

GuardLogix® Controller Systems

Catalog Numbers 1756-L61S, 1756-L62S, 1756-L63S, 1756-L71S, 1756-L72S, 1756-L73S, 1756-L73SXT, 1756-LSP, 1756-L7SP, 1756-L7SPXT, 1768-L43S, 1768-L45S, RSLogix 5000® Version 20 and earlier



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).

This manual contains new and updated information.

New and Updated Information

This table contains the changes made to this revision.

| Topic | Page |
|--|-------------|
| Added the definition of Claim Limit to the Terms and Definitions table. | 5 |
| Removed the Important note. | 11 |
| Removed safety-lock information. | 12 |
| Updated Table 1. | 14 |
| Updated footnotes in Table 2. | 15 |
| Updated information about calculating PFH values. | 18 |
| Added an Important note to the Programming Overview section. | 26 |
| Added information about the On- to Off- delay filter. | 28 |
| Added information about unexpected changes to the configuration signature. | 29 |
| Made minor changes to the information on configuring a safety module. | 30 |
| Added a new graphic and explanatory text, and updated the Configure Always section. | 31 |
| Updated the Important note. | 36 |
| Updated the information on configuring Guard I/O safety modules. Also updated the second Important note. | 40 |
| Added a new Important note. | 44 |
| Added information about the safety task signature. | 55 |
| Added a bullet to the Lock the GuardLogix Controller section. | 58 |
| Deleted the Certification column from the tables. | 71 and 72 |
| Updated the instructions for the connection reaction time limits. | 85 |
| Added references to Guard I/O module user manuals for safety data. | 95 |
| Updated the list of available SIL 2 output modules. | 105 |
| Added an Important note about the safety task signature. | 106 |

Notes:

| Topic | Page |
|---------------------------|------|
| Understanding Terminology | 5 |
| Additional Resources | 6 |

This manual is intended to describe the GuardLogix® controller system, which is **type-approved** and certified for use in safety applications up to and including SIL CL 3 according to IEC 61508 and IEC 62061, safety applications up to and including Performance Level PL_e (Category 4) according to ISO 13849-1.

This publication covers both 1756 and 1768 GuardLogix controller systems. When ‘GuardLogix controllers’ is used alone in this publication, it refers to both 1756 and 1768 GuardLogix controllers. Information specific to one controller type will include the bulletin number, 1756 or 1768.

Use this manual if you are responsible for the development, operation, or maintenance of a GuardLogix controller-based safety system. You must read and understand the safety concepts and requirements presented in this manual prior to operating a GuardLogix controller-based safety system.

Understanding Terminology

The following table defines terms used in this manual.

Table 1 - Terms and Definitions

| Abbreviation | Full Term | Definition |
|--------------|---|--|
| 1oo2 | One out of Two | Identifies the programmable electronic controller architecture. |
| CIP | Common Industrial Protocol | An industrial communication protocol used by Logix 5000™ based automation systems on Ethernet/IP™, ControlNet™, and DeviceNet™ communication networks. |
| CIP Safety | Common Industrial Protocol – Safety Certified | SIL 3 -rated version of CIP. |
| CL | Claim Limit | The maximum SIL level that can be achieved. |
| DC | Diagnostic Coverage | The ratio of the detected failure rate to the total failure rate. |
| EN | European Norm. | The official European Standard. |
| GSV | Get System Value | A ladder logic instruction that retrieves specified controller status information and places it in a destination tag. |
| PC | Personal Computer | Computer used to interface with, and control, a Logix-based system via RSLogix 5000® programming software. |
| PFD | Probability of Failure on Demand | The average probability of a system to fail to perform its design function on demand. |
| PFH | Probability of Failure per Hour | The probability of a system to have a dangerous failure occur per hour. |
| PL | Performance Level | ISO 13849-1 safety rating. |

Table 1 - Terms and Definitions

| Abbreviation | Full Term | Definition |
|--------------|-----------------------|---|
| SNN | Safety Network Number | A unique number that identifies a section of a safety network. |
| SSV | Set System Value | A ladder logic instruction that sets controller system data. |
| -- | Standard | Any object, task, tag, program, or component in your project that is not a safety-related item (that is, standard controller refers generically to a ControlLogix or CompactLogix™ controller). |

