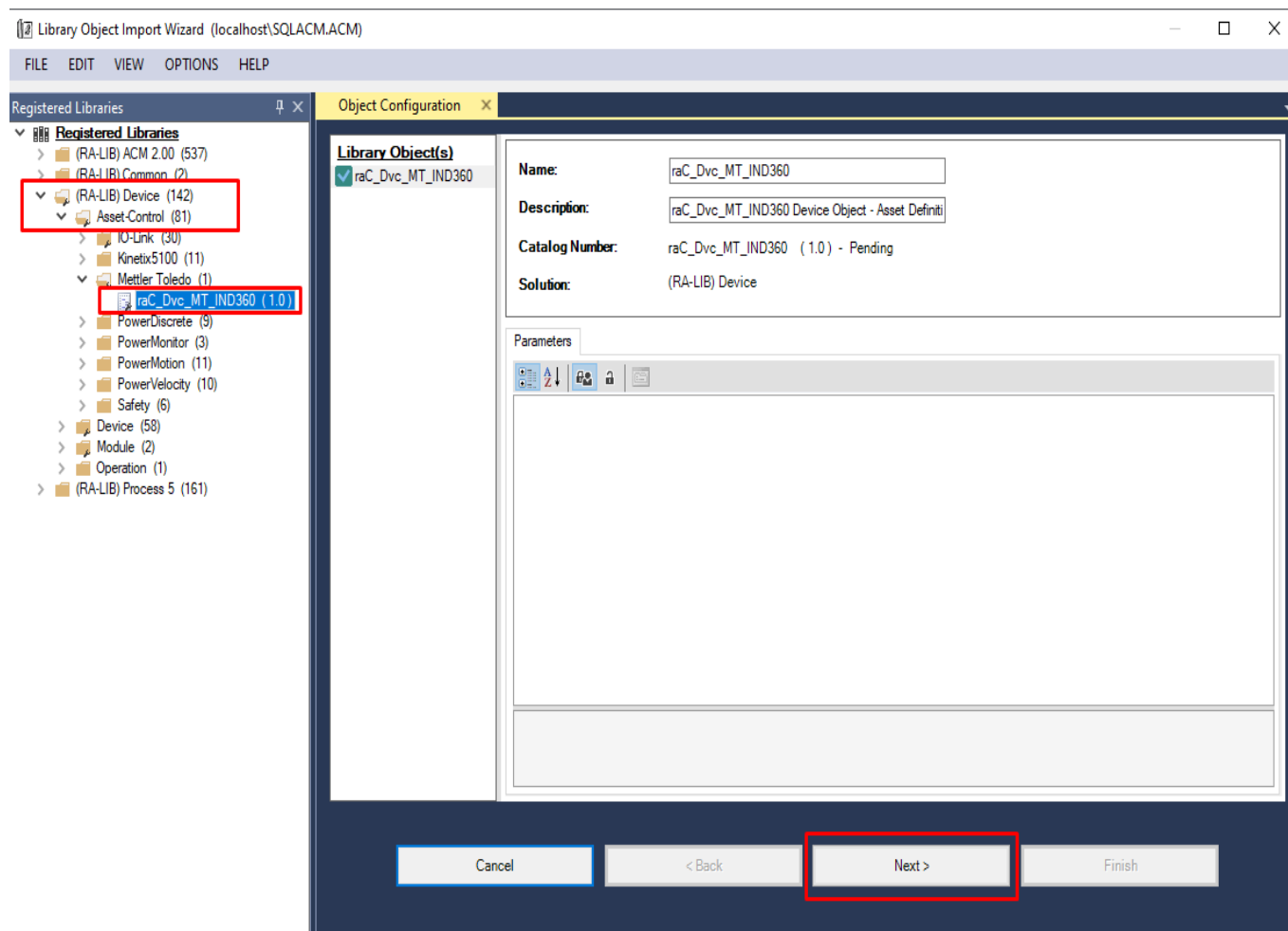
Upgrades Using Studio 5000® Plug-In to Import Library Objects

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](#).

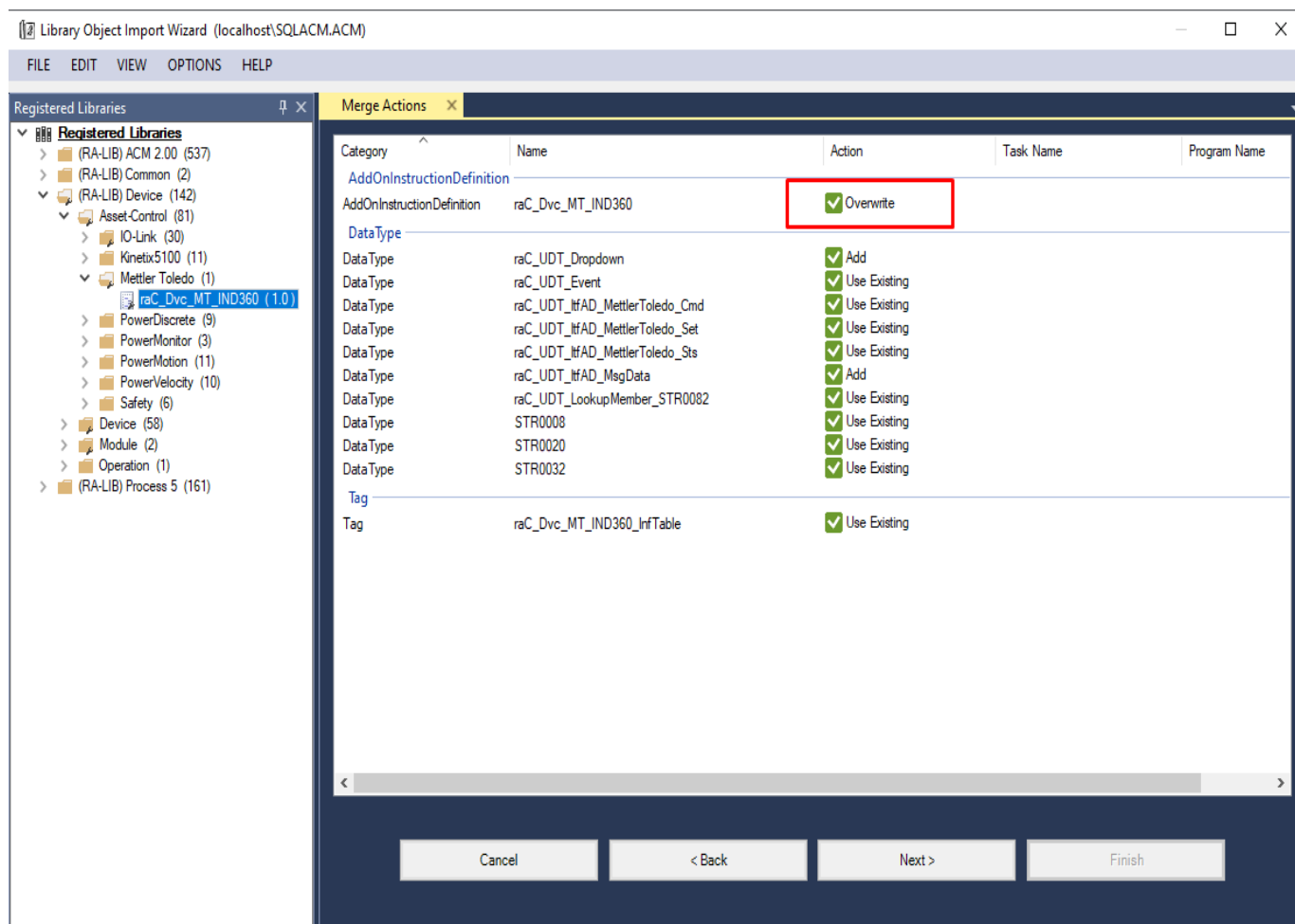
Right-click in your controller organizer or within a routine to access *Plug-Ins > Import Library Objects...*



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



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.



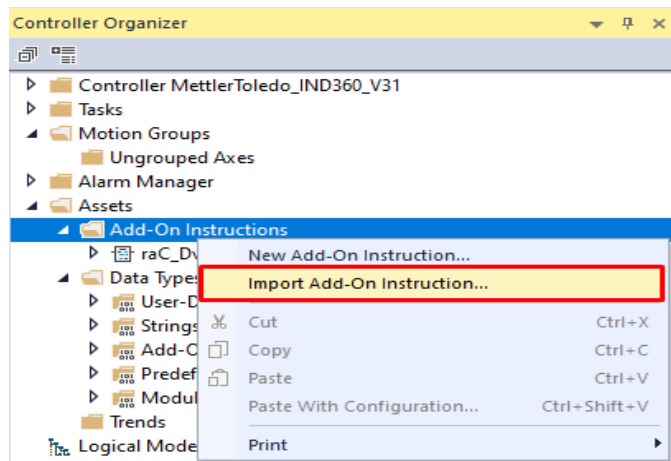
Upgrades by Importing AOI.L5X Files

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

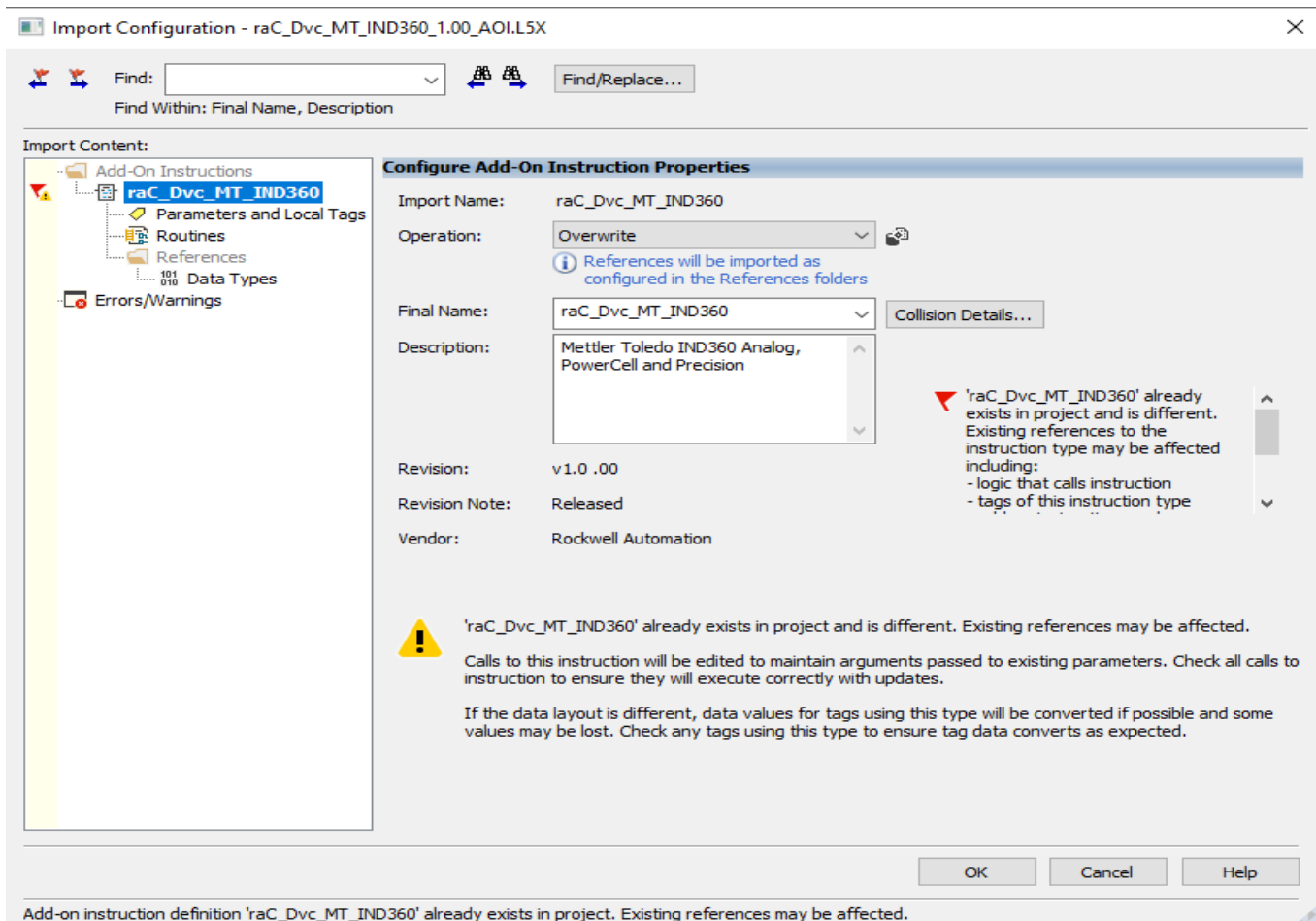
IMPORTANT In newly released versions of this library, it is possible that modifications or improvements have been made to items such as tags, faults tables, etc that are outside of the Add-On Instruction. To ensure all items are updated use the Studio 5000® Import Library Objects plug-in method or import RUNG.L5X files. Read the library release notes to understand what has been updated in the latest version of the library.

