Reference Manual

PlantPAx Distributed Control System

Rockwell Automation Library of Process Objects: Two-speed Motor (P_Motor2Spd)

Version 3.5

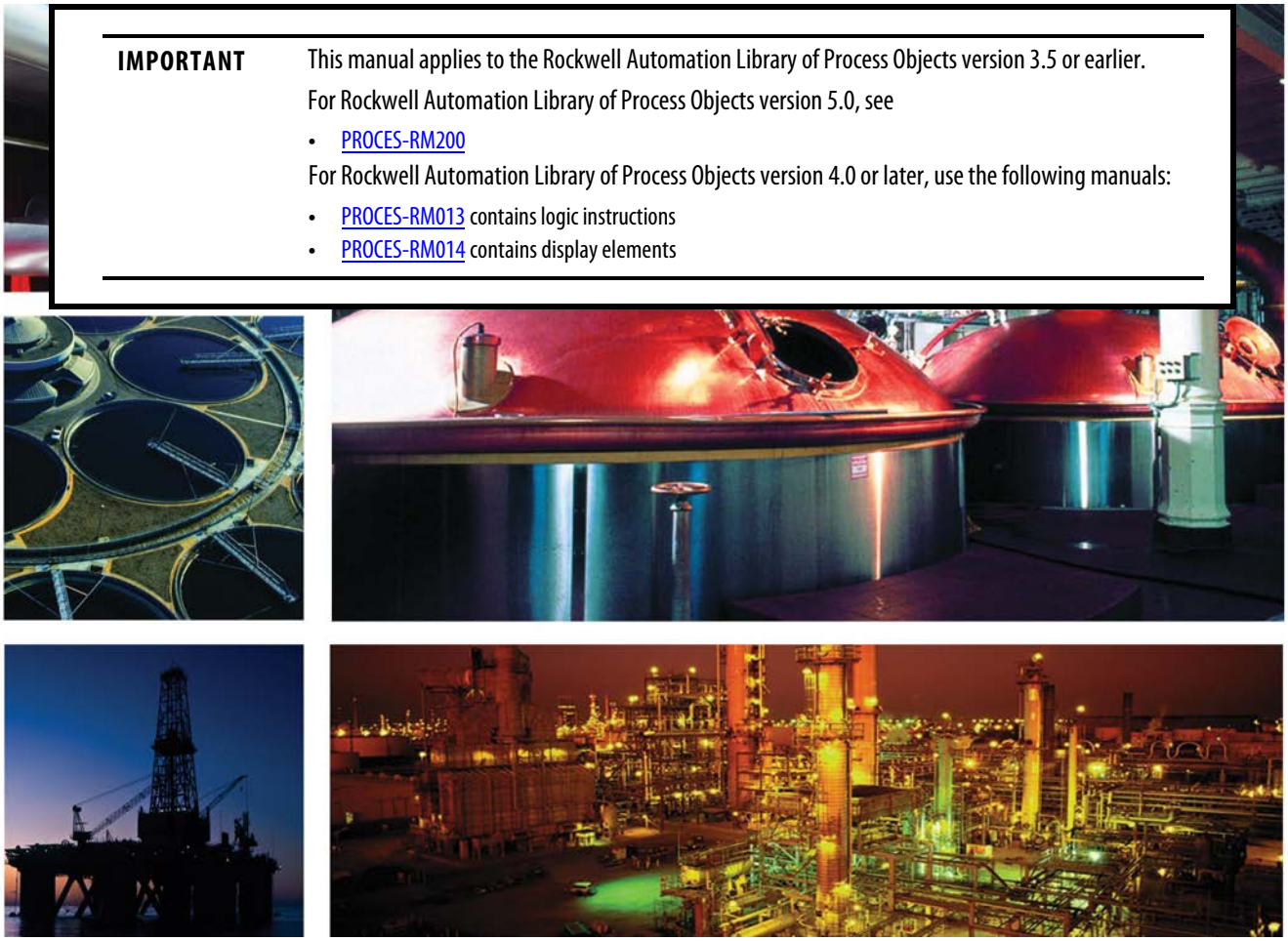
IMPORTANT This manual applies to the Rockwell Automation Library of Process Objects version 3.5 or earlier.

For Rockwell Automation Library of Process Objects version 5.0, see

- PROCES-RM200

For Rockwell Automation Library of Process Objects version 4.0 or later, use the following manuals:

- PROCES-RM013 contains logic instructions
- PROCES-RM014 contains display elements



Allen-Bradley • Rockwell Software

Rockwell Automation

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.

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).

Allen-Bradley, Rockwell Software, Rockwell Automation, RSLogix, Logix5000, FactoryTalk, PlantPAx, and ControlLogix are trademarks of Rockwell Automation, Inc.

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

Table of Contents

Notes:

Software Compatibility and Content Revision

Table 1 - Summary of Changes

Topic	Page
Visualization Files: Global Objects (.ggfx) - Process Diagnostic Objects	10

For the latest compatible software information and to download the Rockwell Automation® Library of Process Objects, see the Product Compatibility and Download Center at http://www.rockwellautomation.com/rockwellautomation/support/pcdc.page.

For general library considerations, see Rockwell Automation Library of Process Objects, publication PROCES-RM002.

Additional Resources

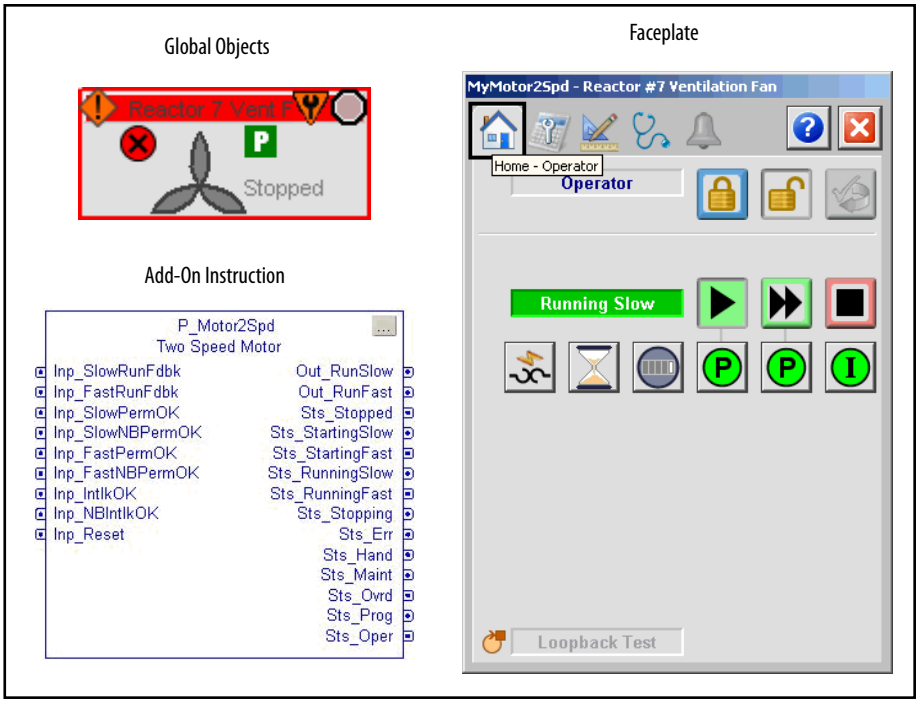
These documents contain additional information concerning related products from Rockwell Automation.

Resource	Description
PlantPAx® Distributed Control System Selection Guide, publication PROCES-SG001	Provides information to assist with equipment procurement for your PlantPAx system.
PlantPAx Distributed Control System Reference Manual, publication PROCES-RM001	Provides characterized recommendations for implementing your PlantPAx system.
Rockwell Automation Library of Process Objects, publication PROCES-RM002	Provides general considerations for the PlantPAx system library of process objects.
FactoryTalk® View Machine Edition User Manual, publication VIEWME-UM004	Provides details on how to use this software package for creating an automation application.
FactoryTalk View Site Edition User Manual, publication VIEWSE-UM006	Provides details on how to use this software package for developing and running human-machine interface (HMI) applications that can involve multiple users and servers, distributed over a network.
Logix5000™ Controllers Add-On Instructions Programming Manual, publication 1756-PM010	Provides information for designing, configuring, and programming Add-On Instructions.
Rockwell Automation Library of Process Objects: Common Alarm Block (P_Alarm) Reference Manual, publication SYSLIB-RM002	Details how to monitor an input condition to raise an alarm. Information includes acknowledging, resetting, inhibiting, and disabling an alarm. Generally the P_Alarm faceplate is accessible from the Alarms tab.
Rockwell Automation Library of Process Objects: Interlocks with First Out and Bypass (P_Intlk) Reference Manual, publication SYSLIB-RM004	Explains how to collect (sum up) the interlock conditions that stop or de-energize a running or energized piece of equipment or prevent it from starting or being energized.
Rockwell Automation Library of Process Objects: Common Mode Block (P_Mode) Reference Manual, publication SYSLIB-RM005	Explains how to choose the Mode (owner) of an instruction or control strategy. The Mode instruction is usually embedded within other instructions to extend their functionality. It is possible to use a standalone Mode instruction to enhance a program where modes are wanted.
Rockwell Automation Library of Process Objects: Permissives with Bypass (P_Perm) Reference Manual, publication SYSLIB-RM007	Details how to collect permissive conditions to start a piece of equipment.
Rockwell Automation Library of Process Objects: Restart Inhibit for Large Motor (P_ResInh) Reference Manual, publication SYSLIB-RM009	Explains how to protect a large motor from damage caused by repeated starts.
Rockwell Automation Library of Process Objects: Run Time and Starts (P_RunTime) Reference Manual, publication SYSLIB-RM010	Explains how to accumulate the total run time and count of starts for a motor or other equipment.
Rockwell Automation Library of Process Objects: E1 Plus™ Overload Relay (P_E1PlusE) Reference Manual, publication SYSLIB-RM049A-EN-P	Details how to control and monitor a 193-ETN (E1 Plus on EtherNet/IP) overload relay.
Rockwell Automation Library of Process Objects: E3™/E3 Plus™ Overload Relay (EtherNet/IP) (P_E3Ovld) Reference Manual, publication SYSLIB-RM050A-EN-P	Details how to control and monitor a 193/592-EC1, -EC2, -EC3, or -EC5 (E3 or E3 Plus) overload relay.
Rockwell Automation Library of Process Objects: E300™ Overload Relay (EtherNet/IP) (P_E300Ovld) Reference Manual, publication SYSLIB-RM051A-EN-P	Details how to controls and monitors a 193-ECM-ETR (E300 on EtherNet/IP) overload relay.

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.

Two-speed Motor (P_Motor2Spd)

The P_Motor2Spd (Two-speed Motor) Add-On Instruction controls a non-reversing, two-speed motor (fast/slow/stopped) in various modes and monitors for fault conditions.



Guidelines

Use this instruction when you need to operate a two-speed motor that runs in one direction. This instruction supports motors that have three controlled states: running fast, running slow, and stopped. The motor can optionally have run feedback that, if available, is used to confirm that the motor is running at the commanded speed, and alarm if not.

Do **not** use this instruction in these situations:

- You are operating a single-speed motor (running/stopped only). Use the P_Motor Single-Speed Motor instruction instead.
- You are operating a continuously-variable speed motor, such as one wired to a variable-frequency AC drive or variable-speed DC drive. Use the P_VSD Variable Speed Drive instruction instead.
- You are operating a simple reversing motor (forward, reverse, and stopped only). Use the P_MotorRev Reversing Motor instruction instead.
- You are monitoring, and optionally tripping, a locally operated (hand-operated) motor. The motor can be single-speed, two-speed, or reversing. Use the P_MotorHO Hand-operated Motor instruction instead.

Functional Description

The P_Motor2Spd instruction provides the following capabilities:

- Controls outputs to start a two-speed motor fast or slow and stop the motor.
- Motors run feedback (optional) and display actual motor status.
- Detects failure to start or stop and generates appropriate alarms.
- Monitors permissive conditions to allow starting. Separate permissives are provided to allow running fast and running slow.
- Monitors interlock conditions to stop the motor or prevent starting, and alarms when an interlock trips the motor.
- Provides alarms for Failure to Start, Failure to Stop, Interlock Trip, and I/O fault.
- Provides for simulation of a working motor while disabling outputs, for use in off-process training, testing, or simulation.
- Monitors I/O communication, and alarms and shuts down on a communication fault.
- Operates in Operator, Program, Override, Maintenance, and Hand modes. (See the following operating modes.)
- Provides an available status for use by automation logic to determine if other program logic can start and stop the motor.

Required 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.

Controller File

The P_Motor2Spd_3_5-**00**_AOI.L5X Add-On Instruction must be imported into the controller project to be used in the controller configuration. The service release number (boldfaced) can change as service revisions are created.

Visualization Files

This Add-On Instruction has associated visualization files that provide a common user interface. These files can be downloaded from the Product Compatibility and Download Center at http://www.rockwellautomation.com/rockwellautomation/support/pcdc.page.

IMPORTANT The visualization file dependencies require Process Library content imports to occur in a specific order as reflected in the following tables:

- Images
- Global Objects
- Standard Displays
- HMI Tags
- Macros

Images are external graphic files that can be used in displays. They must be imported for FactoryTalk View to make use of them.

When PNG files are imported, they are renamed by FactoryTalk View with a .bmp file extension, but retain a .png format.

Table 2 - Visualization Files: Images (.png)

FactoryTalk View SE Software	FactoryTalk View ME Software	Description
All .png files in the images folder	All .png files in the images folder	These are the common icons used in the global objects and standard displays for all Process Objects.

The Global Object files (.ggfx file type) in the following table are Process Library display elements 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.

Table 3 - Visualization Files: Global Objects (.ggfx)

FactoryTalk View SE Software	FactoryTalk View ME Software	Description
(RA-BAS) Common Faceplate Objects	(RA-BAS-ME) Common Faceplate Objects	Global objects used on process object faceplates.