Additional Resources

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

| Resource | Description |
|--|---|
| GuardLogix Controllers User Manual, publication 1756-UM020 | Provides information on installing, configuring, and programming the GuardLogix system |
| CompactLogix Controllers Installation Instructions, publication 1768-IN004 | Provides information on installing Compact GuardLogix controllers |
| 1768 Compact GuardLogix Controllers User Manual, publication 1768-UM002 | Details how to configure, program, and operate a 1768 CompactLogix system, and provides technical specifications |
| GuardLogix Safety Application Instruction Set Reference Manual, publication 1756-RM095 | Provides information on the GuardLogix Safety Application instruction set |
| Guard I/O DeviceNet Safety Modules User Manual, publication 1791DS-UM001 | Provides information on using Guard I/O™ DeviceNet Safety modules |
| Guard I/O EtherNet/IP Safety Modules User Manual, publication 1791ES-UM001 | Provides information on using Guard I/O EtherNet/IP Safety modules |
| POINT Guard I/O Safety Modules User Manual, publication 1734-UM013 | Provides information on installing and operating POINT Guard I/O™ modules |
| Using ControlLogix in SIL 2 Applications Safety Reference Manual, publication 1756-RM001 | Describes requirements for using ControlLogix controllers, and GuardLogix standard task, in SIL 2 safety control applications |
| Logix5000 General Instruction Set Reference Manual, publication 1756-RM003 | Provides information on the Logix5000™ Instruction Set |
| Logix Common Procedures Programming Manual, publication 1756-PM001 | Provides information on programming Logix5000 controllers, including managing project files, organizing tags, programming and testing routines, and handling faults |
| Logix5000 Controllers Add-On Instructions Programming Manual, publication 1756-PM010 | Provides information on using creating and using standard and safety Add-On Instructions in Logix applications |
| ControlLogix System User Manual, publication 1756-UM001 | Provides information on using ControlLogix controllers in non-safety applications |
| DeviceNet Modules in Logix5000 Control Systems User Manual, publication DNET-UM004 | Provides information on using the 1756-DNB module in a Logix5000 control system |
| EtherNet/IP Modules in Logix5000 Control Systems User Manual, publication ENET-UM001 | Provides information on using the 1756-ENBT module in a Logix5000 control system |
| ControlNet Modules in Logix5000 Control Systems User Manual, publication CNET-UM001 | Provides information on using the 1756-CNB module in Logix5000 control systems |
| Logix5000 Controllers Execution Time and Memory Use Reference Manual, publication 1756-RM087 | Provides information on estimating the execution time and memory use for instructions |
| Logix Import Export Reference Manual, publication 1756-RM084 | Provides information on using the RSLogix 5000 Import/Export utility |
| Industrial Automation Wiring and Grounding Guidelines, publication 1770-4.1 | Provides general guidelines for installing a Rockwell Automation industrial system. |
| Product Certifications website, http://www.ab.com | Provides declarations of conformity, certificates, and other certification details |

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.

| | | |
|---|--|----|
| | Preface | |
| | Understanding Terminology | 9 |
| | Additional Resources | 10 |
| | Chapter 1 | |
| Safety Integrity Level (SIL) Concept | SIL 3 Certification | 11 |
| | Functional Verification Tests | 12 |
| | GuardLogix Architecture for SIL 3 Applications | 13 |
| | GuardLogix System Components | 14 |
| | GuardLogix Certifications | 16 |
| | GuardLogix PFD and PFH Specifications | 16 |
| | Safety Integrity Level (SIL) Compliance Distribution and Weight. | 17 |
| | System Reaction Time | 18 |
| | Safety Task Reaction Time | 18 |
| | Safety Task Period and Safety Task Watchdog | 19 |
| | Contact Information if Device Failure Occurs | 19 |
| | Chapter 2 | |
| GuardLogix Controller System | 1756 GuardLogix Controller Hardware | 21 |
| | Primary Controller | 22 |
| | Safety Partner | 22 |
| | Chassis | 22 |
| | Power Supplies | 22 |
| | 1768 Compact GuardLogix Controller Hardware | 23 |
| | CIP Safety Protocol | 23 |
| | Safety I/O | 23 |
| | Communication Bridges | 24 |
| | Programming Overview | 26 |
| | Chapter 3 | |
| CIP Safety I/O for the GuardLogix Control System | Overview | 27 |
| | Typical Safety Functions of CIP Safety I/O Modules | 27 |
| | Diagnostics | 28 |
| | Status Data | 28 |
| | Status Indicators | 28 |
| | On- or Off-delay Function | 28 |
| | Reaction Time | 28 |
| | Safety Considerations for CIP Safety I/O Modules | 29 |
| | Ownership | 29 |
| | Safety I/O Configuration Signature | 29 |
| | I/O Module Replacement | 29 |