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.

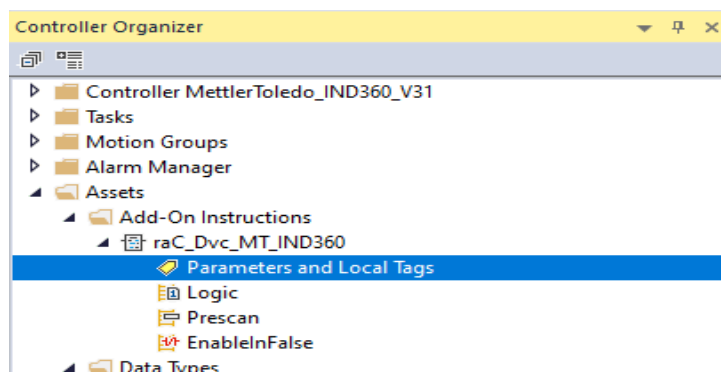


- You will be prompted that there is an existing version of the instruction that is different. Choose *Overwrite* as the operation and select OK once you have read and understood the warnings. Your existing logic will be updated with the new add-on instruction. Verify that your code compiles and test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

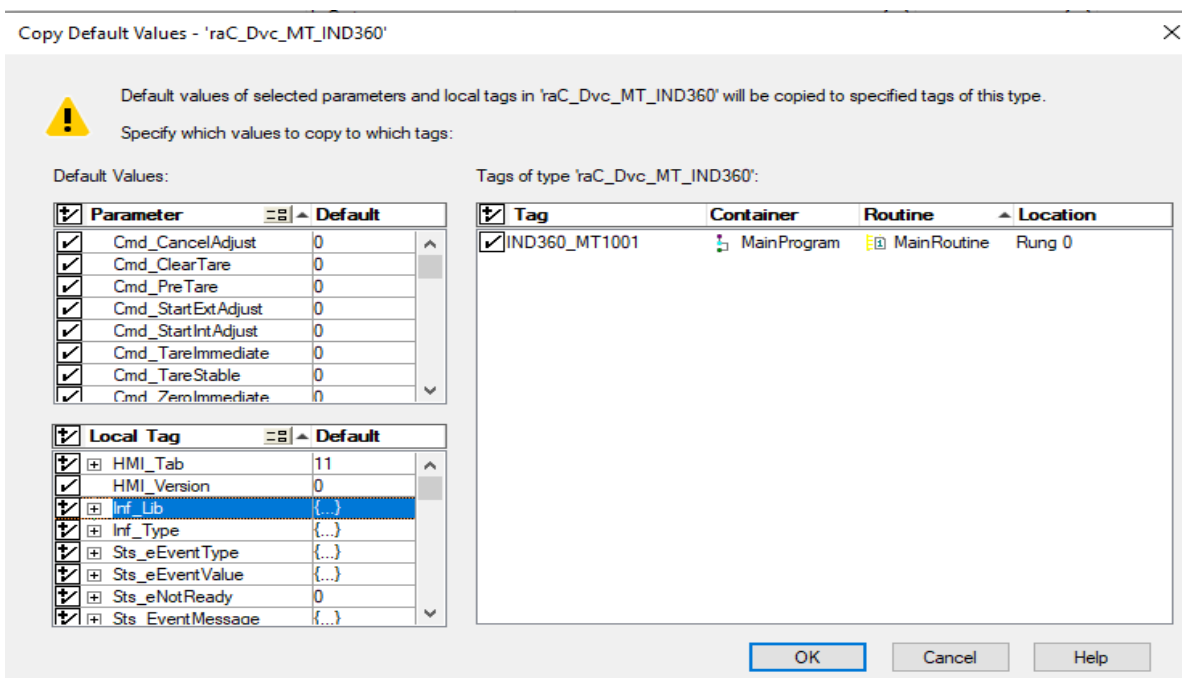
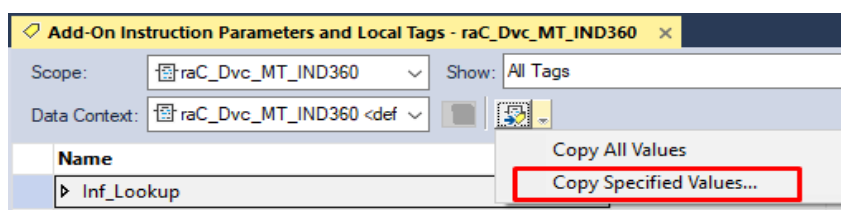


- In order to ensure the HMI faceplate still works properly you will need to update the object's library information stored in the Inf_Lib tag. In

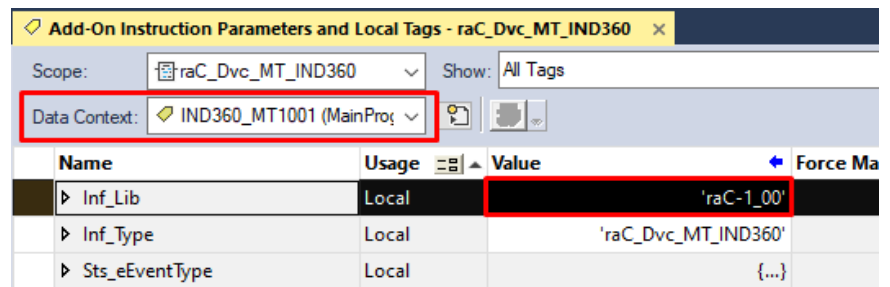
the *Controller Organizer* pane under *Assets > Add-On Instructions* expand the device object that was updated. Double-click on *Parameters and Local Tags* to open up the instructions tags.



- 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-1_00') 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 import the 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_Configure_MettlerToledo_Device_Objects_in_ACM"

Creating a New Project

Begin by opening Application Code Manager.

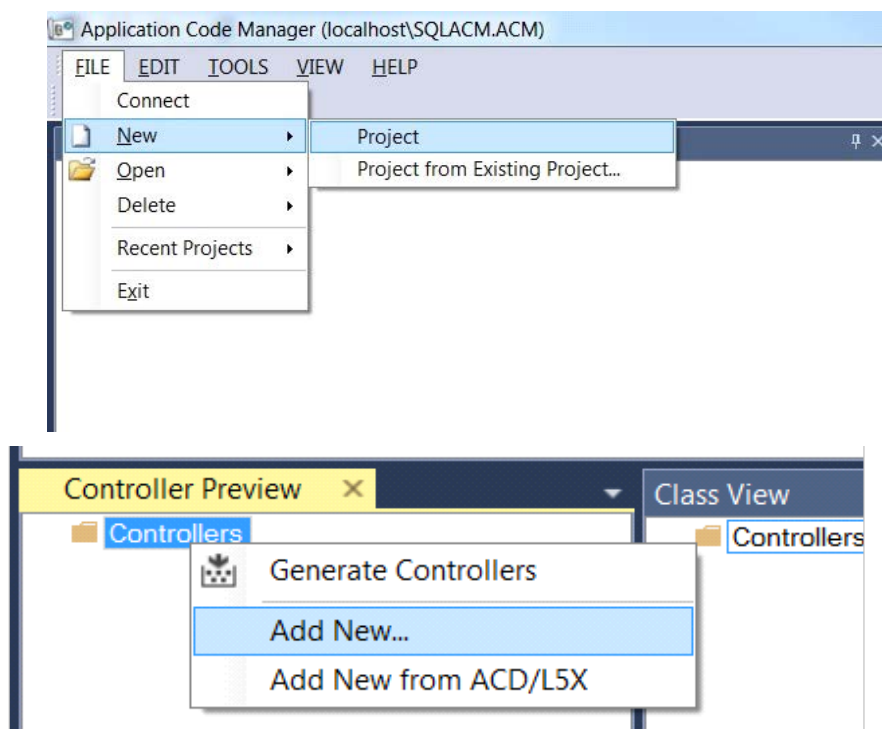


Note: the last project (if any) is opened by default; otherwise a blank screen is displayed.

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. *ControlLogix_Controller*, *GuardLogix_Controller*, *CompactLogix_Controller*, etc). Enter a *Name* and *Description* for the controller. Select the appropriate Chassis and Processor configurations.

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

Object Configuration Wizard

Name: MettlerToledo_Device_v1_00

Description: ControlLogix Controller

Catalog Number: ControlLogix_Controller (2.1) - Published

Solution: (RA-LIB) ACM 2.00

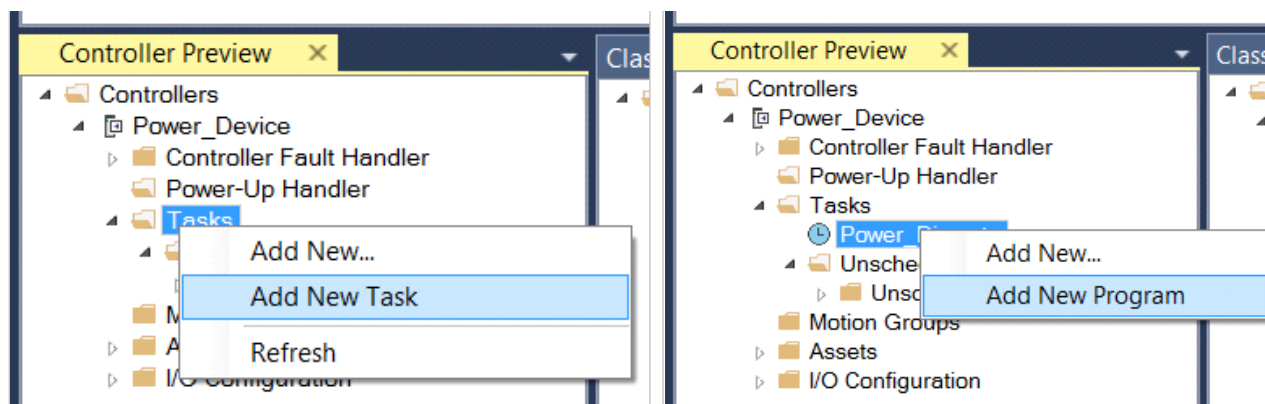
Parameters

01 - Controller	
ChassisName	Local
Slot	0
Size	17
SoftwareRevision	33
ProcessorType	1756-L85EP
PlantPAxTaskingModel_Enabled	False
02 - HMI	
AreaPath	/Area::[shortcut]
AreaPathME	[PAC]
03 - Historian	
HistorianPath	Application/Area:RSLink Enterprise:[shortcut]
FTLDInterfaceNo	1
Motion	
ConfigureMotion	False
Port Configuration	
EthernetPort1_Enabled	True
Time Synchronization	
TimeSync_Priority1	128
TimeSync_Priority2	128
TimeSync_PTPEnable	True

Size
This is the Chassis size of this Controller

Cancel << Previous Next >> Finish

You can now add in any desired tasks and programs to your controller. Right-Click on the *Tasks* folder underneath your controller in the *Controller Preview* and *Add New Task*. Similarly, right-click on any Task and select *Add New Program*. Complete the desired parameters for Tasks and Programs such as name, type, period, etc.