Table 3 - Visualization Files: Global Objects (.ggfx)

FactoryTalk View SE Software	FactoryTalk View ME Software	Description
(RA-BAS) P_Motor2Spd Graphics Library	(RA-BAS-ME) P_Motor2Spd Graphics Library	P_Motor2Spd global object device symbols used to build process graphics.
(RA-BAS) Process Alarm Objects	(RA-BAS-ME) Process Alarm Objects	Global objects used for managing alarms on process object faceplates.
(RA-BAS) Process Diagnostic Objects	(RA-BAS-ME) Process Diagnostic Objects	Diagnostic global objects used on process object faceplates.
(RA-BAS) Process Faceplate Motor Objects	(RA-BAS-ME) Process Faceplate Motor Objects	Motor global objects used on process object faceplates.
(RA-BAS) Process Help Objects	(RA-BAS-ME) Process Help Objects	Global objects used for all process objects help displays.
(RA-BAS) Process Interlock Objects	(RA-BAS-ME) Process Interlock Objects	Global objects used for managing interlocks and permissives on process object faceplates.
(RA-BAS) Process Mode Objects	(RA-BAS-ME) Process Mode Objects	Global objects used for managing modes on process object faceplates.

The Standard Display files (.gfx file type) in the following table are the Process Library displays that you see at runtime.

Table 4 - Visualization Files: Standard Displays (.gfx)

FactoryTalk View SE Software	FactoryTalk View ME Software	Description
(RA-BAS) Common-AnalogEdit	N/A	Faceplate used for analog input data entry. The FactoryTalk View ME faceplates use the native analog input data entry so no file is required.
(RA-BAS) P_Alarm-Faceplate	(RA-BAS-ME) P_Alarm-Faceplate	The faceplate that is used for managing alarms for the object.
(RA-BAS) P_Alarm-Help	(RA-BAS-ME) P_Alarm-Help	Alarm Help information that is accessed from the P_Alarm faceplate.
(RA-BAS) P_Mode-Config	(RA-BAS-ME) P_Mode-Config	The Configuration Display used to configure the P_Mode object.
(RA-BAS) P_Mode-Help	(RA-BAS-ME) P_Mode-Help	Mode Help information that is accessed from the Help faceplate.
(RA-BAS) P_Motor2Spd-Faceplate	(RA-BAS-ME) P_Motor2Spd-Faceplate	The faceplate that is used for the object
(RA-BAS) P_Motor2Spd-Quick	(RA-BAS-ME) P_Motor2Spd-Quick	The Quick display that is used for the object
(RA-BAS) Process Motor Family-Help	(RA-BAS-ME) Process Motor Family-Help	The Help display for Motor objects
(RA-BAS) E1PlusE-Faceplate (RA-BAS) E3Ovld-Faceplate (RA-BAS) E300Ovld-Faceplate	(RA-BAS-ME) E1PlusE-Faceplate (RA-BAS-ME) E3Ovld-Faceplate (RA-BAS-ME) E300Ovld-Faceplate	**Optional** Overload faceplate displays that are used for the object Use one of these files if your object has an associated P_E1PlusE, P_E3Ovld, or P_E300Ovld object and you enable navigation to one of these faceplates from the object faceplate.
(RA-BAS) P_Intlk-Faceplate	(RA-BAS-ME) P_Intlk-Faceplate	**Optional** The interlock faceplate used for the object. Use this file if your Discrete Output has an associated P_Intlk object and you enable navigation to its faceplate from the Discrete Output faceplate.
(RA-BAS) P_Perm-Faceplate	(RA-BAS-ME) P_Perm-Faceplate	**Optional** Permissive faceplate that is used for the object Use this file if your object has an associated P_Perm object and you enable navigation to the P_Perm faceplate from the object faceplate.

Table 4 - Visualization Files: Standard Displays (.gfx)

FactoryTalk View SE Software	FactoryTalk View ME Software	Description
(RA-BAS) P_ResInh-Faceplate	(RA-BAS-ME) P_ResInh-Faceplate	**Optional** Restart/inhibit faceplate display that is used for the object Use this file if your object has an associated P_ResInh object and you enable navigation to the P_ResInh faceplate from the object faceplate.
(RA-BAS) P_RunTime-Faceplate	(RA-BAS-ME) P_RunTime-Faceplate	**Optional** RunTime faceplate display that is used for the object Use this file if your object has an associated P_RunTime object and you enable navigation to the P_RunTime faceplate from the object faceplate.
(RA-BAS) Process Interlock Family-Help	(RA-BAS-ME) Process Interlock Family-Help	**Optional** Interlock/permissives help display that is used for the object Use this file if you use the P_Intlk or P_Perm faceplate.

HMI Tags are created in a FactoryTalk View ME application to support tab switching on Process Library faceplates. The HMI tags may be imported via the comma-separated values file (.csv file type) in the following table.

Table 5 - Visualization Files: HMI Tags (.csv)

FactoryTalk View SE Software	FactoryTalk View ME Software	Description
N/A	FTVME_PlantPAxLib_Tags_3_5_**xx**.csv where **xx** = the service release number.	These tags must be imported into the FactoryTalk View ME project to support switching tabs on any Process Object faceplate.

Controller Code

This section describes the parameter references for this Add-On Instruction.

Two-speed Motor Input Structure

Input parameters include the following:

- Input data elements (Inp_) are typically used to connect field inputs from I/O modules or signals from other objects.
- Configuration data elements (Cfg_) are used to set configurable capabilities and features of the instruction.
- Commands (PCmd_, OCmd_, MCmd_) are used by program logic, operators, and maintenance personnel to request instruction actions.
- Settings (PSet_, OSet_, MSet_) are used by program logic, operators, and maintenance personnel to establish runtime setpoints, thresholds, and so forth. A Setting (without a leading P, O, or M) establishes runtime settings regardless of role or mode.

Table 7 - P_Motor2Spd Input Parameters

Input Parameter	Data Type	Alias For	Default	Description
EnableIn	BOOL		1	**Ladder Diagram:** If the rung-in condition is true, the instruction's Logic routine executes. If the rung-in condition is false, the instruction's EnableInFalse routine executes. **Function Block Diagram:** If true, or not connected, the instruction's Logic routine executes. If the parameter is exposed as a pin and wired, and the pin is false, the instruction's EnableInFalse routine executes. **Structured Text:** No effect. The instruction's Logic routine executes.
Inp_SlowRunFdbk	BOOL		0	Feedback from motor 1 = Running slow or running fast.
Inp_FastRunFdbk				
Inp_SlowPermOK	BOOL		1	1 = Bypassable and Non-bypassable Fast and Slow Permissives are OK, motor can start.
Inp_SlowNBPermOK				
Inp_FastPermOK				
Inp_FastNBPermOK				
Inp_IntlkOK				1 = Bypassable and Non-bypassable Interlocks are OK, motor can run.
Inp_NBIntlkOK				
Inp_IOFault	BOOL		0	Input communication status: 0 = OK 1 = Fail
Inp_Sim	BOOL		0	Simulation input. When set to 1, the instruction keeps the outputs de-energized and simulates a working motor. When set to 0, the instruction controls the motor normally.
Inp_Hand	BOOL	Mode.Inp_Hand	0	1 = Select Hand (hardwired) control strategy.
Inp_Ovrd	BOOL	Mode.Inp_Ovrd	0	1 = Select Override control strategy.
Inp_OvrdCmd	DINT		0	Override mode command: 0 = None 1 = Stop 2 = Run Fast 3 = Run Slow

Table 7 - P_Motor2Spd Input Parameters

Input Parameter	Data Type	Alias For	Default	Description
Inp_Reset	BOOL		0	Input parameter used to programmatically reset alarms. When set to 1, all alarms requiring reset are reset.
Cfg_HasRunFdbk	BOOL		0	This parameter determines whether the run feedback exists and is checked or if the run feedback does not exist and is not used. Use run feedback for failure checking when it is connected and the parameter is 1. When the parameter is 0, the run feedback does not exist and is not checked.
Cfg_UseRunFdbk	BOOL		0	1 = Motor run feedback is used for failure checking. 0 = Motor run feedback is bypassed by Maintenance and not checked for failures.
Cfg_AllowLocal	BOOL		0	1 = Allow local Start/Stop without alarm. 0 = Start/Stop from HMI/program only.
Cfg_HasSlowPermObj Cfg_HasFastPermObj	BOOL		0	1 = Tells the HMI that a permissive object (for example, P_Perm) is connected to Inp_SlowPermOK or Inp_FastPermOK. **IMPORTANT:** The name of the Fast Permissive object in the controller must be this object's name with the suffix '_FastPerm'. For example, if your P_Motor2Spd object has the name 'Motor123', then its Fast Permissive object must be named 'Motor123_FastPerm'. **IMPORTANT:** The name of the Slow Permissive object in the controller must be this object's name with the suffix '_SlowPerm'. For example, if your P_Motor2Spd object has the name 'Motor123', then its Slow Permissive object must be named 'Motor123_SlowPerm'.
Cfg_HasIntlkObj	BOOL		0	1 = Tells the HMI that an interlock object (for example, P_Intlk) is connected to Inp_IntlkOK. **IMPORTANT:** The name of the Interlock object in the controller must be this object's name with the suffix '_Intlk'. For example, if your P_Motor2Spd object has the name 'Motor123', then its Interlock object must be named 'Motor123_Intlk'.
Cfg_HasResInhObj	BOOL		0	1 = Tells the HMI that a restart inhibit (for example, P_ResInh) object is connected. **IMPORTANT:** The name of the Restart Inhibit object in the controller must be this object's name with the suffix '_ResInh'. For example, if your P_Motor2Spd object has the name 'Motor123', then its Restart Inhibit object must be named 'Motor_ResInh'.
Cfg_HasRunTimeObj	BOOL		0	1 = Tells the HMI that a runtime (for example, P_RunTime) object is connected. **IMPORTANT:** The name of the runtime object in the controller must be this object's name with the suffix '_RunTime'. For example, if your P_Motor2Spd object has the name 'Motor123', then its runtime object must be named 'Motor123_RunTime'.
Cfg_HasOvldObj	BOOL		0	1 = Tells the HMI that an overload object is connected. **IMPORTANT:** The name of the Overload object in the controller must be this object's name with the suffix '_Ovld'. For example, if your P_Motor2Spd object has the name 'Motor123', then its Overload object must be named 'Motor123_Ovld'.
Cfg_PCmdClear	BOOL	Mode.Cfg_PCmdClear	1	When this parameter is 1, program commands are cleared once they are acted upon. When set to 0, program commands remain set until cleared by the application program logic. **IMPORTANT:** Clearing this parameter online can cause unintended program command execution.
Cfg_ProgDefault	BOOL	Mode.Cfg_ProgDefault	0	This parameter defines the default mode. When this parameter is 1, the mode defaults to Program if no mode is being requested. When this parameter is 0, the mode defaults to Operator if no mode is being requested. **IMPORTANT:** Changing this parameter online can cause unintended mode changes.
Cfg_OperStopPrio	BOOL		0	1 = OCmd_Stop has priority and is accepted at any time. 0 = OCmd_Stop is accepted only in Operator and Maintenance modes.
Cfg_OCmdResets	BOOL		0	1 = New Operator motor command resets fault. 0 = Reset is required to clear fault.

Table 7 - P_Motor2Spd Input Parameters

Input Parameter	Data Type	Alias For	Default	Description
Cfg_OvrdPermIntlk	BOOL		0	1 = Override mode ignores bypassable permissives/interlock. 0 = Always use permissives/interlock.
Cfg_ShedOnFailToStart	BOOL		1	1 = Stop motor and alarm on Fail to Start or I/O Fault. (requires reset) 0 = Alarm only on Fail to Start or I/O Fault.
Cfg_ShedOnIOFault				
Cfg_HasFailToStartAlm	BOOL	FailToStart.Cfg_Exists	0	These parameters determine whether the corresponding alarm exists and is checked or if the alarm does not exist and is not used. When these parameters are 1, the corresponding alarm exists.
Cfg_HasFailToStopAlm		FailToStop.Cfg_Exists		
Cfg_HasIntlkTripAlm		IntlkTrip.Cfg_Exists		
Cfg_HasIOFaultAlm		IOFault.Cfg_Exists		
Cfg_FailToStartResetReqd	BOOL	FailToStart.Cfg_ResetReqd	0	These parameters determine whether a reset is required to clear the alarm status. When these parameters are 1, the alarm is latched ON when the alarm occurs. After the alarm condition returns to normal, a reset is required to clear the alarm status (for example, OCmd_Reset, Inp_Reset, or IOFault.OCmd_Reset are required to clear Alm_IOFault alarm after the alarm is set and the value returns to normal). When these parameters are 0, no reset is required and the alarm status is cleared when the alarm condition returns to normal. **IMPORTANT:** If the reset clears the alarm, it also acknowledges the alarm.
Cfg_FailToStopResetReqd		FailToStop.Cfg_ResetReqd		
Cfg_IntlkTripResetReqd		IntlkTrip.Cfg_ResetReqd		
Cfg_IOFaultResetReqd		IOFault.Cfg_ResetReqd		
Cfg_FailToStartAckReqd	BOOL	FailToStart.Cfg_AckReqd	1	These parameters determine whether the alarm must be acknowledged. When these parameters are 1, the acknowledge (ack) bit is cleared when the alarm occurs. An acknowledge command (for example, PCmd_FailToStartAck or FailToStart.OCmd_Ack) is required to acknowledge the alarm. When set to 0, the Acknowledge bit is set when an alarm occurs indicating an acknowledged alarm and no acknowledge command is required.
Cfg_FailToStopAckReqd		FailToStop.Cfg_AckReqd	1	
Cfg_IntlkTripAckReqd		IntlkTrip.Cfg_AckReqd	1	
Cfg_IOFaultAckReqd		IOFault.Cfg_AckReqd	1.0	
Cfg_FailToStartSeverity	INT	FailToStart.Cfg_Severity	1000	These parameters determine the severity of each alarm. This drives the color and symbol that are used to indicate alarm status on the faceplate and global object. The following are valid values: 1…250 = Low 251…500 = Medium 501…750 = High 751…1000 = Urgent **IMPORTANT:** For FactoryTalk View software version 7.0, these severity parameters drive only the indication on the global object and faceplate. The Alarms and Events definition severity drives the color and symbol that is used on the alarm banner and alarm summary as well as the value returned by FactoryTalk Alarms and Events display commands.
Cfg_FailToStopSeverity		FailToStop.Cfg_Severity	1000	
Cfg_IntlkTripSeverity		IntlkTrip.Cfg_Severity	500	
Cfg_IOFaultSeverity		IOFault.Cfg_Severity	1000	
Cfg_ChangeSpdT	DINT		3	Delay with contactors open when changing speeds (seconds).
Cfg_SimFdbkT	DINT		2	Delay to echo back of Running/Stopped status when in simulation or when motor does not have or is not using run feedback (seconds).
Cfg_FailToStartT	DINT		10	Time after start to get correct run feedback before fault (seconds).
Cfg_FailToStopT	DINT		10	Time after stop to drop run feedbacks before fault (seconds).
PSet_Owner	DINT		0	Program owner request ID (nonzero) or release (zero).
PCmd_RunFast	BOOL		0	When Cfg_PCmdClear is 1: • Set PCmd_RunFast to 1 to run the motor at fast speed • Set PCmd_RunSlow to 1 to run the motor at slow speed • Set PCmd_Stop to 1 to stop the motor • These parameters are reset automatically When Cfg_PCmdClear is 0: • Set PCmd_RunFast to 1 to Run the motor at fast speed • Set PCmd_RunSlow to 1 to run the motor at the slow speed • Set PCmd_RunFast and PCmd_RunSlow to 0 to stop the motor • PCmd_Stop is not used • These parameters are not reset automatically
PCmd_RunSlow				
PCmd_Stop				

Table 7 - P_Motor2Spd Input Parameters

Input Parameter	Data Type	Alias For	Default	Description
PCmd_Acq	BOOL	Mode.PCmd_Acq	0	When Cfg_PCmdClear is 1: • Set PCmd_Acq to 1 to Acquire • Set PCmd_Rel to 1 to Release • These parameters reset automatically When Cfg_PCmdClear is 0: • Set PCmd_Acq to 1 to Acquire • Set PCmd_Acq to 0 to Release • PCmd_Rel is not used • These parameters do not reset automatically
PCmd_Rel		Mode.PCmd_Rel		
PCmd_Lock	BOOL	Mode.PCmd_Lock	0	When Cfg_PCmdClear is 1: • Set PCmd_Lock to 1 to Lock • Set PCmd_Unlock to 1 to Unlock • These parameters reset automatically When Cfg_PCmdClear is 0: • Set PCmd_Lock to 1 to Lock • Set PCmd_Lock to 0 to Unlock • PCmd_Unlock is not used • These parameters do not reset automatically
PCmd_Unlock		Mode.PCmd_Unlock		
PCmd_Reset	BOOL		0	• Set PCmd_Reset to 1 to reset all alarms requiring reset • This parameter is always reset automatically
PCmd_FailToStartAck	BOOL	FailToStart.PCmd_Ack	0	• Set PCmd_<Alarm>Ack to 1 to Acknowledge alarm • The parameter is reset automatically
PCmd_FailToStopAck		FailToStop.PCmd_Ack		
PCmd_IntlkTripAck		IntlkTrip.PCmd_Ack		
PCmd_IOFaultAck		IOFault.PCmd_Ack		
PCmd_FailToStartSuppress	BOOL	FailToStart.PCmd_Suppress	0	When Cfg_PCmdClear is 1: • Set PCmd_<Alarm>Suppress to 1 to suppress alarm • Set PCmd_<Alarm>Unsuppress to 1 to unsuppress alarm • These parameters reset automatically When Cfg_PCmdClear is 0: • Set PCmd_<Alarm>Suppress to 1 to suppress alarm • Set PCmd_<Alarm>Suppress to 0 to unsuppress alarm • PCmd_<Alarm>Unsuppress is not used • These Parameters do not reset automatically
PCmd_FailToStopSuppress		FailToStop.PCmd_Suppress		
PCmd_IntlkTripSuppress		IntlkTrip.PCmd_Suppress		
PCmd_IOFaultSuppress		IOFault.PCmd_Suppress		
PCmd_FailToStartUnsuppress	BOOL	FailToStart.PCmd_Unsuppress	0	
PCmd_FailToStopUnsuppress		FailToStop.PCmd_Unsuppress		
PCmd_IntlkTripUnsuppress		IntlkTrip.PCmd_Unsuppress		
PCmd_IOFaultUnsuppress		IOFault.PCmd_Unsuppress		
PCmd_FailToStartUnshelve	BOOL	FailToStart.PCmd_Unshelve	0	• Set PCmd_<Alarm>Unshelve to 1 to Unshelve alarm • The parameter is reset automatically
PCmd_FailToStopUnshelve		FailToStop.PCmd_Unshelve		
PCmd_IntlkTripUnshelve		IntlkTrip.PCmd_Unshelve		
PCmd_IOFaultUnshelve		IOFault.PCmd_Unshelve		
OCmd_RunSlow	BOOL		0	Operator command to run motor at slow speed.
OCmd_RunFast	BOOL		0	Operator command to run motor at fast speed.
OCmd_Stop	BOOL		0	Operator command to stop motor.
OCmd_Bypass	BOOL		0	Operator command to bypass all bypassable interlocks and permissives.
OCmd_Check	BOOL		0	Operator command to check (not bypass) all interlocks and permissives.
MCmd_Disable	BOOL		0	Maintenance command to disable motor.
MCmd_Enable	BOOL		0	Maintenance command to enable (allow to run) motor.
MCmd_Acq	BOOL	Mode.MCmd_Acq	0	Maintenance command to acquire ownership (Operator/Program/Override to Maintenance).
MCmd_Rel	BOOL	Mode.MCmd_Rel	0	Maintenance command to release ownership (Maintenance to Operator/Program/Override)

Table 7 - P_Motor2Spd Input Parameters

Input Parameter	Data Type	Alias For	Default	Description
OCmd_AcqLock	BOOL	Mode.OCmd_AcqLock	0	Operator command to Acquire (Program to Operator)/Lock ownership.
OCmd_Unlock	BOOL	Mode.OCmd_UnlockRel	0	Operator command to Unlock/Release (Operator to Program) ownership.
OCmd_Reset	BOOL		0	Operator command to reset all alarms requiring reset.
OCmd_ResetAckAll	BOOL		0	Operator command to acknowledge and reset all alarms and latched shed conditions.

Two-speed Motor Output Structure

Output parameters include the following:

- Output data elements (Out_) are the primary outputs of the instruction, typically used by hardware output modules; however, they can be used by other application logic.
- Value data elements (Val_) are numeric outputs of the instruction for use by the HMI. Values can also be used by other application logic or software packages.
- Source and Quality data elements (SrcQ_) are outputs of the instruction that is used by the HMI to indicate PanelView source and quality.
- Status data elements (Sts_) are bit outputs of the instruction for use by the HMI. Status bits can also be used by other application logic.
- Error data elements (Err_) are outputs of the instruction that indicate a particular configuration error. If any Err_ bit is set, then the Sts_Err configuration error summary status is set and the Invalid Configuration indicator is displayed on the HMI.
- Not Ready data elements (Nrdy_) are bit outputs of the instruction for use by the HMI for displaying the Device Not Ready indicator. These bits can also be used by other application logic.
- Alarm data elements (Alm_) are outputs of the instruction that indicate a particular alarm has occurred.
- Acknowledge data elements (Ack_) are outputs of the instruction that indicate the corresponding alarm has been acknowledged.
- Ready data elements (Rdy_) are bit outputs of the instruction used by the HMI to enable or disable Command buttons and Setting entry fields.

Table 8 - P_Motor2Spd Output Parameters

Output Parameter	Data Type	Alias For	Description
EnableOut	BOOL		Enable Output: The EnableOut signal is not manipulated by this instruction. Its output state always reflects EnableIn Input state.
Out_RunSlow	BOOL		1 = Run motor slow. 0 = Stop motor.
Out_RunFast	BOOL		1 = Run motor fast. 0 = Stop motor.
Out_StartSlow	BOOL		Pulse 1 to start motor at slow speed.

Table 8 - P_Motor2Spd Output Parameters

Output Parameter	Data Type	Alias For	Description
Out_StartFast	BOOL		Pulse 1 to start motor at fast speed.
Out_Stop	BOOL		1 = Stop motor. 0 = Motor is left in current state.
SrcQ_IO	SINT		I/O signal source and quality.
SrcQ			Final motor status source and quality: GOOD 0 = I/O live and confirmed good quality 1 = I/O live and assumed good quality 2 = No feedback configured, assumed good quality TEST 8 = Device simulated 9 = Device loopback simulation 10 = Manually entered value UNCERTAIN 16 = Live input, off-specification 17 = Value substituted at device/bus 18 = Value substituted by maintenance (Has and not Use) 19 = Shed, using last good value 20 = Shed, using replacement value BAD 32 = Signal failure (out-of-range, NaN, invalid combination) 33 = I/O channel fault 34 = I/O module fault 35 = Bad I/O configuration (for example, scaling parameters)
Val_Cmd	SINT		Device command: 0 = None 1 = Stop 3 = Start Fast 4 = Start Slow
Val_Fdbk	SINT		Device feedback: 0 = Stopped 2 = Running fast 3 = Running Slow 6 = Invalid
Val_Sts	SINT		Device confirmed status: 0 = Unknown 1 = Stopped 3 = Running Fast 4 = Running Slow 7 = Stopping 9 = Starting Fast 10 = Starting Slow 33 = Disabled
Val_Fault	SINT		This is the Device Fault Status, used to show the operator the most severe device fault: 0 = None 16 = Fail to Start 17 = Fail to Stop 32 = I/O Fault 34 = Configuration error

Table 8 - P_Motor2Spd Output Parameters

Output Parameter	Data Type	Alias For	Description
Val_Mode	SINT	Mode.Val	The current mode is shown with status bits and also as an enumeration 'Val_Mode' as follows: 0 = No mode 1 = Hand 2 = Maintenance 3 = Override 4 = Program (locked) 5 = Operator (locked) 6 = Program (unlocked, Operator is default) 7 = Operator (unlocked, Program is default) 8 = Program (unlocked, Program is default) 9 = Operator (unlocked, Operator is default)
Val_Owner	DINT		Current object owner ID (0=not owned).
Val_Notify	SINT		Current alarm level and acknowledgement (enumeration): 0 = No alarm 1 = Alarm cleared: a reset or acknowledge is required 2 = Low (acknowledged) 3 = Low (unacknowledged) 4 = Medium (acknowledged) 5 = Medium (unacknowledged) 6 = High (acknowledged) 7 = High (unacknowledged) 8 = Urgent (acknowledged) 9 = Urgent (unacknowledged)
Sts_Stopped	BOOL		1 = Motor requested to stop and is confirmed stopped.
Sts_StartingSlow	BOOL		1 = Motor requested to run slow and awaiting slow feedback.
Sts_StartingFast	BOOL		1 = Motor requested to run fast and awaiting fast feedback.
Sts_RunningSlow	BOOL		1 = Motor requested to run slow and is confirmed running slow.
Sts_RunningFast	BOOL		1 = Motor requested to run fast and is confirmed running fast.
Sts_Stopping	BOOL		1 = Motor requested to stop and awaiting stopped feedback.
Sts_Available	BOOL		1 = Motor available for control by automation (Program).
Sts_Bypass	BOOL		1 = Bypassable interlocks and permissives are bypassed.
Sts_BypActive	BOOL		1 = Bypassing active (bypassed or Maintenance).
Sts_Disabled	BOOL		1 = Motor is disabled.
Sts_NotRdy	BOOL		1 = Motor is not ready to run (independent of mode).
Nrdy_Disabled	BOOL		1 = Device Not Ready: • Device disabled by Maintenance • Configuration error • Interlock not OK • Permissive not OK • Operator state 0 priority command requires reset • Device failure (shed requires reset) • I/O fault (shed requires reset) • Device logic disabled/No mode
Nrdy_CfgErr			
Nrdy_Intlk			
Nrdy_Perm			
Nrdy_OperPrio			
Nrdy_Fail			
Nrdy_IOFault			
Nrdy_NoMode			
Sts_MaintByp	BOOL		1 = A Maintenance bypass is active, display icon.
Sts_AlmInh	BOOL		1 = An alarm is shelved, disabled, or suppressed, display icon.
Sts_Err	BOOL		1 = Error in configuration: see detail bits for reason.

Table 8 - P_Motor2Spd Output Parameters

Output Parameter	Data Type	Alias For	Description
Err_Timer	BOOL		1 = Error in configuration: start/stop check timer preset (use 0…2,147,483).
Err_Sim	BOOL		1 = Error in configuration: simulation timer preset (use 0…2,147,483).
Err_Pause	BOOL		1 = Error in configuration: invalid pause time (use 0…2,147,483).
Err_Alarm	BOOL		1 = Error in configuration: alarm minimum on time or severity.
Sts_Hand	BOOL	Mode.Sts_Hand	1 = Mode is: • Hand (supersedes Maintenance, Override, Program, Operator). • Maintenance (supersedes Override, Program, Operator). • Override (supersedes Program, Operator). • Program (auto). • Operator (manual).
Sts_Maint	BOOL	Mode.Sts_Maint	
Sts_Ovrd	BOOL	Mode.Sts_Ovrd	
Sts_Prog	BOOL	Mode.Sts_Prog	
Sts_Oper	BOOL	Mode.Sts_Oper	
Sts_ProgOperLock	BOOL	Mode.Sts_ProgOperLock	1 = Program or Operator has requested mode lock.
Sts_MAcqRcvd	BOOL	Mode.Sts_MAcqRcvd	1 = Maintenance acquire command received this scan.
Sts_NoMode	BOOL	Mode.Sts_NoMode	1 = No mode (disabled because EnableIn is false).
Sts_FailToStart	BOOL	FailToStart.Inp	1 = Motor failed to start.
Sts_FailToStop		FailToStop.Inp	1 = Motor failed to stop.
Sts_IntlkTrip		IntlkTrip.Inp	1 = Motor stopped by an Interlock Not OK (one-shot).
Sts_IOFault		IOFault.Inp	I/O Communication Fault Status: 0 = OK 1 = Bad
Alm_FailToStart	BOOL	FailToStart.Alm	1 = Motor Fail to Start alarm.
Alm_FailToStop		FailToStop.Alm	1 = Motor Fail to Stop alarm.
Alm_IntlkTrip		IntlkTrip.Alm	1 = Alarm: motor stopped by an Interlock Not OK.
Alm_IOFault		IOFault.Alm	1 = I/O fault alarm.
Ack_FailToStart	BOOL	FailToStart.Ack	1 = Fail to Start, Fail to Stop, Interlock Trip, or I/O Fault alarm acknowledged.
Ack_FailToStop		FailToStop.Ack	
Ack_IntlkTrip		IntlkTrip.Ack	
Ack_IOFault		IOFault.Ack	
Sts_FailToStartDisabled	BOOL	FailToStart.Disabled	1 = Fail to Start, Fail to Stop, Interlock Trip or I/O Fault alarm has been disabled (by Maintenance).
Sts_FailToStopDisabled		FailToStop.Disabled	
Sts_IntlkTripDisabled		IntlkTrip.Disabled	
Sts_IOFaultDisabled		IOFault.Disabled	
Sts_FailToStartShelved	BOOL	FailToStart.Shelved	1 = Fail to Start, Fail to Stop, Interlock Trip or I/O Fault alarm has been shelved by Operator.
Sts_FailToStopShelved		FailToStop.Shelved	
Sts_IntlkTripShelved		IntlkTrip.Shelved	
Sts_IOFaultShelved		IOFault.Shelved	
Sts_FailToStartSuppressed	BOOL	FailToStart.Suppressed	1 = Fail to Start, Fail to Stop, Interlock Trip or I/O Fault alarm has been suppressed (by Logic).
Sts_FailToStopSuppressed		FailToStop.Suppressed	
Sts_IntlkTripSuppressed		IntlkTrip.Suppressed	
Sts_IOFaultSuppressed		IOFault.Suppressed	

Table 8 - P_Motor2Spd Output Parameters

Output Parameter	Data Type	Alias For	Description
Rdy_RunSlow	BOOL		1 = Ready to receive Operator command (Run Slow, Run Fast, Stop, Bypass, or Check) (enables HMI button).
Rdy_RunFast			
Rdy_Stop			
Rdy_Bypass			
Rdy_Check			
Rdy_Disable	BOOL		1 = Ready to receive Maintenance command (Disable or Enable) (enables HMI button).
Rdy_Enable			
Rdy_Reset	BOOL		1 = At least one alarm or latched shed requires reset or acknowledgement.
Rdy_ResetAckAll			
P_Motor2Spd	BOOL		Unique parameter name for auto-discovery.