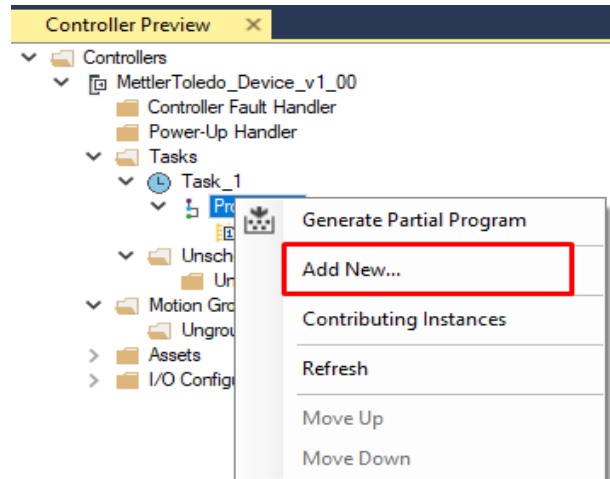


Adding & Configuring Device Objects

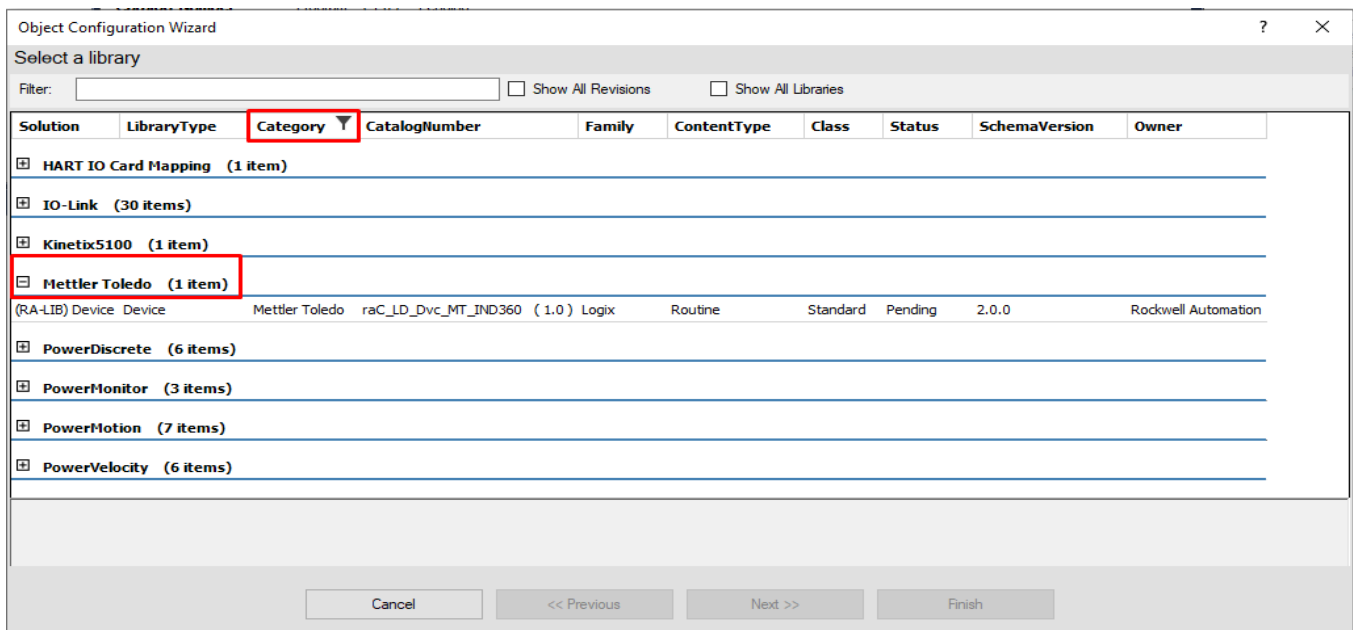
Prior to adding in any Device Objects, ensure you have registered the library in Application Code Manager. Refer to [Registering Libraries in Studio 5000® Application Code Manager](#) for details.

Adding Mettler Toledo Device Objects

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



Select the Device Object that you wish to import. You can click on the *Category* heading to group items by category and easily find *Mettler Toledo* devices.



Fill in all of the required configuration parameters for the device object. The following example shows a configuration of the raC_LD_Dvc_MT_IND360 object.

Perform the following configuration:

o0 General:

- 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**. Note these are pre-assigned if you added the object to a specific program by right-clicking in the *Controller Preview* pane.

o1 Module:

- The **ModuleName** will default to *Mod_ObjectName*
- Set **IncludeHW** to True to allow Application Code Manager to create the Mettler Toledo Module.
- Assign **IPAddress** along with the **ParentModule**. Note that *ParentModule* may show a red 'x' when using "Local" or an ethernet adapter that is not created in the Application Code Manager project. "Local" is used for embedded ethernet ports such as in the L8xE controller family.
- Select the **FieldBusFormat** and **RPI** Parameters. The *FieldBusFormat* will default to 2-Block, and *RPI* will default to 2.oms.

HMI Configuration:

- Set *SEAssocDisplay* to a FactoryTalk® View SE display configured in the project if you plan on generating displays using Studio 5000® Application Code Manager.
- Set the desired *SymbolStyle* to either "GraphicalButton" to use the graphical style launch button for schematic/system displays; or "GenericTextButton" to use a simple rectangular text button. For more information on HMI Configuration refer to [Configuring Displays](#).

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

Object Configuration Wizard

Name: WeighingTerminal_1001

Description: WT_IND360_1001

Catalog Number: raC_LD_Dvc_MT_IND360 (1.0) - Pending

Solution: (RA-LIB) Device Task: Task_1 Program: Program_1

Parameters Linked Libraries

00 General

RoutineName	WeighingTerminal_1001
TagName	WeighingTerminal_1001
TagDescription	WT_IND360_1001

01 Module

ModuleName	Mod_WeighingTerminal_1001
IncludeHW	True
IPAddress	192.168.1.23
FieldBusFormat	2-Block
RPI	2.0
ParentModule	Local

HMI Configuration

SEAssocDisplay	FTViewSE_Server.GraphicDisplays.IND360_SELaunch
MEAssocDisplay	FTViewME_Panel.GraphicDisplays.IND360_MELaunch

MEAssocDisplay
Associated HMI ME Display

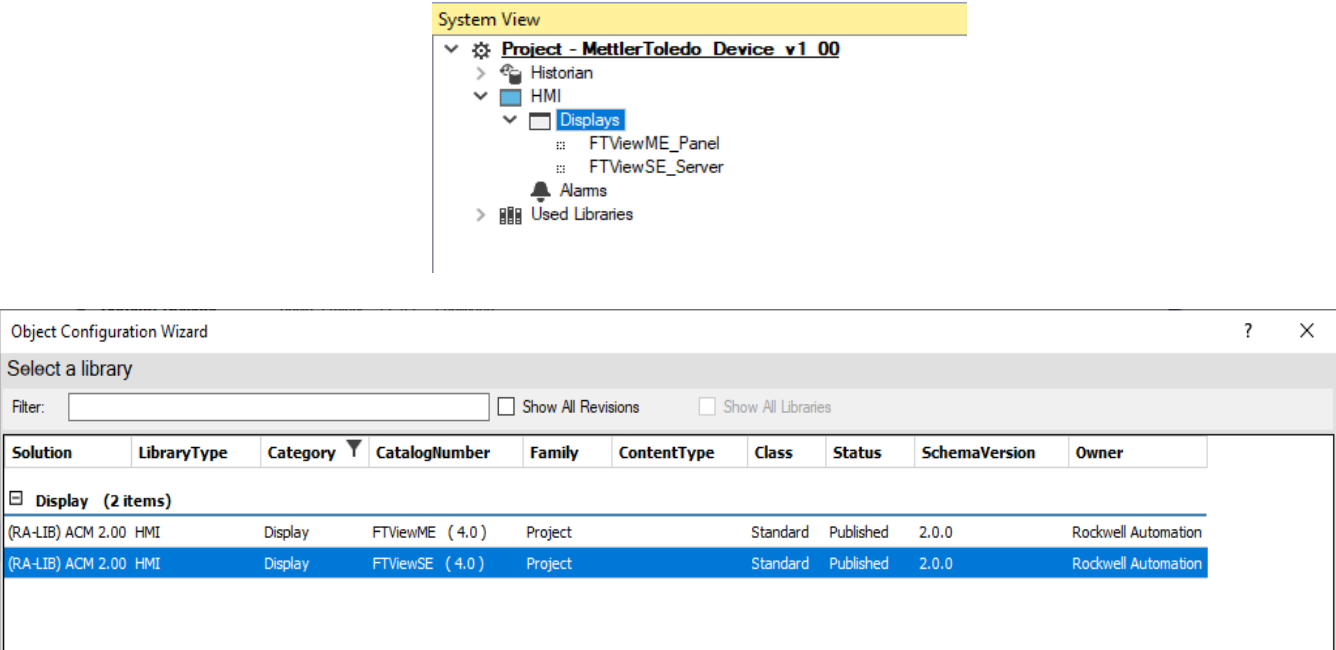
Cancel << Previous Next >> Finish

Click Finish to complete the import. For specific devices details, refer to the appropriate chapter in this manual.

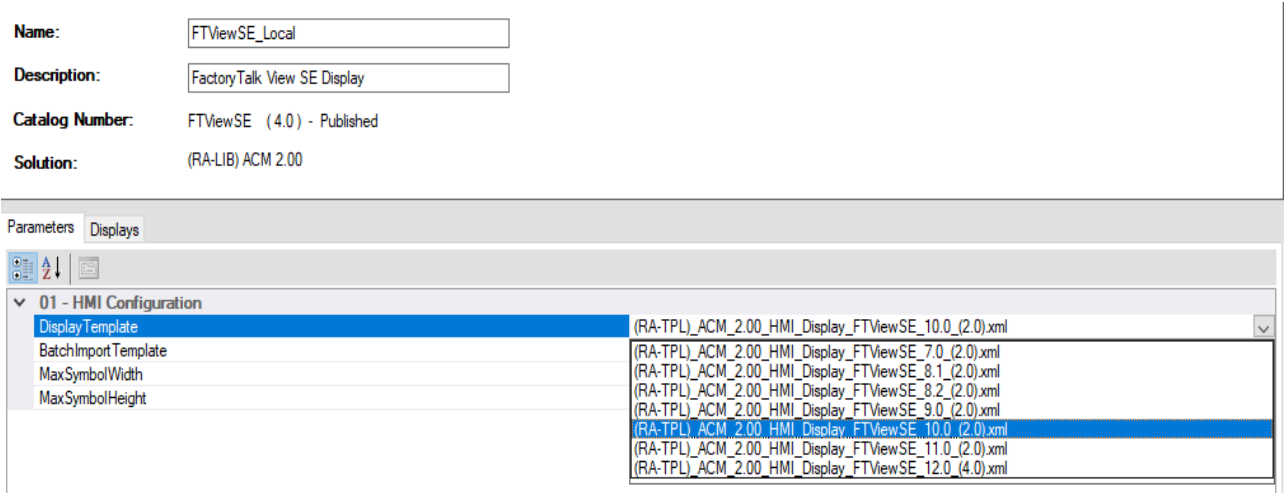
Configuring Displays

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.

Name: FTViewSE_Local

Description:

Catalog Number: FTViewSE (4.0) - Published

Solution: (RA-LIB) ACM 2.00

Parameters		Displays	
Name	SubObject Description		
<div><div>Add New</div><div>Copy</div><div>Paste</div><div>Delete</div><div>Reset Grouping...</div></div>			

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.

Name: FTViewSE_Local

Description:

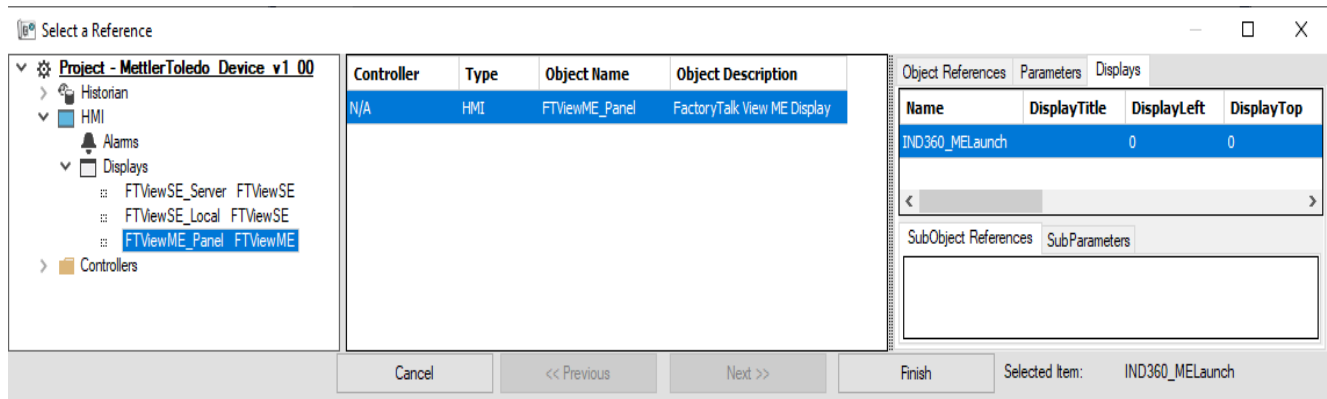
Catalog Number: FTViewSE (4.0) - Published

Solution: (RA-LIB) ACM 2.00

Parameters		Displays			
Name	DisplayTitle	DisplayLeft	DisplayTop	DisplayWidth	
IND360_SELaunch		0	0	3000	

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.

▼ HMI Configuration	
SEAssocDisplay	➡ FTViewSE_Server.GraphicDisplays.IND360_SELaunch
MEAssocDisplay	➡ FTViewME_Panel.GraphicDisplays.IND360_MELaunch

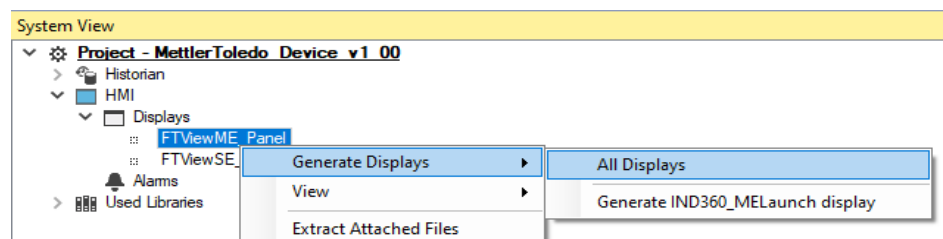


If you browse for the display, select the desired display server in the left panel, then click on the *Display* tab in the right panel and select the specific display. Click finish.