Two-speed Motor Local Configuration Tags

Configuration parameters that are array, string, or structure data types cannot be configured as parameters for Add-On Instructions. Configuration parameters of these types appear as local tags to the Add-On Instruction. Local tags can be configured through the HMI faceplates or in Studio 5000 Logix Designer® application by opening the Instruction Logic of the Add-On Instruction instance and then opening the Data Monitor on a local tag. These parameters cannot be modified by using controller logic or Logix Designer application export/import functionality.

Table 9 - P_Motor2Spd Input Local Configuration Tags

Tag Name	Data Type	Default	Description
Cfg_Desc	STRING_40	'Two Speed Motor'	Description for display on HMI. This string is shown in the title bar of the faceplate.
Cfg_FastText	STRING_16	'Fast'	Description for Fast speed.
Cfg_Label	STRING_20	'Motor Control'	Label for graphic symbol displayed on HMI. This string appears on the graphic symbol.
Cfg_SlowText	STRING_16	'Slow'	Description for Slow speed.
Cfg_Tag	STRING_20	'P_Motor2Spd'	Tag name for display on HMI. This string is shown in the title bar of the faceplate.

Operations

This section describes the primary operations for Add-On Instructions.

Modes

This instruction uses the following standard modes, which are implemented by using an embedded P_Mode Add-On Instruction.

Table 10 - Modes

Mode	Description
Operator	The Operator owns control of the device. Operator commands (OCmd_) and Operator settings (OSet_) from the HMI are accepted.
Program	Program logic owns control of the device. Program commands (PCmd_) and Program settings (PSet_) are accepted.
Override	Priority logic owns control of the device and supersedes Operator and Program control. Override Inputs (Inp_OvrdCmd and other Inp_OvrdXxxx values) are accepted. If so configured, bypassable interlocks and permissives are bypassed.
Maintenance	Maintenance owns control of the device and supersedes Operator, Program, and Override control. Operator commands and settings from the HMI are accepted. Bypassable interlocks and permissives are bypassed, and device timeout checks are not processed.
Hand	Hardwired logic or other logic outside the instruction owns control of the device. The instruction tracks the state of the device for bumpless transfer back to one of the other modes.
No Mode	The device is disabled and has no owner because the EnableIn input is false. The main instruction Logic routine is not being scanned. See Execution section for more information on EnableInFalse processing.

See Rockwell Automation Library of Process Objects: Common Mode Block (P_Mode) Reference Manual, publication SYSLIB-RM005, for more information.

Alarms

This instruction uses the following alarms, which are implemented by using embedded P_Alarm and P_Gate Add-On Instructions.

Alarm Name	P_Alarm Name	P_Gate Name	Description
Fail to Start	FailToStart	None	Raised when the motor has and is using run feedback, an attempt is made to start the motor, and the run feedback does not indicate that the motor is running within the configured time. If Fail to Start is configured as a shed fault, the motor is stopped and a reset is required in order to start the motor.
Fail to Stop	FailToStop	None	Raised when the motor has and is using run feedback, an attempt is made to stop the motor, and the run feedback does not indicate that the motor stopped within the configured time.
Interlock Trip	IntlkTrip	None	Raised when the motor is running and an interlock 'not OK' condition causes the motor to stop. If interlocks are not bypassed, a bypassable interlock or a non-bypassable interlock 'not OK' condition initiates an interlock trip. If interlocks are bypassed, only a non-bypassable interlock 'not OK' condition initiates an interlock trip.
I/O Fault	IOFault	None	Raised when the Inp_IOFault input is true. This input is usually used to indicate to the instruction that a communication failure has occurred for its I/O. If the I/O Fault is configured as a shed fault, the motor is stopped and not permitted to start until reset.

Parameters of the P_Alarm object can be accessed by using the following convention: [P_Alarm Name].[P_Alarm Parameter].

The Fail to Start and Fail to Stop status and alarms have a configurable delay to allow the run feedback time to align with the commanded output. This delay provides time for the motor to actually start or stop.

The Fail to Start and I/O fault conditions can be configured to alarm only, or to de-energize the motor (shed). If one of these conditions stops the motor, a reset is required to run.

See Rockwell Automation Library of Process Objects: Common Alarm Block (P_Alarm) Reference Manual, publication SYSLIB-RM002, for more information.

Simulation

When P_Motor2Spd is in simulation, the instruction keeps the outputs de-energized and simulates a working motor.

You must set the Inp_Sim parameter in the Controller to '1' to enable simulation.

The Loopback Test icon  is displayed at the bottom left of the Operator faceplate, indicating the device is in simulation.

You can set the number of seconds for the delay parameter, Cfg_SimFdbkT, to echo back running/stopped status when in simulation.

When you have finished in simulation, set the Inp_Sim parameter in the Controller to '0' to return to normal operation.

Execution

The following table explains the handling of instruction execution conditions.

Condition	Description
EnableIn False (false rung)	Handled the same as if the motor were disabled by command. The motor outputs are de-energized and the motor is shown as disabled on the HMI. The mode is shown as No mode. All alarms are cleared.
Powerup (prescan, first scan)	Any commands received before first scan are discarded. The motor is de-energized and treated as if it were commanded to stop. Embedded P_Mode and P_Alarm instructions are handled in accordance with their standard powerup procedures. See the Reference Manuals for the P_Mode and P_Alarm instructions for details.
Postscan (SFC transition)	No SFC postscan logic is provided.

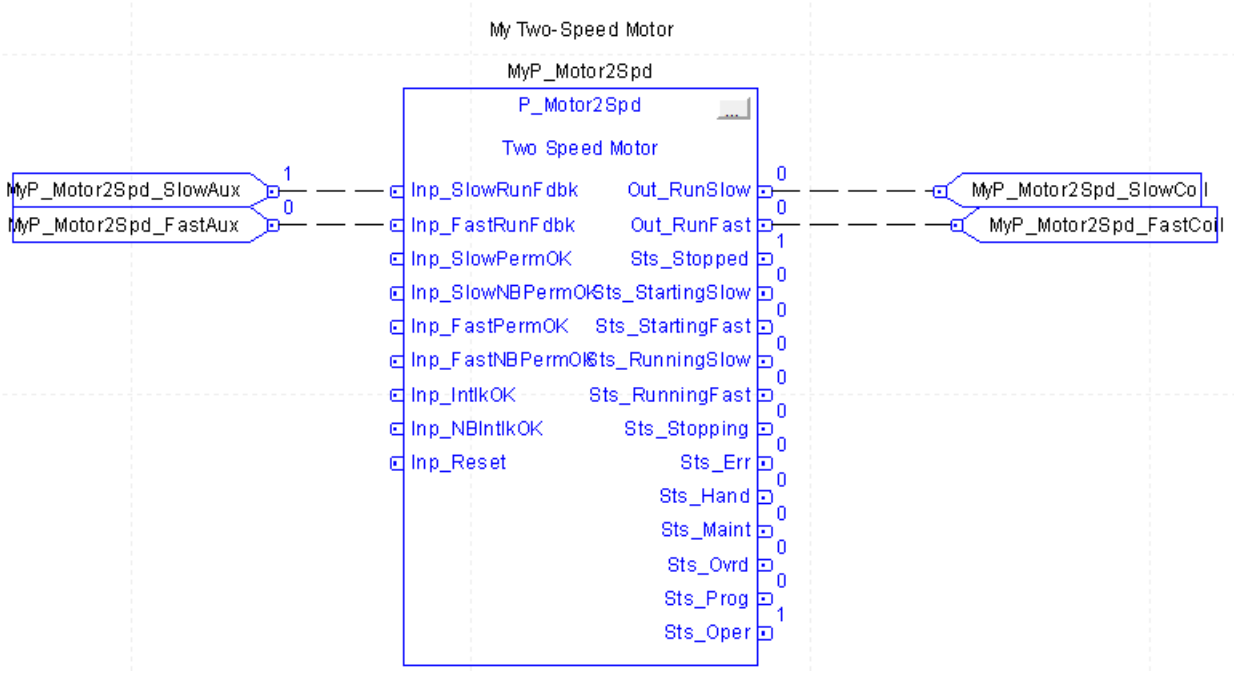
See the Logix5000 Controllers Add-On Instructions Programming Manual, publication 1756-PM010, for more information.

Programming Example

The following is a simple example of P_ Motor2Spd.

Boolean parameters MyP_ Motor2Spd_FastAux and MyP_Motor2Spd_SlowAux are used as inputs. Outputs MyP_ Motor2Spd_SlowCoil and MyP_Motor2Spd_FastCoil are wired to energize the appropriate starter circuits.

IMPORTANT To use inputs MyP_ Motor2Spd_FastAux and MyP_Motor2Spd_SlowAux, both Cfg_HasRunFdbk and Cfg_UseRunFdbk must be set to 1 (the default for both of these parameters is 0).



Display Elements

A display element (global object) is created once and can be referenced multiple times on multiple displays in an application. When changes are made to the original (base) object, the instantiated copies (reference objects) are automatically updated. Use of global objects, in conjunction with tag structures in the ControlLogix system, aid consistency and save engineering time.

Table 11 - P_Motor2Spd Display Elements Description

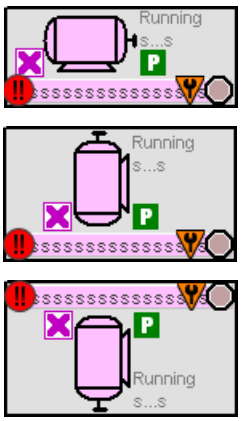
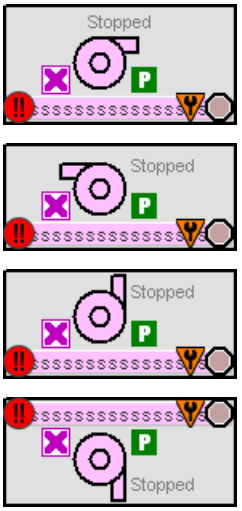
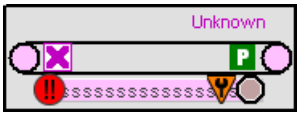
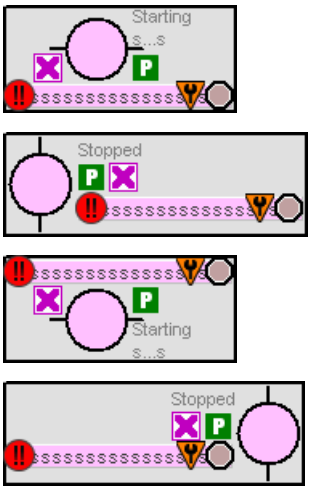
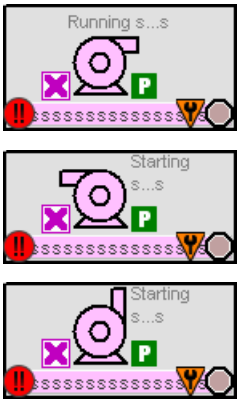
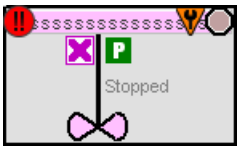
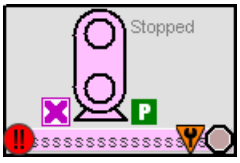
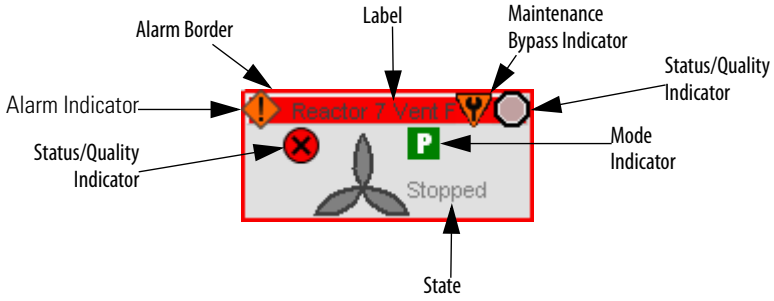
Display Element Name	Display Element	Description
GO_P_Motor_R GO_P_Motor_U GO_P_Motor_D		Motors operate in different positions: right, up, and down.
GO_P_Motor_Blower_R GO_P_Motor_Blower_L GO_P_Motor_Blower_U GO_P_Motor_Blower_D		Blowers operate in different positions: right, left, up, and down.
GO_P_Motor_Conveyer_R		Conveyor shown as a display element.

Table 11 - P_Motor2Spd Display Elements Description

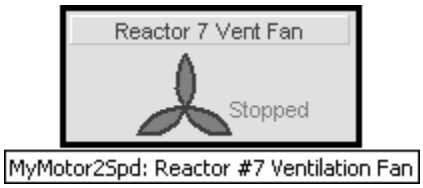
Display Element Name	Display Element	Description
GO_P_Motor_Inline_U GO_P_Motor_Inline_R GO_P_Motor_Inline_L GO_P_Motor_Inline_D		Inline motors operate in several positions: up, right, left, and down.
GO_P_Motor_Pump_R GO_P_Motor_Pump_L GO_P_Motor_Pump_U		Pumps operate in several positions: right, left, and up.
GO_P_Motor_Agitator_D		Agitator shown as a display element.
GO_P_Motor_Mixer_U		Mixer shown as a display element.
GO_P_Motor_RPump_U		Rotary gear pump shown as a display element.
GO_P_Motor_Fan_D		Fan shown as a display element.

Common attributes of the P_Motor2Spd graphic symbols include the following:

- Graphical representation of the device
- Current state of the motor in text
- Status/quality indicators
- Mode indicator
- Label
- Maintenance Bypass indicator
- Alarm indicator that changes color for the severity of the alarm
- Color-changing alarm border that blinks on unacknowledged alarm



Each graphic symbol includes a touch field over it that opens the object's faceplate. In addition, there is a tooltip on the graphic symbol that displays the object's configured tag and description.



State Indicators

The state indicator text changes and the graphic symbol color changes depending on the state of the motor.

Table 12 - Motor State Indicator Colors

Color	State
Blue	Starting or stopping
White	Running
Dark Gray	Stopped

Status/Quality Indicators

One of these symbols appears on the graphic symbol when the described condition is true.

Graphic Symbol	Description
	Invalid configuration.
	Data quality bad/failure.
	Data Quality degraded: uncertain, test, simulation, substitution, or out of specification.
	The input or device has been disabled.
	Device not ready to operate.

TIP When the Invalid Configuration indicator appears, you can find what configuration setting is invalid by following the indicators. Click the graphic symbol to open the faceplate. The Invalid Configuration indicator appears next to the appropriate tab at the top of the faceplate to guide you in finding the configuration error. Once you navigate to the tab, the misconfigured item is flagged with this indicator or appears in a magenta box.

For the two-speed motor instruction, the Invalid Configuration indicator appears under the following conditions:

- Fail to Start check time or Fail to Stop check time is set to a value less than zero or greater than 2,147,483 seconds.
- Change Speed time is set to a value less than zero or greater than 2,147,483 seconds.

- Feedback simulation time is set to a value less than zero or greater than 2,147,483 seconds.
- Alarm Severity is set to a value less than 1 or greater than 1000.
- Alarm Minimum On time is set to a value less than zero or greater than 2,147,483 seconds.

TIP When the Not Ready indicator appears, you can find what condition is preventing operation by following the indicators. Click the graphic symbol to open the faceplate. The Not Ready indicator appears next to the appropriate tab at the top of the faceplate to guide you in finding the condition. When you navigate to the tab, the condition preventing operation is flagged.

For the two-speed motor instruction, the Device Not Ready indicator appears under the following conditions:

- Device has been disabled by Maintenance.
- There is a configuration error.
- Interlock or Permissive is not OK.
- Operator State 0 priority command requires reset.
- Device Failure and shed requires reset.
- I/O Fault and shed requires reset.

Device logic is disabled or there is no mode.

Mode Indicators

One of these symbols appears on the right side of the graphic symbol to indicate the mode of the object instruction.

Graphic Symbol	Description
Transparent	Operator mode (if the default mode is Operator and the current mode is Operator, the mode indicator is transparent).
	Operator mode (if the default mode is Program).
	Operator mode locked.
Transparent	Program mode (if the default mode is Program and the current mode is Program, the mode indicator is transparent).
	Program mode (if the default mode is Operator).
	Program mode locked.
	Override mode

Graphic Symbol	Description
	Maintenance mode.
	Hand mode
	No mode.

TIP The images provided for the Operator and Program default modes are transparent; therefore, no mode indicators are visible if the device is in its default mode. This behavior can be changed by replacing the image files for these mode indicators with images that are not transparent.

See Rockwell Automation Library of Process Objects: Common Mode Block (P_Mode) Reference Manual, publication SYSLIB-RM005, for more information.

Alarm Indicators

One of these symbols appears on the left side of the label to indicate the described alarm condition and the alarm border and label background change color. The alarm border and label background blink if acknowledgement of an alarm condition is required. Once the alarm is acknowledged, the alarm border and label background remain the color that corresponds to the severity of the alarm.

Symbol	Border and Label Background	Description
	No change in color	Alarm Inhibit: an alarm is suppressed by the Program, disabled by Maintenance, or shelved by the Operator.
	White	Return to normal (no alarm condition), but a previous alarm has not been acknowledged.
	Blue	Low severity alarm.
	Yellow	Medium severity alarm.
	Red	High severity alarm.
	Magenta	Urgent severity alarm.
No symbol	No change in color	No alarm or alarm inhibit condition, and all alarms are acknowledged.

See Rockwell Automation Library of Process Objects: Common Alarm Block (P_Alarm) Reference Manual, publication SYSLIB-RM002, for more information.

Maintenance Bypass Indicator

This symbol appears to the right of the label to indicate that a maintenance bypass has been activated.

-

TIP When the Maintenance Bypass indicator appears, you can find what condition was bypassed by following the indicators. Click the graphic symbol to open the faceplate. The Maintenance Bypass indicator appears next to the appropriate tab at the top of the faceplate to guide you in finding the bypass. Once you navigate to the tab, the bypassed item is flagged with this indicator.

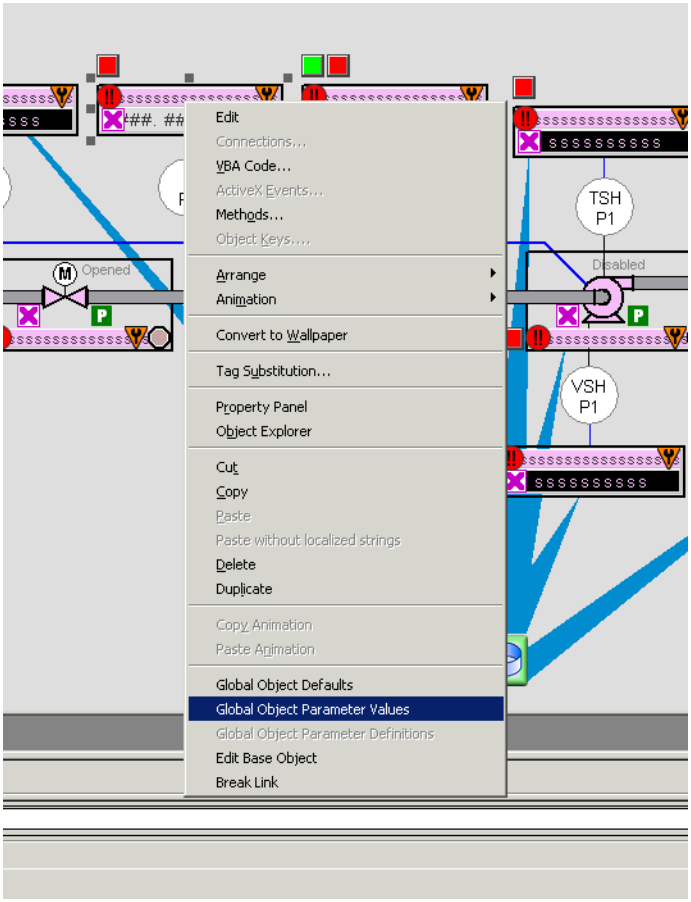
For the two-speed motor instruction, the Maintenance Bypass indicator appears under the following conditions:

- Bypassable interlocks and permissives have been bypassed.
- Motor is configured to have run feedbacks (on the Engineering tab of the faceplate), but the run feedbacks are not being used (selection on the Maintenance tab of the faceplate).

Using Display Elements

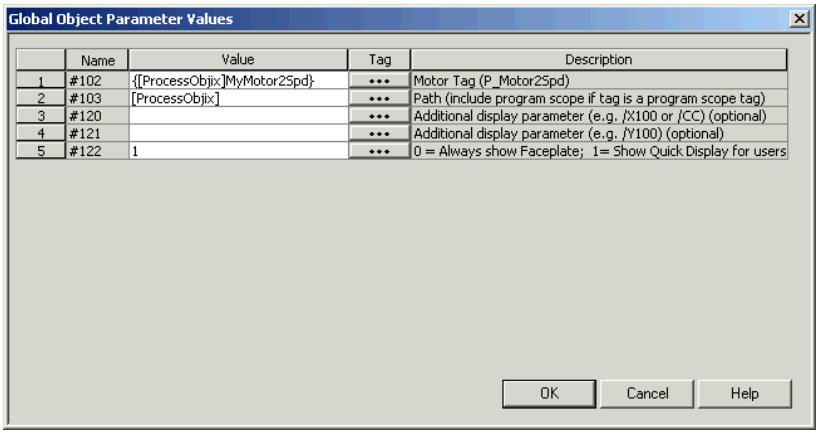
The global objects for P_Motor2Spd can be found in the global object file (RA-BAS) P_Motor2Spd Graphics Library.ggfx. Follow these steps to use a global object.

1. Copy the global object from the global object file and paste it in the display file.



2. In the display, right-click the global object and choose Global Object Parameter Values.

The Global Object Parameter Values dialog box appears.



The global object parameters are as follows.

Parameter	Required	Description
#102	Y	Object tag to point to the name of the associated object Add-On Instruction in the controller.
#103	Y	Path used for display navigation features to other objects. Include program scope if tag is a program scope tag.
#120	N	Additional parameter to pass to the display command to open the faceplate. Typically used to define position for the faceplate.
#121	N	Additional parameter to pass to the display command to open the faceplate. if defining X and Y coordinate, separate parameters so that X is defined by #120 and Y is defined by #121. This lets these same parameters to be used in subsequent display commands originating from the faceplate.
#122	Y	These are the options for the global object display: 0 = Always show faceplate 1 = Show Quick Display for users without Maintenance access (Code C) 2= Always show Quick Display

3. In the Value column, type the tag or value as specified in the Description column.

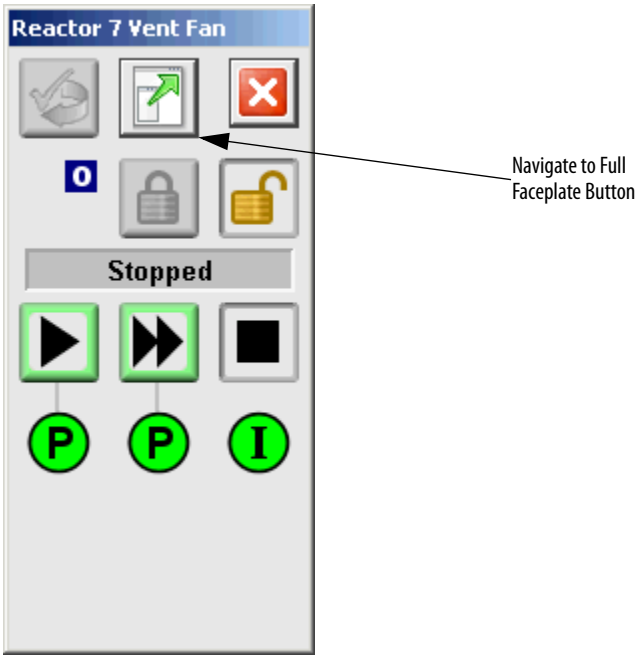
TIP Click the ellipsis (. . .) to browse and select a tag.

Values for items marked '(optional)' can be left blank.

4. Click OK.

Quick Display

The Quick Display screen provides means for operators to perform simple interactions with the P_Motor2Spd instruction instance. From the Quick Display, you can navigate to the faceplate for full access.

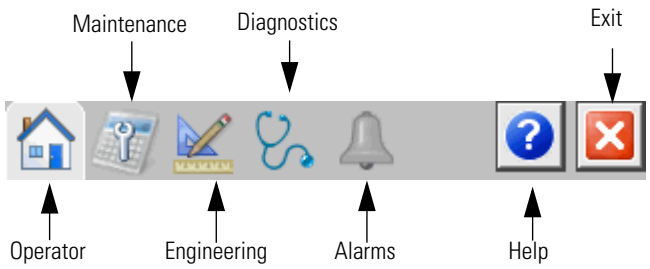


Faceplate

The P_Motor2Spd faceplate consists of five tabs and each tab consists of one or more pages.

The title bar of each faceplate contains the value of local configuration tags Cfg_Tag and Cfg_Desc.

The Operator tab is displayed when the faceplate is initially opened. Click the appropriate icon at the top of the faceplate to access a specific tab.



The faceplate provides the means for operators, maintenance personnel, engineers, and others to interact with the P_Motor2Spd instruction instance, including viewing its status and values and manipulating it through its commands and settings. When a given input is restricted via FactoryTalk View security, the required user security code letter is shown in the tables that follow.