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

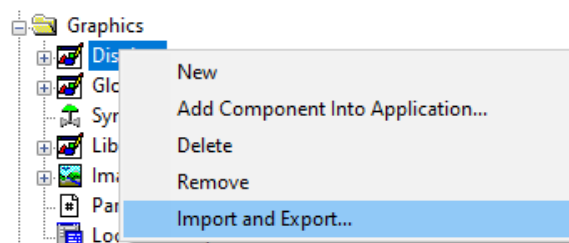
Generating Displays

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

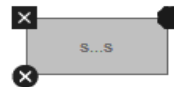
To import the configured displays, open your FactoryTalk® View ME/SE project in FactoryTalk® View Studio. Right-click on *Graphic* > *Displays* and select *Import and Export...*



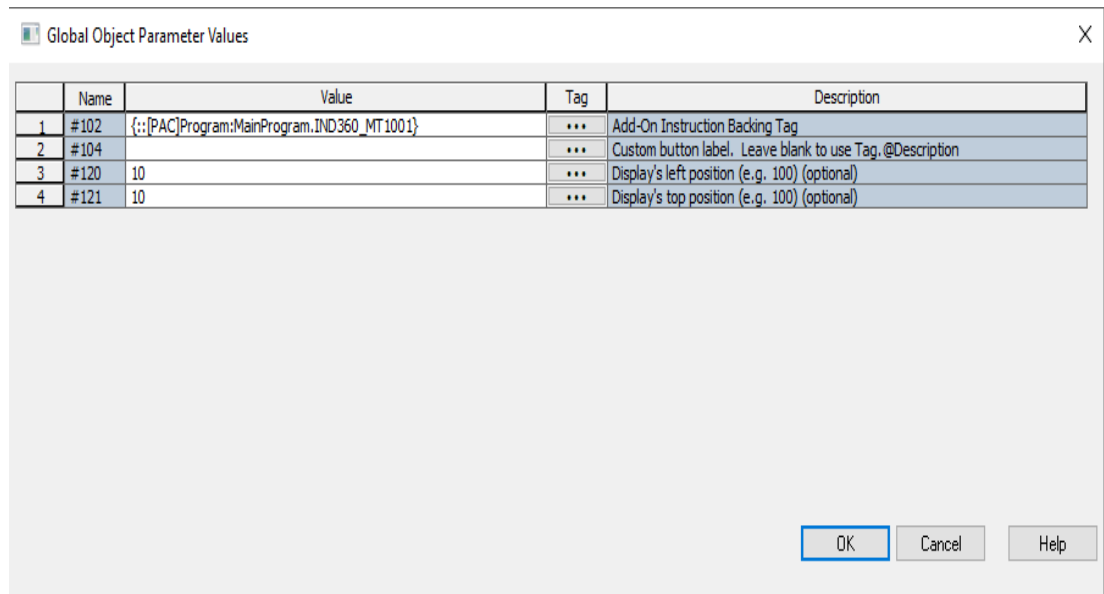
Follow the required prompts:

- 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.

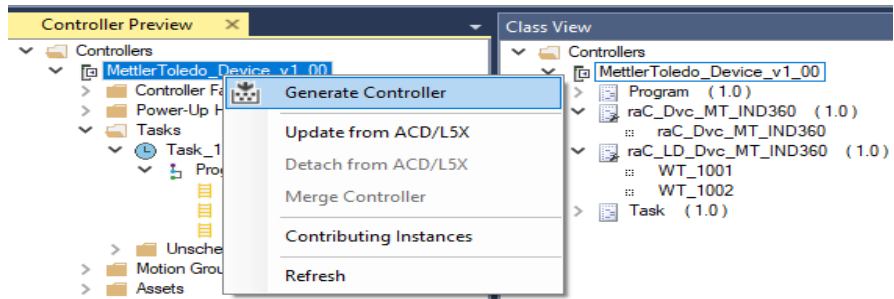


You may not copy and paste this graphic symbol onto any other display in your application.

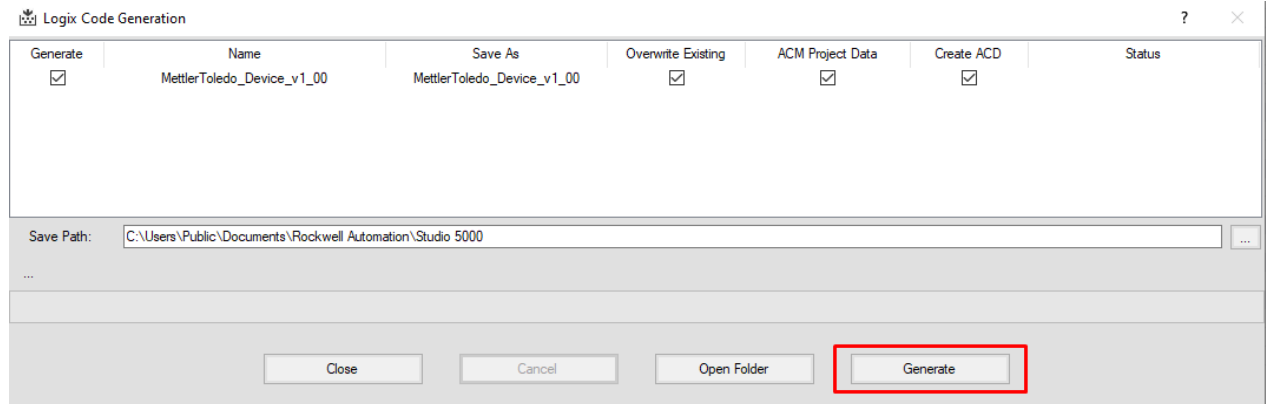
Generating Controller Files

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 may also need to check *Overwrite Existing* if this is not the first time generating the controller code.



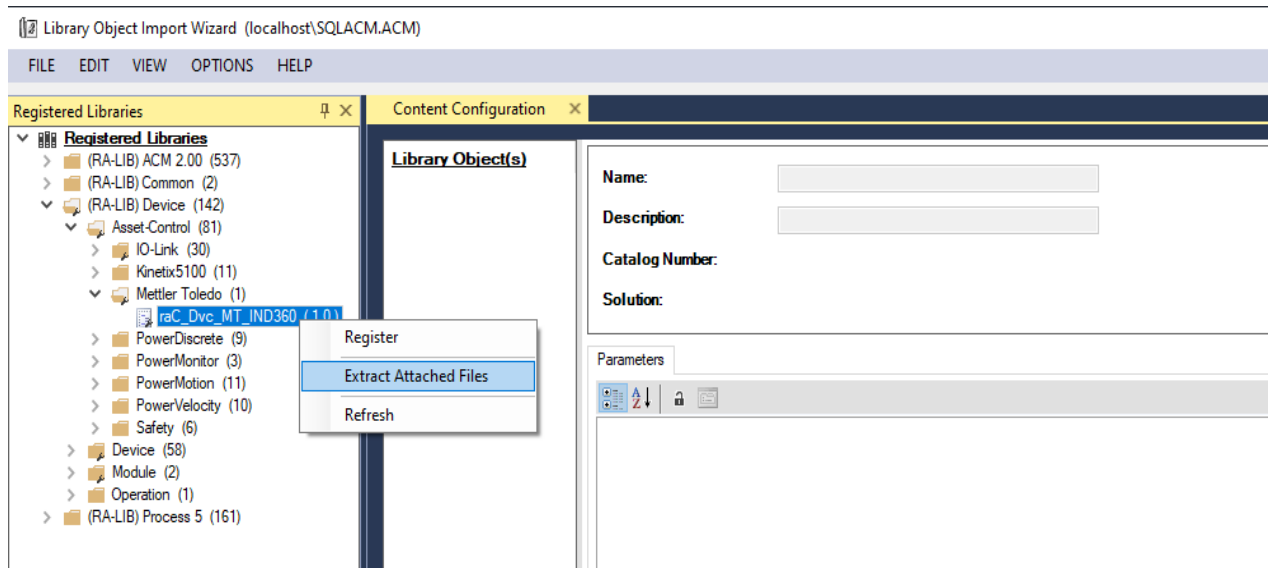
Once the controller file is generated, you can navigate to the location set in *Save Path* and open your file. Note that all of the configuration that was done in Application Code Manager is now shown in your Logix Designer ACD file.

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_MT_IND360*) 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
- FactoryTalk® View Site Edition Display
- FactoryTalk® View Site Edition Global Objects
- FactoryTalk® View Machine Edition Display
- FactoryTalk® View Machine Edition Global Objects
- View Designer Faceplate File

Using the Mettler Toledo Device Library with Other Application Code Libraries

Application Code Libraries

The Mettler Toledo Device Library is commonly used alongside other Application Code Libraries.

Other libraries utilize the common device interface UDTs to interact with device level objects. This is covered in detail in [Interfaces](#) section of this document.

Using Mettler Toledo Device Objects with PlantPax® Process Objects Library

The PlantPax® Process Object Library application level library objects may be used in the same applications with the device-level objects in the Mettler Toledo 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.



PlantPax® 5.xx only supports FactoryTalk® View SE as an HMI platform. You should not be using the FactoryTalk® View ME or Studio 5000 View Designer® faceplates with a PlantPax® application.

Using Mettler Toledo Device Objects with Machine Builder Library

The Machine Builder Library objects may be used in the same applications with the device-level objects in the Mettler Toledo 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.

Mettler Toledo IND360 (raC_Dvc_MT_IND360)

Overview

The Mettler Toledo device object includes a faceplate which displays status and configuration information of Mettler Toledo IND360 Indicator and Transmitter (raC_Dvc_MT_IND360). The IND360 Weighing Terminal faceplate displays weight and load cell readings, along with the status of motion, center of zero, and device failure reasons. It also allows operators to perform tare and zero commands, and configure precision calibrations.



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_MT_IND360_Faceplate"

Functional Description

The IND360 Weighing Terminal 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. 1.00) used in file names 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
IND360	raC_Dvc_MT_IND360_1.00_AOI.L5X	raC_Dvc_MT_IND360_1.00_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
IND360	Display	(raC-1.00-ME) raC_Dvc_MT_IND360-Faceplate.gfx	(raC-1.00-SE) raC_Dvc_MT_IND360-Faceplate.gfx
Graphic Symbols	Global Object	(raC-1-ME) Graphic Symbols - MettlerToledo Device.ggfx	(raC-1-SE) Graphic Symbols - MettlerToledo Device.ggfx
Toolbox	Global Object	(raC-1-ME) Toolbox - MettlerToledo Device.ggfx	(raC-1-SE) Toolbox - MettlerToledo 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
IND360	(raC-1.00-VD) raC_Dvc_MettlerToledo.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)
IND360	(RA-LIB)_Device_Asset-Control_MettlerToledo_raC_Dvc_MT_IND360_(1.0)	(RA-LIB)_Device_Device_MettlerToledo_raC_LD_Dvc_MT_IND360_(1.0)

Device Definition

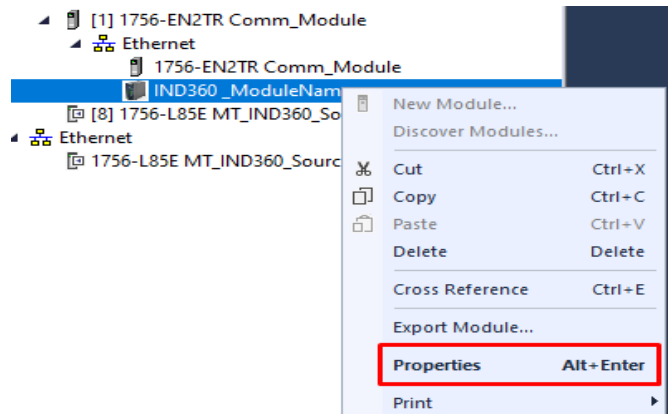
The device (ie: IND360) must be configured with the correct device definition. Proper device configuration enables the required cyclic device data to pass information from the device into the add-on instruction.



Note that this configuration is completed automatically when using Application Code Manager or the Studio 5000® Import Library Objects wizard plug-in.