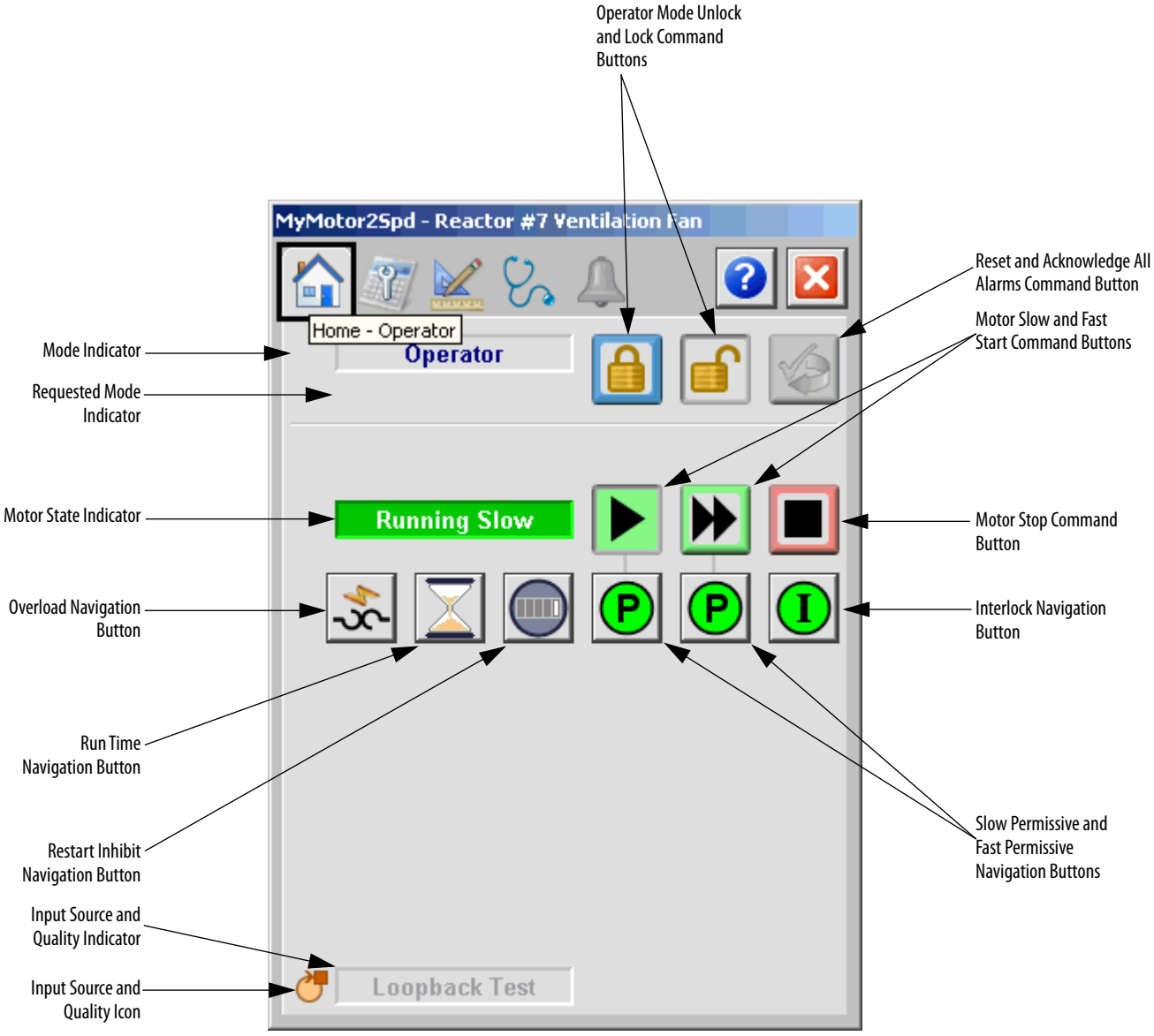
Operator Tab

The Faceplate initially opens to the Operator ('Home') tab. From here, an operator can monitor the device status and manually operate the device when it is in Operator Mode.

The Operator tab has buttons to start and stop the motor when it is in the proper mode and shows the following information:

- Current mode (Operator, Program, Override, Maintenance, or Hand)
- Requested mode indicator (appears only if Operator or Program mode has been superseded by another mode)
- Motor state (stopping, stopped, starting slow, starting fast, running slow, running fast, disabled, or I/O fault)
- Interlock and permissive states
- Input Source and Quality indicator (See 'SrcQ' in the Output parameters table on page 18 for details)

IMPORTANT This indicator is not displayed if 'Motor has Run Feedback' on page 1 of the Engineering tab is not checked. (See Engineering Tab Page 1 on page 44.)



The following table shows the functions included on the Operator tab.

Table 13 - Operator Tab Description

Function	Action	Security
	Click to lock in Operator mode. Function locks the mode in Operator mode, preventing the program from taking control.	Manual Device Operation (Code B)
	Click to unlock Operator mode. Function unlocks Operator mode, letting the program take control.	
	Click to request Program mode.	

Table 13 - Operator Tab Description

Function	Action	Security
	Click to request Operator mode.	Manual Device Operation (Code B)
	Click to start motor slow.	
	Click to start motor fast.	
	Click to stop motor.	
	Click to open Interlocks faceplate.	None
	Click to open Permissives (Fast or Slow) faceplate.	
	Click to open Restart Inhibit faceplate.	
	Click to open runtime faceplate.	
	Click to open Overload faceplate.	
	Click to reset and acknowledge all alarms.	Acknowledge Alarms (Code F)

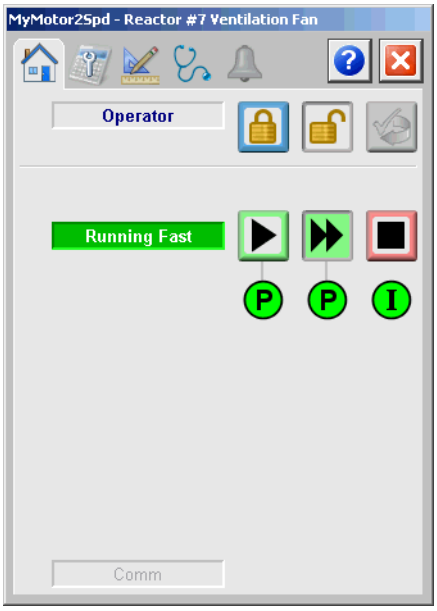
If the object is configured to have permissive and interlock objects (for example, Cfg_HasPermObj (Fast or Slow) or Cfg_HasIntlkObj is true), the permissive and interlock indication become buttons that open the faceplates of the source objects used as a permissive or interlock. Often this is a P_Perm or P_Intlk object). If the object is not configured in this way, the permissive or interlock symbols are indicators only.

The Operator tab also has a button to open the Restart Inhibit faceplate if the valve is configured to use the P_ResInh object (Cfg_HasResInh = 1). When the object is not configured to have an P_ResInh instruction, the Restart Inhibit button is not displayed.

The Operator tab also has a button to open the runtime faceplate if the valve is configured to use the P_RunTime object (Cfg_HasRunTime = 1). When the object is not configured to have an P_RunTime instruction, the runtime button is not displayed.

The Operator tab also has a button to open an Overload faceplate if the motor is configured to have an associated Overload object (Cfg_HasOvld = 1). When the object is not configured to have an associated Overload object, the Overload button is not displayed.

The following image shows the Operator tab when it is not configured to have Permissive (Fast or Slow) or Interlock objects. It is also not configured to have P_ResInh, P_RunTime, or P_Ovld instructions.



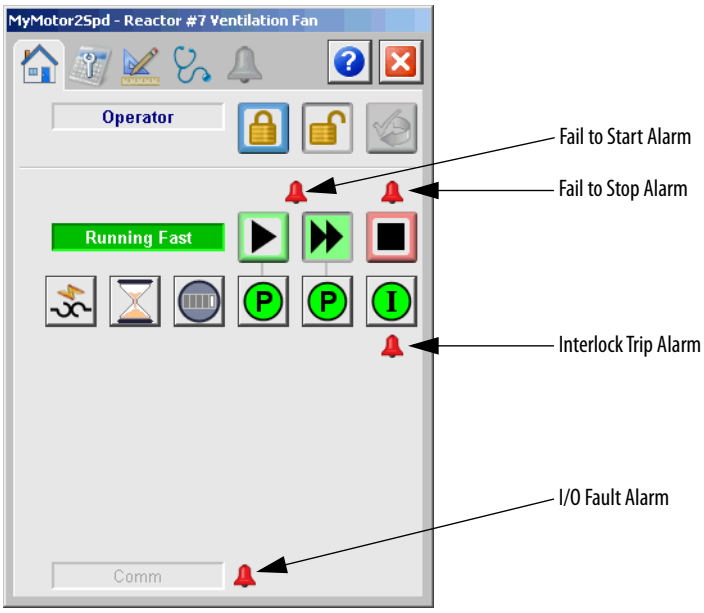
See these publications for more information:

- Rockwell Automation Library of Process Objects: Permissives with Bypass (P_Perm) Reference Manual, publication SYSLIB-RM007
- Rockwell Automation Library of Process Objects: Interlock with First Out and Bypass (P_Intlk) Reference Manual, publication SYSLIB-RM004
- Rockwell Automation Library of Process Objects: Restart Inhibit for Large Motor (P_ResInh) Reference Manual, publication SYSLIB-RM009
- Rockwell Automation Library of Process Objects: Runtime and Starts (P_RunTime) Reference Manual, publication SYSLIB-RM010
- Rockwell Automation Library of Process Objects: E1 Plus Overload Relay (EtherNet/IP) (P_E1PlusE) Reference Manual, publication SYSLIB-RM049
- Rockwell Automation Library of Process Objects: E3/E3Plus Overload Relay (P_E3Ovld) Reference Manual, publication SYSLIB-RM050
- Rockwell Automation Library of Process Objects: E300 Overload Relay (EtherNet/IP) (P_E300Ovld) Reference Manual, publication SYSLIB-RM051

One of these symbols appears to indicate the described Permissive or Interlock condition.

Permissive Symbol	Interlock Symbol	Description
		One or more conditions not OK
		Non-bypassed conditions OK
		All conditions OK, bypass active
		All conditions OK

Alarm indicators appear on the Operator tab when the corresponding alarm occurs.



The following table shows the alarm status symbols used on the Operator tab.

Table 14 - Operator Tab Alarm Status

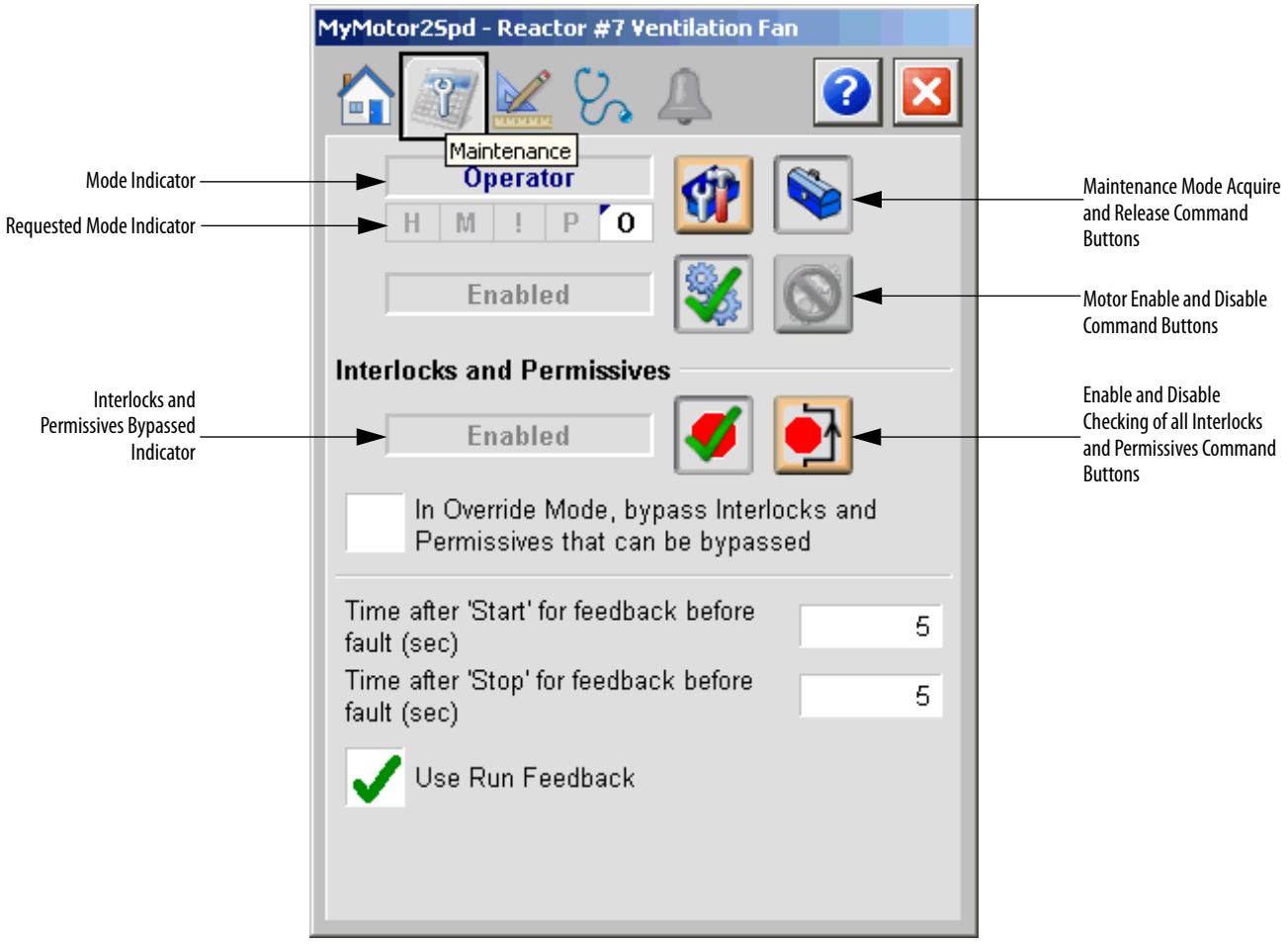
Graphic Symbol	Alarm Status
	In Alarm (Active Alarm)
	In Alarm and Acknowledged
	Out of Alarm but not Acknowledged
	Alarm Suppressed (by Program)
	Alarm Disabled (by Maintenance)
	Alarm Shelved (by Operator)

Maintenance Tab

Maintenance personnel use the information and controls on the Maintenance tab to adjust device parameters, troubleshoot and temporarily work around device problems, and disable the device for routine maintenance.

The Maintenance tab shows the following information:

- Current mode (Operator, Program, or Maintenance).
- Requested modes indicator highlights all modes that have been requested. The leftmost highlighted mode is the active mode.
- Whether the motor is enabled or disabled.
- Interlock and permissive bypassed/enabled indicator.
- Whether Override mode bypasses the bypassable interlocks and permissives.
- The amount of time after commanding the motor to start or stop for the motor feedback to show running or stopped before declaring a Fail to Start or Fail to Stop fault.
- Whether to check the motor's run feedbacks when starting or stopping the motor.



The following table shows the functions on the Maintenance tab.