To verify the device definition:

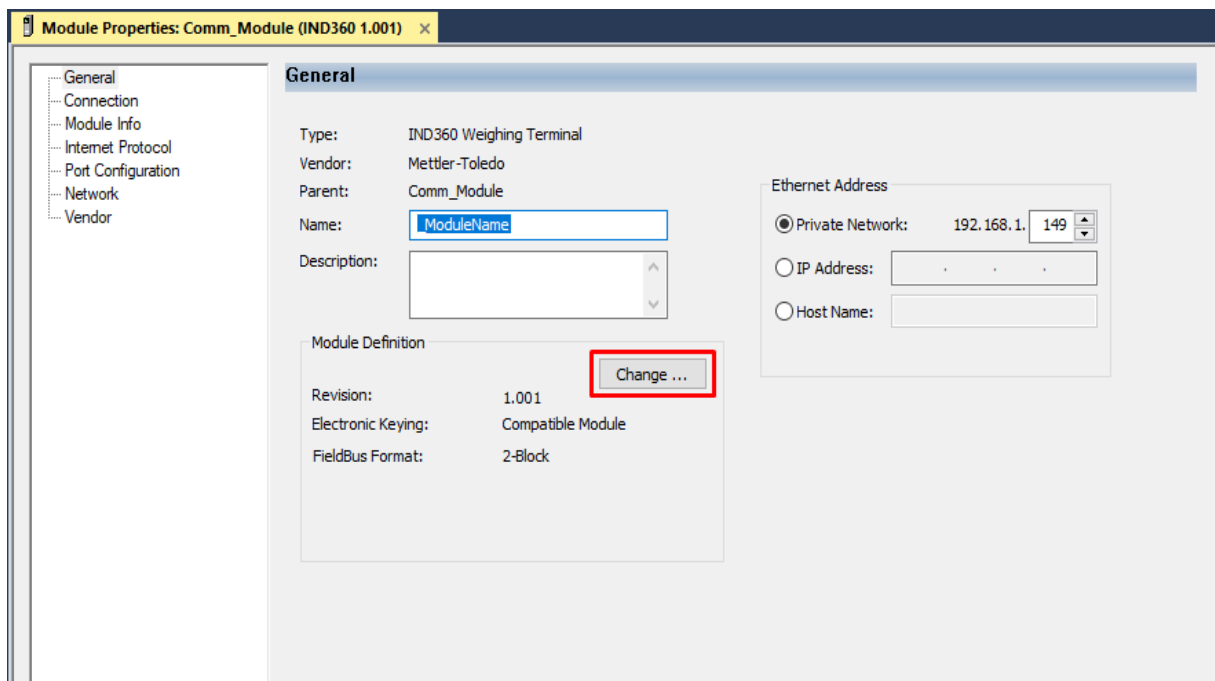
- Find the device in the *Controller Organizer* pane in Studio 5000 Logix Designer® and open the *Module Properties* by double-clicking or right-click and select *Properties*.



- Refer to the following sections for specific device configuration.

HARDWARE Definition

- On the *General* tab click on the *Change...* button.



- Specify the FieldBus Format that is selected on the device.

Operations

The Mettler Toledo objects provide only physical mode of operation. There is no virtual device mode offered.

Faults & Warnings

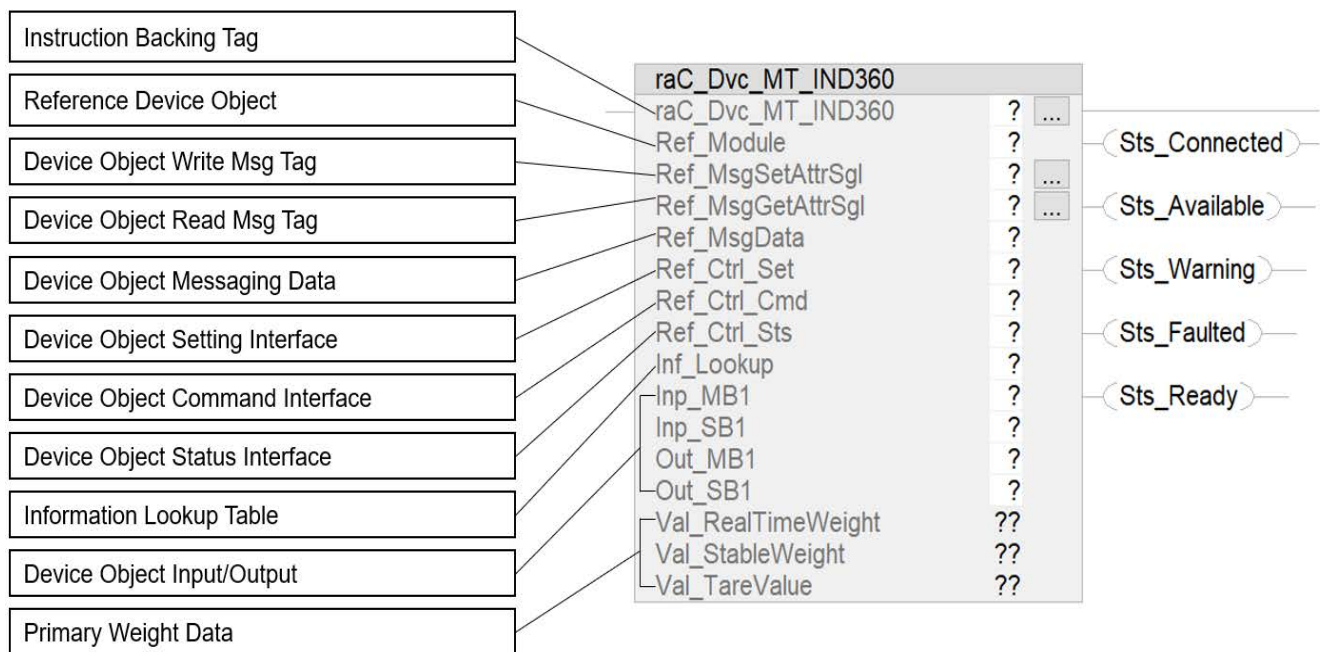
- **First Warning:** This function helps in capturing the first warning triggered in the device. Display the respective description in faceplate.
- **First Fault:** Capture the first fault from device. Display the respective description in faceplate.
- **Event log:** Log Warning and Fault the last 4 events in a log queue. The queue contains fault code, description, and time stamp. Display the same in faceplate.

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 Add-On Instruction Ladder Implementation



InOut Data

InOut	Function / Description	DataType
Ref_Module	Reference to module in I/O tree	MODULE
Ref_MsgGetAttrSgl	Message Configuration Get Attribute Single	MESSAGE
Ref_MsgSetAttrSgl	Message Configuration Set Attribute Single	MESSAGE
Ref_MsgData	Messaging Data	raC_UDT_ItfAD_MettlerToledo_MsgDa
Ref_Ctrl_Cmd	Mettler Toledo Device Command Interface	raC_UDT_ItfAD_MettlerToledo_Cmd
Ref_Ctrl_Set	Mettler Toledo Device Setting Interface	raC_UDT_ItfAD_MettlerToledo_Set
Ref_Ctrl_Sts	Mettler Toledo Device Status Interface	raC_UDT_ItfAD_MettlerToledo_Sts
Inf_Lookup	Code / Description List Entry	raC_UDT_LookupMember_STR0082[31]
Inp_MB1	Device Object Inputs	ME:IND360_MBlockIn:I:0
Inp_SB1	Device Object Inputs	ME:IND360_StatusIn:I:0
Out_MB1	Device Object Output	ME:IND360_MBlockOut:O:0
Out_SB1	Device Object Output	ME:IND360_StatusOut:O:0

Input Data

Input	Function/Description	DataType
Cmd_CancelAdjust	1 = Cancel adjustment / test (Abort Test Function / Adjustment)	BOOL
Cmd_ClearTare	1 = Trigger execution of Clear Tare	BOOL
Cmd_ConfirmWtSpanAdjust	1 = Test weight matching the value in CurrentWeight is loaded onto the scale	BOOL
Cmd_ExtLoadWeight	User needs to load test weight corresponding to the value of External Weight and ack.	BOOL
Cmd_ExtUnloadWeight	User needs to remove all weight from the scale and ack.	BOOL
Cmd_PreTare	1 = Trigger execution of PreTare	BOOL
Cmd_StartExtAdjust	1 = Start adjustment with external weight	BOOL
Cmd_StartIntAdjust	1 = Start adjustment with internal weight	BOOL
Cmd_StartSpanAdjust	1 = To begin span adjustment process	BOOL
Cmd_StartZeroAdjust	1 = To begin zero adjustment process	BOOL
Cmd_TareImmediate	1 = Trigger execution of Immediate Tare	BOOL
Cmd_TareStable	1 = Trigger execution of Stable Tare	BOOL
Cmd_ZeroImmediate	1 = Trigger execution of Immediate Zero	BOOL
Cmd_ZeroStable	1 = Trigger execution of Stable Zero	BOOL
Set_ExtAdjustWeight	Set external adjustment weight and start adjust	REAL
Set_PreTareValue	Setpoint of PreTare Value	REAL
Set_SpanHiWeight	Highest adjustment weight used in adjustment process. Used with all linearity ranges	REAL
Set_SpanLinearityRange	0 = 2-point adjustment; 1 = 3-point adjustment; 2 = 4-point adjustment; 3 = 5-point adjustment	INT
Set_SpanLowWeight	Low weight used in adjustment process. Used with four and five point linearity ranges	REAL

Input	Function/Description	DataType
Set_SpanMidWeight	Middle weight used in adjustment process. Used with three point, four point and five point linearity ranges	REAL
Set_SpanxLowWeight	xLow weight used in adjustment process. Used with five point linearity range	REAL
EnableIn	Enable Input - System Defined Parameter	BOOL

Output Data

Output	Function/Description	DataType
Sts_Connected	Device is connected to the Programmable Controller	BOOL
Sts_Available	Device is available for interaction with user code	BOOL
Sts_Warning	Device warning status: 1 = an active alarm or warning exists	BOOL
Sts_Faulted	Device faulted status: 1 = an active fault exists	BOOL
Sts_Ready	Device ready status: 1 = ready to activate power structure	BOOL
Sts_Active	Device active status: 1 = output power structure is active	BOOL
Sts_AnalogSensorFailure	Analog Sensor Error detected by measuring system total resistance	BOOL
Sts_bNotReady	Bitwise device 'not ready' reason 0 = Reserved 1 = Device not connected 2 = Device not available 3 = Device Faulted 4 - 31 = Reserved	DINT
Sts_CalibrationError	Weight data can no longer be trusted due to loss of calibration data or an algorithm running in the product to detect weighing irregularities	BOOL
Sts_CenterOfZero	Gross weight value is at a value of zero +/- one quarter of a weight and measures verification interval denoted as "e"	BOOL
Sts_CheckFunction	Imminent failure according to Smart5 definition	BOOL
Sts_ChecksumFailure	A checksum analysis of memory does not yield the expected result	BOOL
Sts_CommunicationOk	Communication Between Controller & IND360 working OK	BOOL
Sts_CustomerDefinedOverload	Weight is equal to or greater than the maximum load allowed	BOOL
Sts_CustomerDefinedUnderload	Weight is equal or less than the minimum load allowed	BOOL
Sts_DataOkay	0 = Data Not Ok ; 1 = Data Ok	BOOL
Sts_ExtAdjustDone	0=Another external adjustment begins ; 1=Adjustment completes successfully	BOOL
Sts_ExtAdjustError	Error occurred and adjustment could not complete	BOOL
Sts_ExtInProcess	External adjustment is in process	BOOL
Sts_ExtLoadWeight	User needs to load test weight corresponding to the value of External Weight	BOOL
Sts_ExtUnloadWeight	User needs to remove all weight from the scale	BOOL
Sts_Failure	Application fault; predictive diagnostics alarm triggered or command cannot be executed as requested	BOOL
Sts_FastGrossWtNA	Fast Gross Weight Parameter is Not Available for this Device	BOOL
Sts_FastNetWtNA	Fast Net Weight Parameter is Not Available for this Device	BOOL
Sts_GrossWtLCNA	Gross Weight for Each Load Cell Parameter is Not Available for this Device	BOOL
Sts_InhibitCfg	Disable Configuration inputs from external sources	BOOL
Sts_InhibitCmd	Disable Command inputs from external sources	BOOL