Table 15 - Maintenance Tab 1 Description

Function	Action	Security	Configuration Parameters
	Click for Maintenance mode.	Equipment Maintenance (Code C)	None
	Click to release Maintenance mode.		
	Click to enable motor.		
	Click to disable motor.		
	Click to enable checking of all interlocks and permissives.	Disable Alarm Bypass Permissives and Interlocks (Code H)	
	Click to bypass checking of bypassable interlocks and permissives.		
In Override Mode, bypass Interlocks and Permissives that can be bypassed	Check to bypass bypassable interlocks and permissives in Override mode.		Cfg_OvrdPermIntlk
Time after 'Start' for Feedback before Fault (seconds)	Type the time to allow the run feedback to show that the motor has started before raising a fail to start alarm.	Configuration and Tuning Maintenance (Code D)	Cfg_failToStartT
Time after 'Stop' for Feedback before Fault (seconds)	Type the time to allow the run feedback to show that the motor has stopped before raising a Fail to Stop alarm.		Cfg_FailToStopT
Use Run feedback	Check to have this instruction use the run feedback to check for motor Fail to Start or Fail to Stop. **IMPORTANT:** This option is only available if 'Motor has Run Feedback' on page 1 of the Engineering tab is checked. (See Engineering Tab Page 1 on page 44.)	Equipment Maintenance (Code C)	Cfg_UseRunFdbk

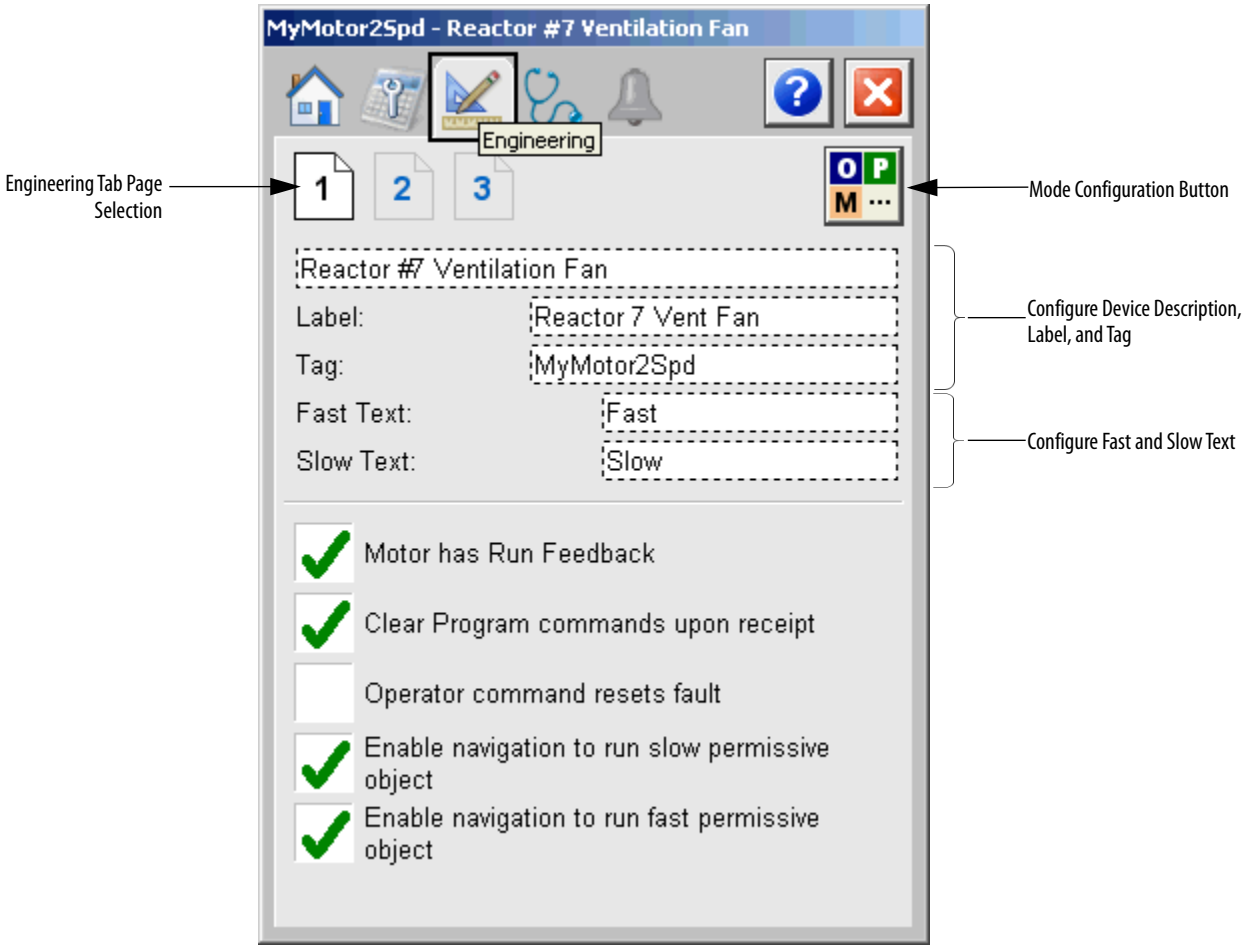
Engineering Tab

The Engineering tab provides access to device configuration parameters and ranges, options for device and I/O setup, displayed text, and faceplate-to-faceplate navigation settings, for initial system commissioning or later system changes.

The Engineering tab is divided into three pages.

Engineering Tab Page 1

On page 1 of the Engineering tab, you can configure the description, label, tag, and enable navigation to a run slow or run fast permissive object.

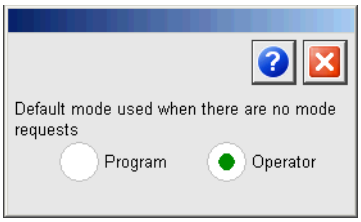


The following table lists the functions on page 1 of the Engineering tab.

Table 16 - Engineering Tab Page 1 Descriptions

Function	Action	Security	Configuration Parameters
	Click to navigate to the Mode Configuration display.	Engineering Configuration (Code E)	See Mode Configuration display on page 46
Description	Type the device description to be used on the faceplate title bar.		Cfg_Desc
Label	Type the device label to be used on the graphic symbol.		Cfg_Label
Tag	Type the tagname to be used on the faceplate title bar and in the Tooltip.		Cfg_Tag
Fast Text	Type the text to display when the motor is running fast.		Cfg_FastText
Slow Text	Type the text to display when the motor is running slow.		Cfg_SlowText
Motor has Run Feedback	Check if the motor provides run feedback to Inp_SlowRunFdbk and Inp_FastRunFdbk. Clear this checkbox if there is no run feedback. **IMPORTANT:** Checking this places the device in Maintenance Bypass unless 'Use Run Feedback' on the Maintenance tab is checked. (See Maintenance Tab on page 42.)		Cfg_UseRunFdbk
Clear Program Commands on receipt	Check to clear Program commands on receipt See Edge and Level.		Cfg_PCmdClear
Operator command resets fault	Check to allow the Operator Start Slow, Start Fast, or Stop command to reset any previous faults (I/O fault, Fail to Start, Fail to Stop, Interlock Trip), then start or stop motor. Clear this checkbox to only reset faults using the reset code.		• Cfg_FailToStartT • Cfg_FailToStopT
Enable navigation to run slow or run fast permissive object	Check if a run slow or run fast permissive object is used with this motor. **IMPORTANT:** The name of the Slow or Fast Permissive object in the controller must be this object's name with the suffix '_FastPerm' or '_SlowPerm'. For example, if your P_Motor2Spd object has the name 'Motor123', then its Slow Permissive object must be named 'Motor123_SlowPerm' or its Fast Permissive object must be named 'Motor123_FastPerm'.		• Cfg_HasSlowPermObj • Cfg_HasFastPermObj

Mode Configuration Display

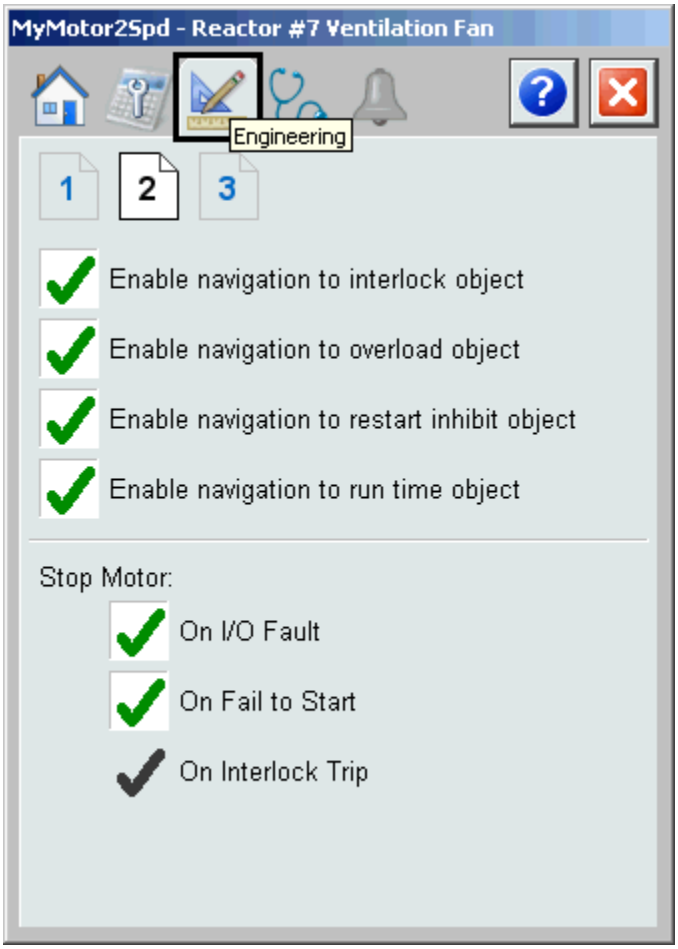


This display lets you select the default mode for the object by selecting the appropriate mode.

IMPORTANT	If no mode is being requested, changing the default mode changes the mode of the instruction.

You must have FactoryTalk View security code E to select the default mode on this display.

Engineering Tab Page 2



The following table shows the functions on page 2 of the Engineering tab.

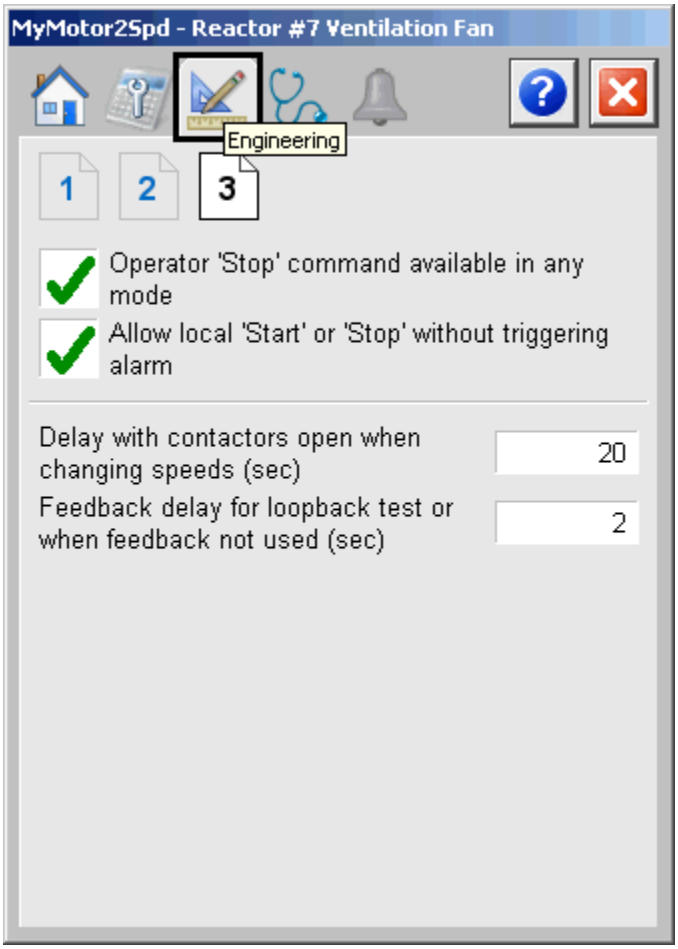
Table 17 - Engineering Tab Page 2 Descriptions

Function	Action	Security	Configuration Parameters
Enable navigation to an interlock object	Check if a run slow or run fast permissive object is used with this motor. **IMPORTANT:** The name of the Interlock object in the controller must be this object's name with the suffix '_Intlk'. For example, if your P_Motor2Spd object has the name 'Motor123', then its Interlock object must be named 'Motor123_Intlk'.	Engineering Configuration (Code E)	Cfg_HasIntlkObj
Enable navigation to an overload object	Check if an overload object is used with this motor. **IMPORTANT:** The name of the Overload object in the controller must be this object's name with the suffix '_Ovld'. For example, if your P_Motor2Spd object has the name 'Motor123', then its Overload object must be named 'Motor123_Ovld'.		Cfg_HasOvldObj
Enable navigation to a restart inhibit object	Check the appropriate checkbox if a restart inhibit object is used with this motor. **IMPORTANT:** The name of the Restart Inhibit object in the controller must be this object's name with the suffix '_ResInh'. For example, if your P_Motor2Spd object has the name 'Motor123', then its Restart Inhibit object must be named 'Motor123_ResInh'.		Cfg_HasResInhObj
Enable navigation to a runtime object	Check if a runtime object is used with this motor. **IMPORTANT:** The name of the runtime object in the controller must be this object's name with the suffix '_RunTime'. For example, if your P_Motor2Spd object has the name 'Motor123', then its runtime object must be named 'Motor123_RunTime'.		Cfg_HasRunTimeObj

Table 17 - Engineering Tab Page 2 Descriptions

Function	Action	Security	Configuration Parameters
Stop Motor: On I/O Fault	Check to stop the motor if an I/O fault is detected. Clear this checkbox to show only the I/O fault status/alarm and not stop the motor if an I/O fault is detected.	Engineering Configuration (Code E)	Cfg_ShedOnIOFault
Stop Motor: On Fail to Start	Check to stop the motor if a fail to Start fault is detected. Clear this checkbox to show only the Fail to Start status/ alarm and not stop the motor if a fail to Start fault is detected.		Cfg_ShedOnFailToStart
Stop Motor: On Interlock Trip	The motor always stops on an interlock trip. This item cannot be cleared. It is displayed as a reminder that the Interlock Trip function always trips the motor.		None

Engineering Tab Page 3



The following table shows the functions on page 3 of the Engineering tab.