Output	Function/Description	Data Type
Sts_InhibitSet	Disable Setting inputs from external sources	BOOL
Sts_InSetup	Sensor is in setup mode	BOOL
Sts_IntAdjustDone	Internal adjustment successfully completed	BOOL
Sts_IntAdjustError	Error occurred and adjustment could not complete	BOOL
Sts_LFTSwitchEnabled	Weights and measures switch is enabled for transactional weighing	BOOL
Sts_MaintenanceRequired	Predictive Alarm according to Smart5 definition	BOOL
Sts_Motion	Weight is unstable	BOOL
Sts_NetMode	0 = Gross Weight ; 1 = Net Weight	BOOL
Sts_NetworkFailure	Failure of the entire network. No cells are responding	BOOL
Sts_NumberOfLCNA	Number Of Load Cell Parameter is Not Available for this Device	BOOL
Sts_OutOfADRangeOverUnder	Weight data can no longer be trusted due to loss of data or mechanical damage of the weigh module	BOOL
Sts_OutOfSpecification	Out of specification according to Smart5 definition	BOOL
Sts_PeakWtNA	Peak Weight Parameter is Not Available for this Device	BOOL
Sts_PowerUpZeroFailure	Scale has not been able to complete its power-up restore / reset of zero	BOOL
Sts_RawCountNA	Raw Count Parameter is Not Available for this Device	BOOL
Sts_RawLCCCountNA	Raw Load Cell Count Parameter is Not Available for this Device	BOOL
Sts_RoundedRateNA	Rounded Rate Parameter is Not Available for this Device	BOOL
Sts_SingleSensorCommFailure	One or more of the connected sensors are not working properly	BOOL
Sts_SpanAdjustDone	Span adjustment successfully completed	BOOL
Sts_SpanAdjustError	Error occurred and adjustment could not complete	BOOL
Sts_SpanAdjustLoadWt	To load test weight corresponding to the value Val_SpanAdjustCurrentWt output	BOOL
Sts_TestMode	Device is in a mode in which live data is being replaced with special test data	BOOL
Sts_WeightBlocked	Weight data does not change appreciably over a defined period of time	BOOL
Sts_ZeroAdjustDone	Zero adjustment successfully completed	BOOL
Sts_ZeroAdjustError	Error occurred and adjustment could not complete	BOOL
Sts_ZeroOutOfRange	A control system attempted a zero command but the device did not accept the command because the weight is outside the specified limits or the weights and measure limits	BOOL
Val_AdjustStep	Adjustment Steps	INT
Val_CurrentUnit	Current Weight Unit : 0 = g; 1 = kg; 2 = lb; 3 = t; 4 = ton; 5 = lboz; 6 = otz; 7 = dwt; 8 = oz; 9 = mg; 10 = ug; 11 = cus	SINT
Val_ExtCurrentWeight	Get currently requested external calibration weight during ongoing adjustment or calibration procedure	REAL

Output	Function/Description	Data Type
Val_FastGrossWeight	Fast Gross Weight	REAL
Val_FastNetWeight	Fast Net Weight	REAL
Val_GrossWeight	Gross Weight	REAL
Val_GrossWeightLC1	Gross Weight For Load Cell 1(POWERCELL)	REAL
Val_GrossWeightLC2	Gross Weight For Load Cell 2(POWERCELL)	REAL
Val_GrossWeightLC3	Gross Weight For Load Cell 3(POWERCELL)	REAL
Val_GrossWeightLC4	Gross Weight For Load Cell 4(POWERCELL)	REAL
Val_GrossWeightLC5	Gross Weight For Load Cell 5(POWERCELL)	REAL
Val_GrossWeightLC6	Gross Weight For Load Cell 6(POWERCELL)	REAL
Val_GrossWeightLC7	Gross Weight For Load Cell 7(POWERCELL)	REAL
Val_GrossWeightLC8	Gross Weight For Load Cell 8(POWERCELL)	REAL
Val_GrossWeightLC9	Gross Weight For Load Cell 8(POWERCELL)	REAL
Val_GrossWeightLC10	Gross Weight For Load Cell 10 (POWERCELL)	REAL
Val_GrossWeightLC11	Gross Weight For Load Cell 11 (POWERCELL)	REAL
Val_GrossWeightLC12	Gross Weight For Load Cell 12 (POWERCELL)	REAL
Val_GrossWeightLC13	Gross Weight For Load Cell 13 (POWERCELL)	REAL
Val_GrossWeightLC14	Gross Weight For Load Cell 14 (POWERCELL)	REAL
Val_GrossWeightLC15	Gross Weight For Load Cell 15 (POWERCELL)	REAL
Val_MaxWeightCapacity	Maximal Scale/Sensor Capacity	REAL
Val_MaxWeightMarginCapacity	Maximal Scale/Sensor Capacity With 30%Margin	REAL
Val_MinWeightCapacity	Minimal Scale/Sensor Capacity	REAL
Val_NetWeight	Net Weight	REAL
Val_NoOfDigitsDP	No. Of Digits & Decimal Points : 0.0001 = 1; 0.001 = 2; 0.01 = 3; 0.1 = 4; 1 = 5; 2 = 6; 5 = 7; 10 = 8	SINT
Val_NumberOfLC	Number Of Load Cell	INT
Val_PeakWeight	Peak Weight	REAL
Val_RawCounts	Raw Counts	DINT
Val_RawLCCounts	Raw Load Cell Counts	DINT
Val_RealTimeWeight	Unstable Real Time Weight	REAL
Val_RoundedGrossWeight	Rounded Gross Weight	REAL
Val_RoundedNetWeight	Rounded Net Weight	REAL
Val_RoundedRate	Rounded Rate	REAL
Val_RoundedTareWeight	Rounded Tare Weight Value	REAL
Val_SpanAdjustCurrentPt	Current adjustment point being calibrated	DINT
Val_SpanAdjustCurrentWeight	The value of the test weight that should be placed on the scale	REAL

Output	Function/Description	Data Type
Val_StableWeight	Stable Real Time Weight	REAL
Val_StatusAdjust	0 = Adjustment Not Started, 2047 = Adjustment In Process, 2046 = Adjustment completed successfully, 2045 = Adjustment Not Started, 32768 = Error or Timeout	INT
Val_TareWeight	Tare Weight	REAL
raC_Dvc_ADFramework_DV_LD	Unique Parameter Name for auto - discovery	BOOL
EnableOut	Enable Output - System Defined Parameter	BOOL

Data Types

The following Mettler Toledo Common Control Interface tags are the primary device program tags to read and write to when interfacing to Mettler Toledo devices. The value of using these tags in your specific application code is that you may use a number of different Mettler Toledo devices such as IND360 without having to update your application device interface tags.

Refer to the [Interfaces](#) section for detailed information on interfaces.



For further information and examples on how to interface the Mettler Toledo device objects with your specific application code refer to the "How_To_Import_and_Configure_MettlerToledoDevice_Objects_in_LogixD esigner.mp4" video within the Videos folder of the Mettler Toledo Device Library Download files.

raC_UDT_ItfAD_MettlerToledo_Set

This is the Mettler Toledo Common Control Interface User-Defined Data Type for device settings. Its members provide application program access to allow or inhibit commands and settings from the device faceplate or other external sources. The table below shows member names, descriptions, and tag data types.

For example, to inhibit write commands from the device faceplate or other external sources write a 1 to the _InstanceName_CtrlSet.InhibitCmd program tag from your application program. This would prevent a Clear Tare command from the device faceplate. You may also set the Pre-Tare Value for the device.

Member	Description	Data Type
InhibitCmd	1 = Inhibit user Commands from external sources, 0 = Allow.	BOOL
InhibitSet	1 = Inhibit user Settings from external sources, 0 = Allow.	BOOL
InhibitCfg	1 = Inhibit user Configuration from external sources, 0 = Allow.	BOOL
PreTareValue	Setpoint of PreTare Value.	REAL

raC_UDT_ItfAD_MettlerToledo_Cmd

This is the Mettler Toledo Common Control Interface User-Defined Data Type for device commands. Its members provide application program access to common device commands.

Only write to these common command members to control the device. If you write directly to the device's output command tags directly unexpected device operation could occur.

For example, to tare the weight write a 1 to the `_InstanceName_CtrlCmd.TareImmediate`. Although, you can write to the uncommon command tags in the device's output tag if a specific common control interface tag does not exist.

The table below shows member names, descriptions, and tag data types.

Member	Description	Data Type
bCmd	Commands (Bit Overlay).	INT
ResetWarn	1 = Reset device warning [No warning reset].	BOOL
ResetFault	1 = Reset device trip or fault [No Fault reset,- Automatic fault reset only].	BOOL
Physical	1 = Operate as Physical Device - hold for future use.	BOOL
Virtual	Virtual mode not implemented - hold for future use.	BOOL
TareStable	1 = Trigger execution of Stable Tare.	BOOL
ZeroStable	1 = Trigger execution of Stable Zero.	BOOL
TareImmediate	1 = Trigger execution of Immediate Tare.	BOOL
ZeroImmediate	1 = Trigger execution of Immediate Zero.	BOOL
ClearTare	1 = Trigger execution of Clear Tare.	BOOL
PreTare	1 = Trigger execution of PreTare.	BOOL

raC_UDT_ItfAD_MettlerToledo_Sts

This is the Mettler Toledo 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
eState	Enumerated state value: 0 = Unused, 1 = Initializing, 2 = Disconnected, 3 = Disconnecting, 4 = Connecting, 5 = Idle, 6 = Configuring, 7 = Available.	DINT
FirstWarning	First Warning Event Data.	raC_UDT_Event
FirstFault	First Fault Event Data.	raC_UDT_Event
eCmdFail	Enumerated command failure code. See extended help for enumeration values.	DINT
bSts	Status (Bit Overlay).	DINT
Connected	1 = PAC to device connection has been established.	BOOL
Available	1 = The device is available for interaction with the user program.	BOOL

Input	Description	Data Type
Warning	1 = A warning is active on the device.	BOOL
Faulted	1 = A fault is active on the device.	BOOL
Physical	1 = Controlling physical device.	BOOL
Virtual	1 = Controlling virtual device.	BOOL
RealTimeWeight	Unstable Real Time Weight	REAL
StableWeight	Stable Real Time Weight	REAL
TareValue	Tare Weight Value	REAL
EU	Current Weight Unit :0 = g; 1 = kg; 2 = lb; 3 = t; 4 = ton; 5 = lboz; 6 = otz; 7 = dwt; 8 = oz; 9 = mg; 10 = ug; 11 = cus	SINT

raC_UDT_Event

An array of size 4 is to be used to log the FirstWarning and FirstFault capture. The data should be FIFO order. The same should be displayed on the Faceplate.

Member	Description	Data Type
Type	Event type: 1 = Status, 2 = Warning, 3 = Fault, 4...n = User.	DINT
ID	User definable event ID.	DINT
Category	User definable category (Electrical,Mechanical,Materials,Utility,etc.).	DINT
Action	User definable event action code.	DINT
Value	User definable event value or fault code.	DINT
Message	Event message text.	STRING
EventTime_L	Timestamp (Date/Time format).	LINT
EventTime_D	Timestamp (Y,M,D,h,m,s,us).	DINT[7]

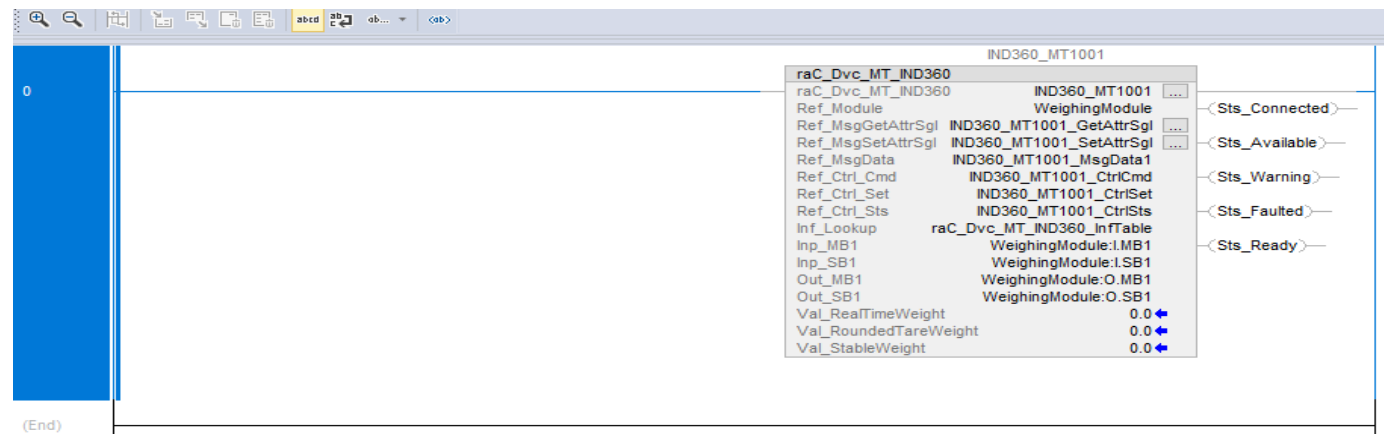
raC_UDT_LookupMember_STR0082

Member	Description	Data Type
Code	Code	DINT
Desc	Code Description	STRING

Programming Example

Fully configured device on a rung is provided below for reference. This example includes the device objects for a IND360 (raC_Dvc_MT_IND360).

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.



The device (ie: IND360) must also be configured with the correct device definition. Note that this configuration is completed automatically when using Application Code Manager or the Studio 5000® Import Library Objects wizard plug-in. For details on setting up the device, refer to the [Device Definition](#) section.

Graphic Symbols

Graphic Symbols are used as launch buttons within HMI applications to open up faceplate displays. Alternatively, faceplates may also be launch from related instructions such as the navigate to device faceplate buttons in the Process Library or the Machine Builder Library faceplates.

All icons display the following information:

- Device label (Tag.@Description or custom label entered in parameter #104)
- Device Warning/Fault Indication
- Device not ready indication

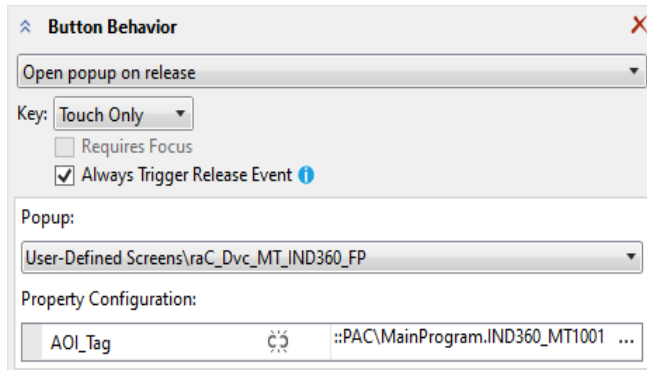
See [Launch Buttons](#) for more general information on launch button diagnostics and usage.


FactoryTalk® View ME/SE Graphic Symbols

Graphic Symbol Name	Graphic Symbol	Description	Global Object Parameter Values
GO_LaunchFP		Faceplate navigation button with string tag label. This launch button graphic object allows the user to navigate to the device object faceplate.	#102: Backing Tag (e.g. [PAC]Program::Program_InstanceName)) #104: Custom button label. Leave blank to use Tag.@Description #120: Display's left position (e.g. 100, optional) #121: Display's top position (e.g. 100, optional)

Studio 5000 View Designer® Graphic Symbols

All Studio 5000 View Designer® graphic symbols must be configured with an *Event* to open up the appropriate Popup screen. Select the graphic symbol and in the *Properties* window navigate to the *Events* tab. Assign a *Button Behavior* event to *Open popup on release*. Assign the required Popup screen (e.g. User-Defined Screens\raC_Dvc_MT_IND360_FP). The required *Property Configurations* are found in the following table where you may assign the *AOI_Tag* to the object's Add-On Instruction tag.



Graphic Symbol Name	Graphic Symbol	Description	Property Configuration
Launch		Faceplate navigation button with string tag label. Use Properties > General > Text to modify the button label text.	AOI_Tag: Object's Add-On Instruction Tag

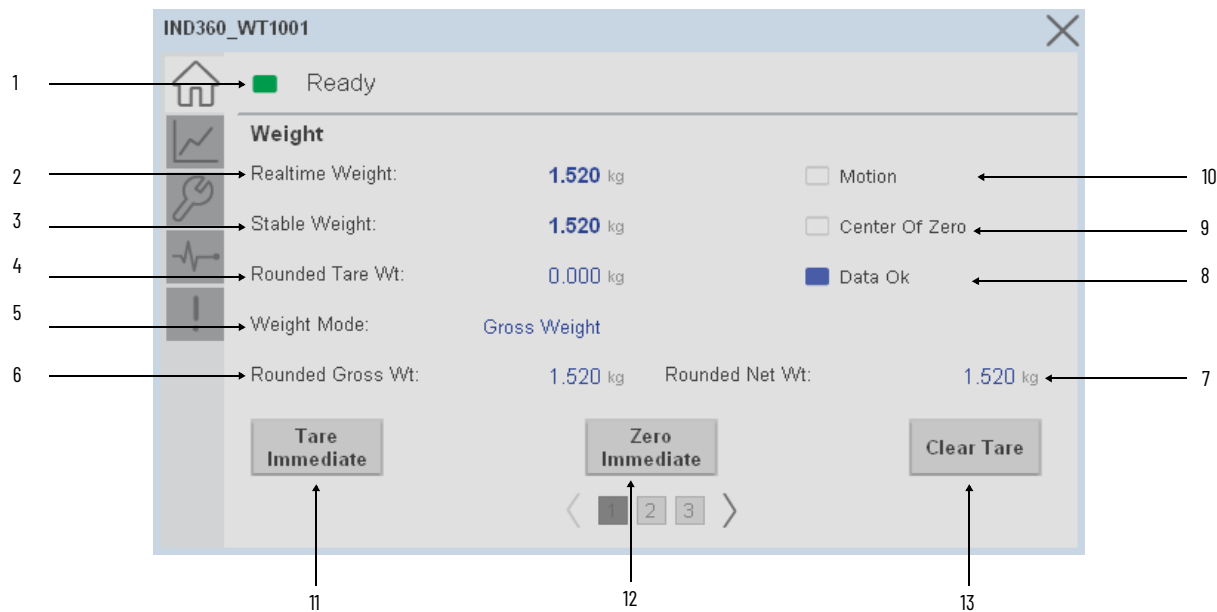
Faceplates

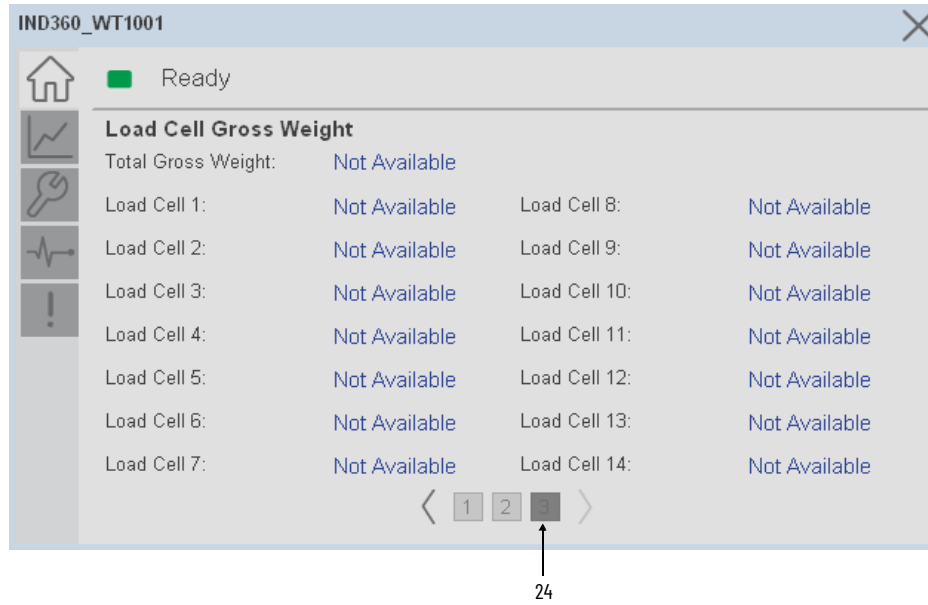
There are basic faceplate attributes that are common across all instructions. See [Basic Faceplate Attributes on page 16](#).

The faceplate title is linked to `_InstanceName.@description`, the `._@description` extended tag property of the Add-On Instruction instance. This is user configurable from controller/program tags in Studio 5000 Logix Designer.

Home Tab

The Home tab is the main tab of the faceplate. It contains Primary weight parameters as well weighing terminal parameters of the device, Device status information and primary commands of the device.

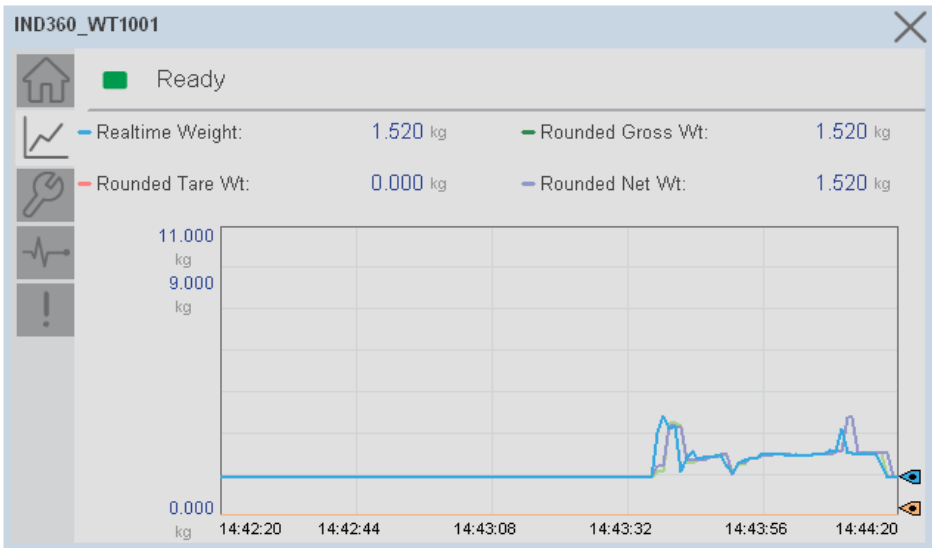




Item	Description
1	Banner
2	Unstable Real Time Weight
3	Stable Real Time Weight
4	Rounded Tare Weight Value
5	Weight Mode :- 0 = Gross Weight ; 1 = Net Weight
6	Rounded Gross Weight
7	Rounded Net Weight
8	0 = Data Not Ok ; 1 = Data Ok
9	The "Center of zero" status indicates that the gross weight value is at zero, within a tolerance of 25 percent
10	The "Motion" status is "active" when the weight value is not stable. The "Zero stable" and "Tare stable" commands become "active" when the Motion status is "inactive"
11	The "Tare Immediate button" is pressed to initiate the "tare command" regardless of the stability of the weight value
12	The "Zero Immediate button" is pressed to execute the "zero command" regardless of the stability of the weight value. This feature is specifically designed for making minor adjustments to the "zero point" to compensate for drifting
13	The "Clear Tare button" to clear the current tare value
14	Gross Weight Value
15	Tare Weight Value
16	Net Weight Value
17	Rounded Rate
18	Peak Weight Value
19	Fast Gross Weight
20	Fast Net Weight
21	Raw Counts
22	Raw Load Cell Counts
23	Number Of Load Cell
24	This page includes all Load Cells Gross weight and applicable for POWERCELL Device type.

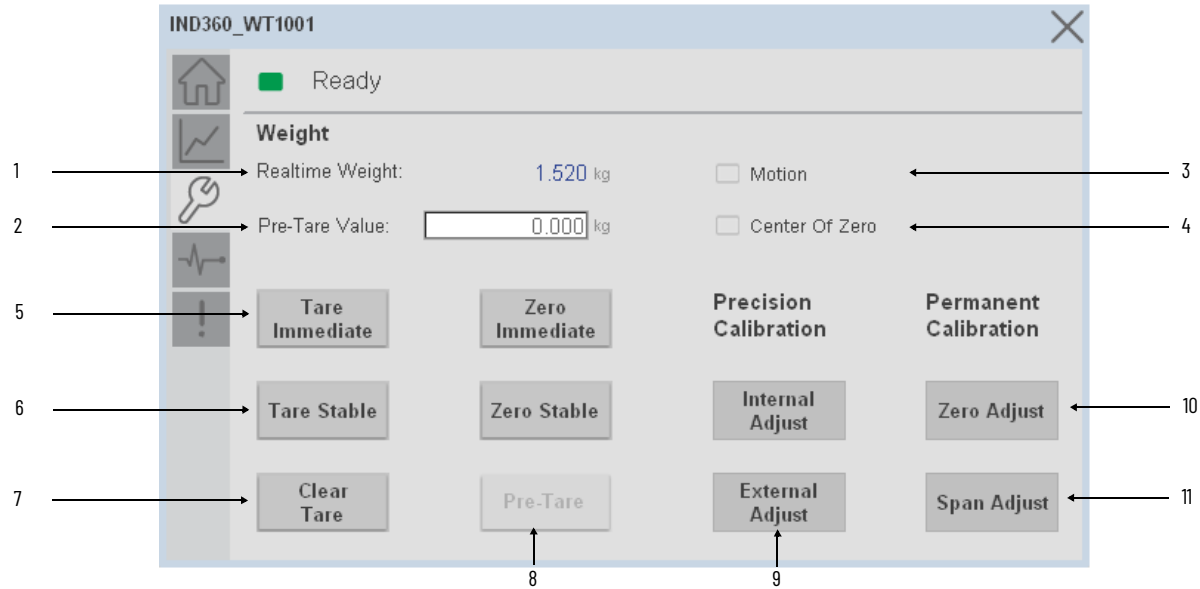
Trend Tab

Trends display values over time, often used to compare similar or related values and to allow operators to predict future states to make control action decisions. There are total four trends are displayed as follows Realtime Weight, Rounded Gross Weight, Rounded Net Weight and Rounded Tare Weight.