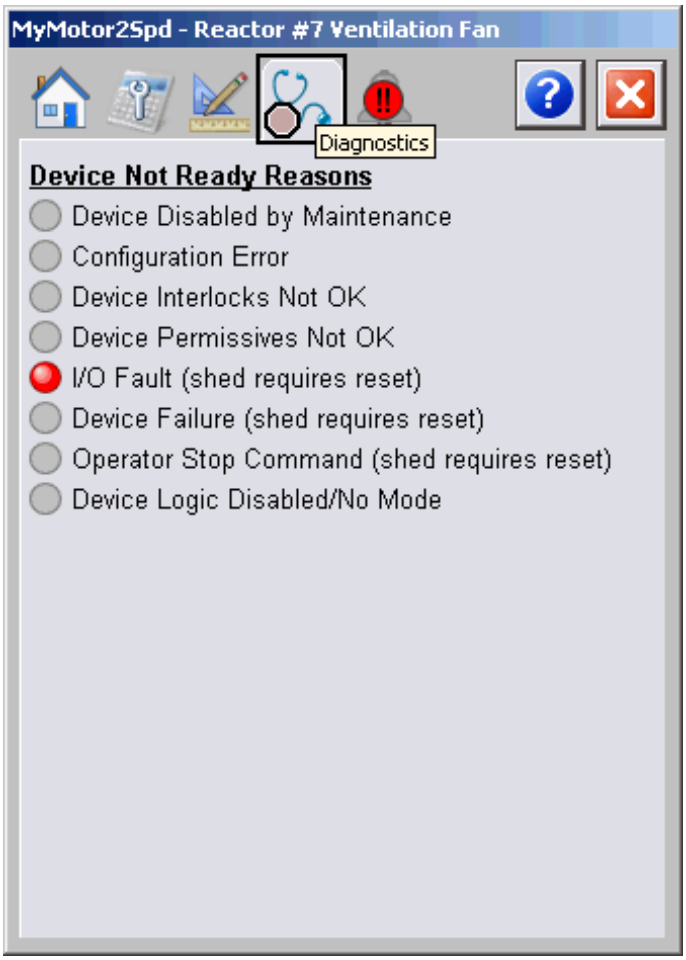
Table 18 - Engineering Tab Page 3 Descriptions

Function	Action	Security	Configuration Parameters
Operator 'Stop' command available in any mode	Check to have the Operator Stop command available in any mode. Clear this checkbox to have the Operator Stop command available only in the Operator and Maintenance modes.	Engineering Configuration (Code E)	Cfg_OperStopPrio
Allow local 'Start' or 'Stop' without triggering alarm	Check to allow local Start/Stop without alarm. Clear this checkbox to allow start/stop only using Program or Operator commands or Override Logic.		Cfg_AllowLocal
Delay with contactors open when changing speeds (seconds)	Type the time delay between when the run output has turned off for one speed and when it is turned on for the other speed.		Cfg_ChangeSpdT
Feedback Delay in simulation (seconds)	Type the time delay (in seconds) for the running or stopped status to be echoed back when the simulation is enabled or when run feedback is not used.		Cfg_SimFdbkT

Diagnostics Tab

The Diagnostic tab provides indications that are helpful in diagnosing or preventing device problems, which can include specific reasons a device is 'Not Ready', device warnings and faults, warning and fault history, and predictive/preventive maintenance data.

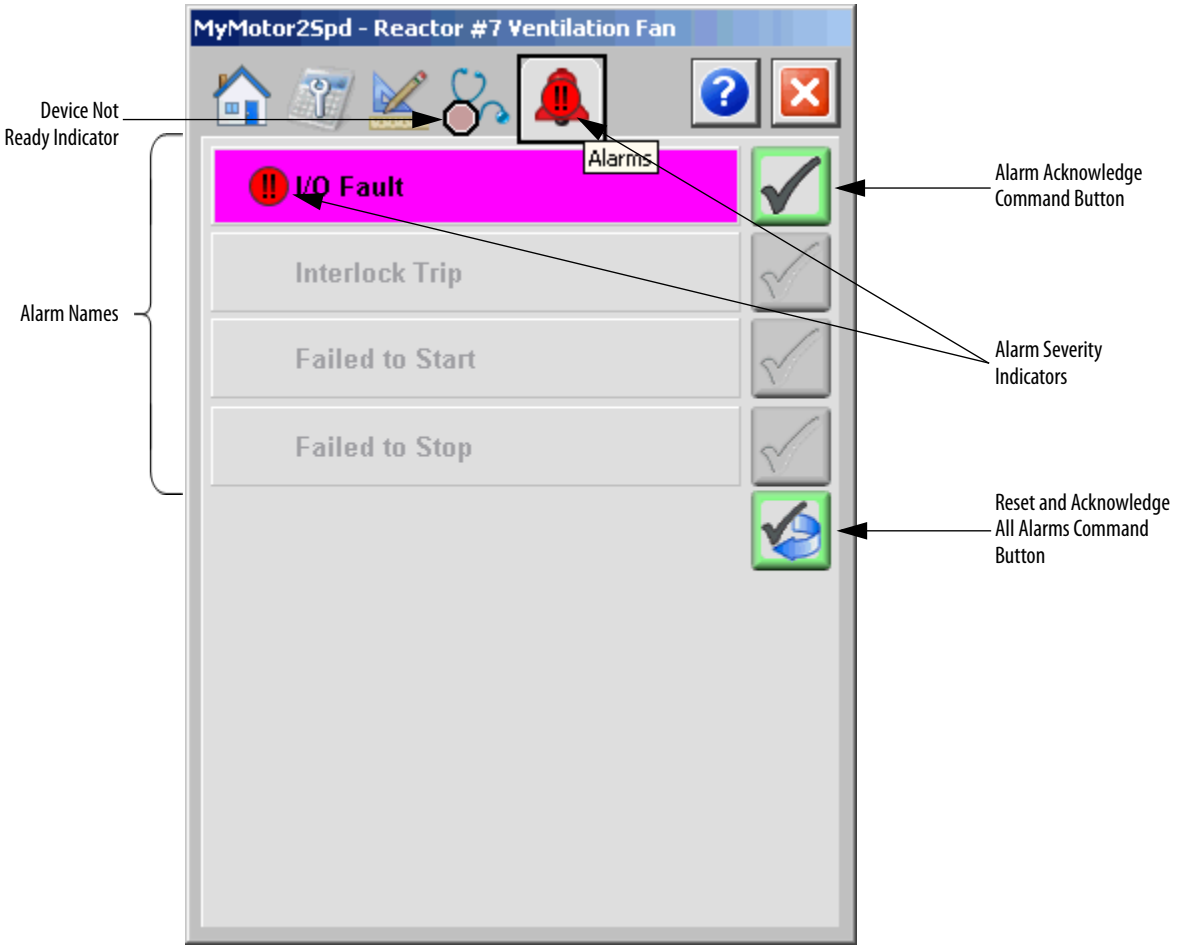
The Diagnostics tab displays possible reasons the device is not ready.



The preceding image indicates that the device is not ready because of an I/O fault shed condition that requires reset.

Alarms Tab

The Alarms tab shows all available alarms for the device and their current status. From here, alarms can be acknowledged and reset. Click an alarm name to open the alarm detail faceplate for that alarm, where the alarm can be shelved by the operator, disabled by maintenance personnel, or configured by engineering.



IMPORTANT The 'Failed to Start' and 'Failed to Stop' alarms are not available if 'Motor has Run Feedback' on page 1 of the Engineering tab is not checked.
(See Engineering Tab Page 1 on page 44.)

Click an alarm name to open the P_Alarm faceplate for that alarm. From the P_Alarm faceplate, you can configure and perform additional operations on the alarm.

If an alarm is active, the panel behind the alarm changes color to match the severity of the alarm. The color of the bell icon at the top of the faceplate shows the severity of the highest active alarm, and the icon blinks if any alarm is unacknowledged or requires reset.

Table 19 - Alarm Severity Colors

Color	Definition
Magenta	Urgent
Red	High
Yellow	Medium
Blue	Low
White (bell icon)	Alarm has cleared but is unacknowledged
Background (Light Gray)	No alarm

The following table shows the functions on the Alarms tab.

Table 20 - Alarms Tab Description

Function	Action	Security
Alarm Name	Click an alarm name to open the associated P_Alarm faceplate.	None
	Click to acknowledge the alarm.	Acknowledge Alarms (Code F)
	Click to reset and acknowledge all alarms.	

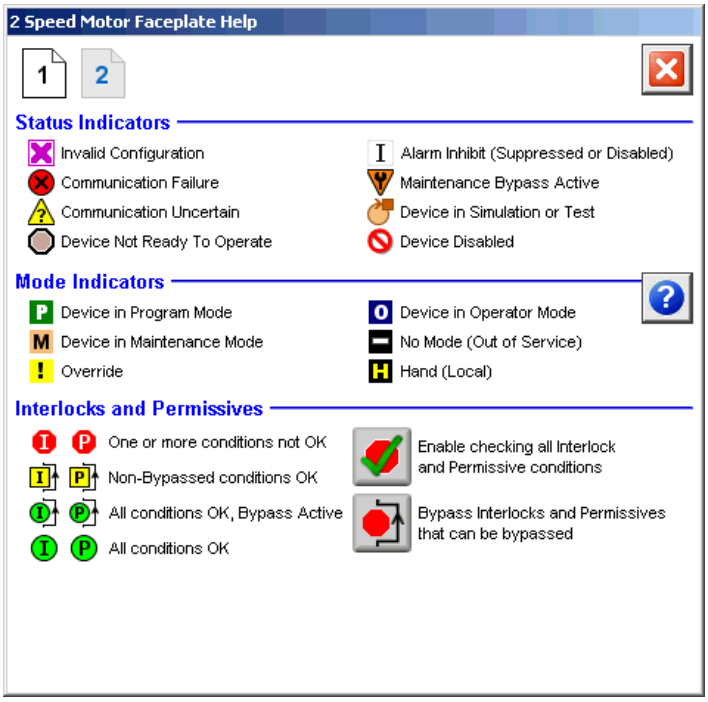
When the Reset and Acknowledge All Alarms button is enabled, the panel behind the alarm blinks, indicating the alarm requires acknowledgement or reset. The Alarm Acknowledge button is enabled if the alarm requires acknowledgment. Click the button with the check mark to acknowledge the alarm.

See Rockwell Automation Library of Process Objects: Common Alarm Block (P_Alarm) Reference Manual, publication SYSLIB-RM002, for more information.

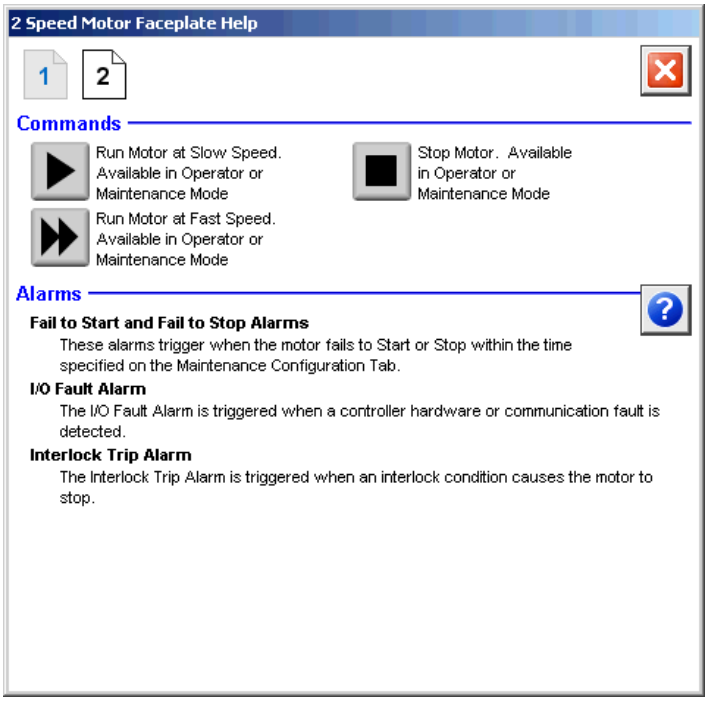
Two-speed Motor Faceplate Help

The Two-speed Faceplate Help is divided into two pages.

Faceplate Help Page 1



Faceplate Help page 2



Notes:

Rockwell Automation Support

Rockwell Automation provides technical information on the Web to assist you in using its products. At http://www.rockwellautomation.com/support you can find technical and application notes, sample code, and links to software service packs. You can also visit our Support Center at https://rockwellautomation.custhelp.com/ for software updates, support chats and forums, technical information, FAQs, and to sign up for product notification updates.

In addition, we offer multiple support programs for installation, configuration, and troubleshooting. For more information, contact your local distributor or Rockwell Automation representative, or visit http://www.rockwellautomation.com/services/online-phone.

Installation Assistance

If you experience a problem within the first 24 hours of installation, review the information that is contained in this manual. You can contact Customer Support for initial help in getting your product up and running.

United States or Canada	1.440.646.3434
Outside United States or Canada	Use the Worldwide Locator at http://www.rockwellautomation.com/rockwellautomation/support/overview.page, or contact your local Rockwell Automation representative.

New Product Satisfaction Return

Rockwell Automation tests all of its products to help ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

United States	Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor to complete the return process.
Outside United States	Please contact your local Rockwell Automation representative for the return procedure.

Documentation Feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete this form, publication RA-DU002, available at http://www.rockwellautomation.com/literature/.

Rockwell Automation maintains current product environmental information on its website at http://www.rockwellautomation.com/rockwellautomation/about-us/sustainability-ethics/product-environmental-compliance.page.

Rockwell Otomasyon Ticaret A.Ş., Kar Plaza İş Merkezi E Blok Kat:6 34752 İçerenköy, İstanbul, Tel: +90 (216) 5698400

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

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 SYSLIB-RM012G-EN-E - February 2017

Supersedes Publication SYSLIB-RM012F-EN-E - January 2016

Copyright © 2017 Rockwell Automation, Inc. All rights reserved. Printed in the U.S.A.