Configure Tab

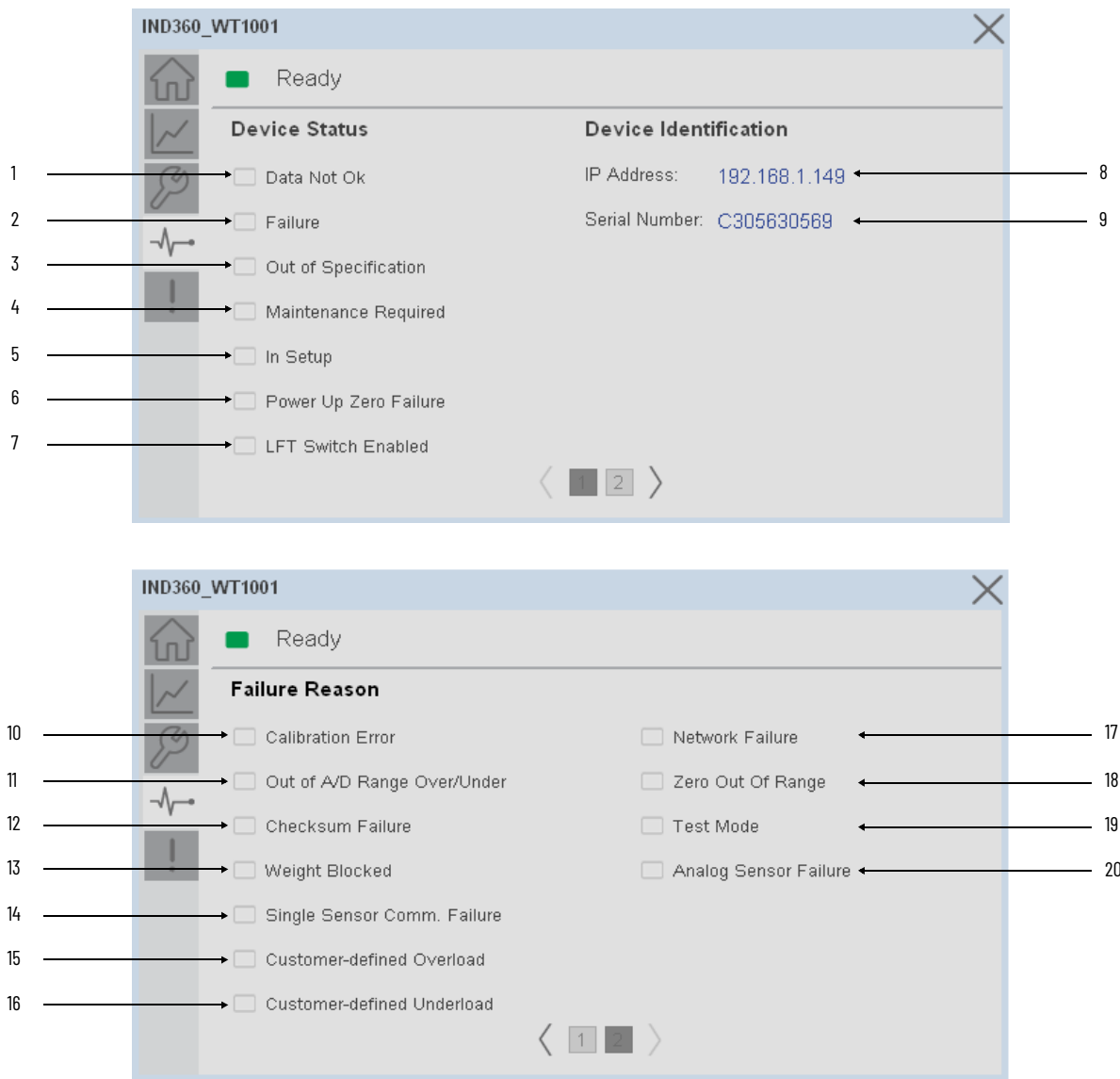
The Configure tab contains configuration elements that a maintenance technician would need to troubleshoot and adjust for an object on another tab. Configuration tab includes, Real time weight, Pre-Tare Value, Tare Commands, Zero Commands, Device Status as well as navigation buttons for Calibration when using precision type device.



Item	Description
1	Unstable Real Time Weight
2	Setpoint of PreTare Value
3	The "Motion" status is "active" when the weight value is not stable. The "Zero stable" and "Tare stable" commands become "active" when the Motion status is "inactive"
4	The "Center of zero" status indicates that the gross weight value is at zero, within a tolerance of 25 percent
5	The "Tare Immediate button" is pressed to initiate the "tare command" regardless of the stability of the weight value and The "Zero Immediate button" is pressed to execute the "zero command" regardless of the stability of the weight value. This feature is specifically designed for making minor adjustments to the "zero point" to compensate for drifting
6	The "Tare stable button" initiates the tare command to the "IND360 device" when the weight is stable and the "Zero stable button" initiates the zero command to "IND360 device" when the weight is stable
7	The "Clear tare" button to clear the current tare value
8	The "Pre-tare value" allows for the configuration of a pre-set tare value. After setting the "Pre-tare value", the user needs to press the "Pre-tare button" to apply the configured value to the "IND360 device"
9	Calibration parameter windows are applicable only when the device model is of the "precision type"
10	Navigation button for Zero Adjustment calibration parameters window
11	Navigation button for Span Adjustment calibration parameters window

Diagnostics Tab

The device diagnostics tab includes a list of information available in the device for troubleshooting. Diagnostics tab includes Device status, Device Identification and Failure Reason.



Item	Description
1	The "Data Ok" status is inactive when the device is still operational but the value being reported cannot be guaranteed to be valid
2	Application fault; predictive diagnostics alarm triggered or command cannot be executed as requested
3	Out of specification according to Smart5 definition
4	Predictive Alarm according to Smart5 definition
5	Sensor is in setup mode
6	Scale has not been able to complete its power-up restore / reset of zero
7	Weights and measures switch is enabled for transactional weighing

Item	Description
8	Device IP Address
9	Device Serial Number
10	Weight data can no longer be trusted due to loss of calibration data or an algorithm running in the product to detect weighing irregularities
11	Weight data can no longer be trusted due to loss of data or mechanical damage of the weigh module
12	A checksum analysis of memory does not yield the expected result
13	Weight data does not change appreciably over a defined period of time
14	One or more of the connected sensors are not working properly
15	Weight is equal to or greater than the maximum load allowed
16	Weight is equal or less than the minimum load allowed
17	Failure of the entire network. No cells are responding
18	A control system attempted a zero command but the device did not accept the command because the weight is outside the specified limits or the weights and measure limits
19	Device is in a mode in which live data is being replaced with special test data
20	Analog Sensor Error detected by measuring system total resistance

Fault Warning Tab

The Fault Warning tab displays information for up to four faults for the device. The fault table displays the Severity level (Fault, Warning or Active Fault), time (and date) and a description of the fault.

Note, only row 1 will display the “Active Fault” in the severity column if there is a current active fault, else it will display the last fault. Rows 2-4 only display past faults and warnings, not an active fault.

IND360_WT1001

1 → [Yellow Square] Not Ready Communication Loss

Severity	Time	Description
Active Fault	2023-12-20 09:40:39	Device Communication Loss
Warning	2023-12-20 09:27:12	Data Not Okay
Fault	2023-12-20 09:26:54	Device Communication Loss
Fault	2023-12-20 09:26:20	Device Communication Loss

2 → [Active Fault Row]

3 → [Warning Icon]

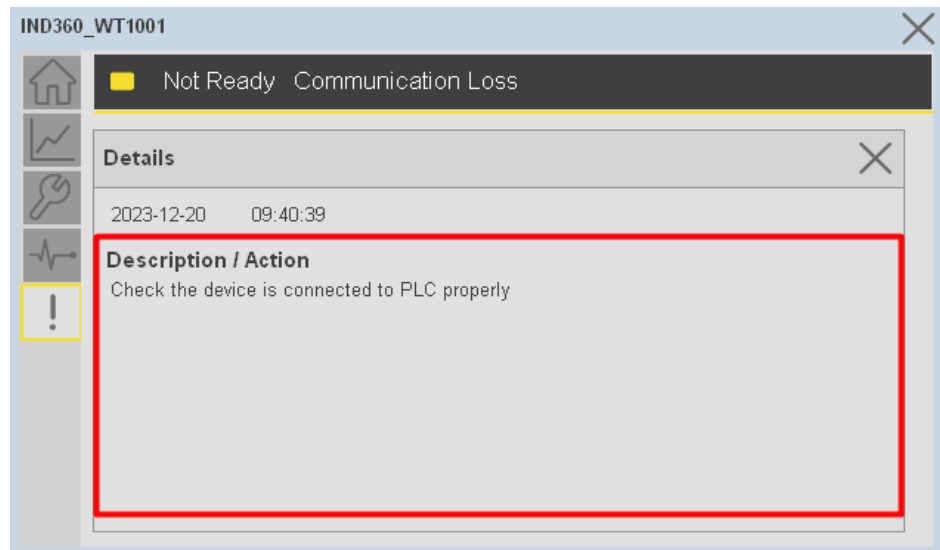
4 → [Severity Column]

5 → [Time Column]

6 → [Description Column]

Item	Description
1	Banner
2	Last fault is in first row and show in bold if active
3	Yellow border visible when a fault is active
4	Fault severity
5	Fault event time
6	4 most recent fault/warning event messages

Click on any row in the fault table to view fault details. The details window provides a more detailed description and possible action steps to remedy condition.



Application Code Manager

All Mettler Toledo device objects have similar configuration parameters in Application Code Manager. The following section defines the common parameters.

Refer to the section [Using Application Code Manager](#) for complete details.

Definition Object: raC_Dvc_MT_IND360

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_MT_IND360

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
ModuleName	Mod_{ObjectName}	{ModuleName}	Input Parameter	Enter the Module Name. This is the name for the module that appears in the Controller Organizer tree.
IncludeHW	1			Allow ACM to create the Hardware Module. If the module already exists in the Controller Organizer, select False or existing module properties will be overwritten.
IPAddress	192.168.1.0		Input Parameter	Enter a valid network address for the hardware module. It must be of form X.X.X.X
FieldBusFormat	2-Block		Input Parameter	Select FieldBus Format selected on device
RPI	2.0		Input Parameter	This is the Requested Packet Interval (RPI) of the module (1.0ms - 100ms).
ParentModule	Local		Input Parameter	Select the Parent Module. This represents the name of the communication adapter this module will communicate through. If connecting to a non-library object module, enter the name of the module only. If the module is connected directly to the controller ethernet port, enter "Local". Note: entering non-library object modules will result in the parameter displaying a red X. This will still generate properly as long as the entered name exists in the project.
ChassisName	{ParentModule}			Warning removal

Linked Libraries

Link Name	Catalog Number	Revision	Solution	Category
raC_Dvc_MT_IND360	raC_Dvc_MT_IND360	1.0	(RA-LIB) Device	MettlerToledo

Configured HMI Content

HMI Content	Instance Name	Description
Launch Button SE	{ObjectName}_G0_LaunchFP	Global Object configured callout instance
Launch Button ME	{ObjectName}_G0_LaunchFP	Global Object configured callout instance

Attachments

Name	Description	File Name	Extraction Path
VL_raC_Dvc_Global	Graphic Symbols SE	(raC-1-SE) Graphic Symbols - MettlerToledo Device.ggfx	{ProjectName}\Visualization\FTViewSE\GlobalObjects
VL_raC_Dvc_Global	Graphic Symbols ME	(raC-1-ME) Graphic Symbols - MettlerToledo Device.ggfx	{ProjectName}\Visualization\FTViewME\GlobalObjects
VL_raC_Dvc_Global	Toolbox SE	(raC-1-SE) Toolbox - MettlerToledo Device.ggfx	{ProjectName}\Visualization\FTViewSE\GlobalObjects
VL_raC_Dvc_Global	Toolbox ME	(raC-1-ME) Toolbox - MettlerToledo Device.ggfx	{ProjectName}\Visualization\FTViewME\GlobalObjects
VL_raC_Dvc_MT_IND360	Faceplate SE	(raC-1_00-SE) raC_Dvc_MT_IND360-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
VL_raC_Dvc_MT_IND360	Faceplate ME	(raC-1_00-ME) raC_Dvc_MT_IND360-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
VL_raC_Dvc_MettlerToledo	View Designer	(raC-1_00-VD) raC_Dvc_MettlerToledo.vpd	{ProjectName}\Visualization\ViewDesigner
VL_MettlerToledo_Manual	Reference Manual	DEVICE-RM905A-EN-P.pdf	{ProjectName}\Documentation
VL_MettlerToledo_Images	HMI Image Set	MettlerToledo_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.





Allen-Bradley, ArmorStart, CompactLogix, ControlLogix, FactoryTalk, Integrated Architecture, iTRAK, Kinetix, Logix 5000, MagneMotion, PlantPAx, PowerFlex, RSLinx, RSLogix, RSLogix 5000, SoftLogix, Stratix, Studio 5000, Studio 5000 Logix Designer, TechConnect, Rockwell Automation, and Rockwell Software are trademarks of Rockwell Automation, Inc.

EtherNet/IP is a trademark of ODVA, Inc.

Microsoft and Windows are trademarks of Microsoft.

Trademarks not belonging to Rockwell Automation are property of their respective companies.

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-RM905A-EN-P - December 2023

Supersedes Publication DEVICE-RM905A-EN-P - December 2023

Copyright © 2023 Rockwell Automation, Inc. All rights reserved. Printed in the U.S.A