| | | |
|---|--|---|
| | Chapter 4 | |
| CIP Safety and the Safety Network Number | Routable CIP Safety Control System..... | 33 |
| | Unique Node Reference..... | 34 |
| | Safety Network Number..... | 34 |
| | Considerations for Assigning the Safety Network Number (SNN) ... | 35 |
| | Safety Network Number (SNN) for Safety Consumed Tags..... | 35 |
| | Safety Network Number (SNN) for Out-of-box Modules..... | 36 |
| | Safety Network Number (SNN) for Safety Module with a Different Configuration Owner | 36 |
| | | Safety Network Number (SNN) When Copying a Safety Project . |
| | | 36 |
| | Chapter 5 | |
| Characteristics of Safety Tags, the Safety Task, and Safety Programs | Differentiate Between Standard and Safety | 37 |
| | SIL 2 Safety Applications | 38 |
| | SIL 2 Safety Control in the Safety Task..... | 38 |
| | SIL 2 Safety Control in Standard Tasks (1756 GuardLogix controllers only)..... | 40 |
| | SIL 3 Safety—the Safety Task | 41 |
| | Safety Task Limitations | 41 |
| | Safety Task Execution Details | 42 |
| | Use of Human-to-machine Interfaces | 43 |
| | Precautions | 43 |
| | Accessing Safety-related Systems | 44 |
| | Safety Programs | 45 |
| | Safety Routines | 45 |
| | Safety Tags | 46 |
| Standard Tags in Safety Routines (tag mapping)..... | 47 | |
| Additional Resources | 48 | |
| | Chapter 6 | |
| Safety Application Development | Safety Concept Assumptions..... | 49 |
| | Basics of Application Development and Testing | 50 |
| | Commissioning Life Cycle | 51 |
| | Specification of the Control Function..... | 52 |
| | Create the Project | 53 |
| | Test the Application Program | 53 |
| | Generate the Safety Task Signature | 53 |
| | Project Verification Test | 54 |
| | Confirm the Project | 55 |
| | Safety Validation | 56 |
| | Lock the GuardLogix Controller..... | 56 |
| | Downloading the Safety Application Program | 57 |
| | Uploading the Safety Application Program..... | 57 |
| | Online Editing..... | 57 |
| | Storing and Loading a Project from Nonvolatile Memory..... | 58 |
| Force Data..... | 58 | |

| | | |
|---|---|----|
| | Inhibit a Module | 58 |
| | Editing Your Safety Application | 59 |
| | Performing Offline Edits | 60 |
| | Performing Online Edits | 60 |
| | Modification Impact Test | 60 |
| | Chapter 7 | |
| Monitor Status and Handle Faults | Monitoring System Status | 63 |
| | CONNECTION_STATUS Data | 63 |
| | Input and Output Diagnostics | 64 |
| | I/O Module Connection Status | 64 |
| | De-energize to Trip System | 65 |
| | Get System Value (GSV) and Set System Value (SSV) Instructions | 65 |
| | GuardLogix System Faults | 66 |
| | Nonrecoverable Controller Faults | 66 |
| | Nonrecoverable Safety Faults | 66 |
| | Recoverable Faults | 67 |
| | Appendix A | |
| Safety Instructions | Safety Application Instructions | 69 |
| | Metal Form Safety Application Instructions | 70 |
| | Safety Instructions | 71 |
| | Additional Resources | 72 |
| | Appendix B | |
| Safety Add-On Instructions | Creating and Using a Safety Add-On Instruction | 73 |
| | Create Add-On Instruction Test Project | 75 |
| | Create a Safety Add-On Instruction | 75 |
| | Generate Instruction Signature | 75 |
| | Download and Generate Safety Instruction Signature | 76 |
| | SIL 3 Add-On Instruction Qualification Test | 76 |
| | Confirm the Project | 76 |
| | Safety Validate Add-On Instructions | 76 |
| | Create Signature History Entry | 77 |
| | Export and Import the Safety Add-On Instruction | 77 |
| | Verify Safety Add-On Instruction Signatures | 77 |
| | Test the Application Program | 77 |
| | Project Verification Test | 78 |
| | Safety Validate Project | 78 |
| | Additional Resources | 78 |

| | |
|---|--|
| Reaction Times | <p>Appendix C</p> <p>System Reaction Time 79</p> <p>Logix System Reaction Time 79</p> <p> Simple Input-logic-output Chain 80</p> <p> Logic Chain Using Produced/Consumed Safety Tags..... 81</p> <p>Factors Affecting Logix Reaction-time Components 82</p> <p> Accessing Input Module Delay Time Settings..... 82</p> <p> Accessing Input and Output Safety Connection Reaction Time Limit 83</p> <p> Configuring the Safety Task Period and Watchdog..... 84</p> <p> Accessing Produced/Consumed Tag Data 85</p> <p>Additional Resources 86</p> |
| Checklists for GuardLogix Safety Applications | <p>Appendix D</p> <p>Checklist for GuardLogix Controller System 88</p> <p>Checklist for Safety Inputs 89</p> <p>Checklist for Safety Outputs 90</p> <p>Checklist for Developing a Safety Application Program..... 91</p> |
| GuardLogix Systems Safety Data | <p>Appendix E</p> <p>PFH Values 93</p> <p>PFH Values..... 94</p> |
| RSLogix 5000 Software, Version 14 and Later, Safety Application Instructions | <p>Appendix F</p> <p>De-energize to Trip System 95</p> <p>Use Connection Status Data to Initiate a Fault Programmatically 95</p> |
| Using 1794 FLEX I/O Modules and 1756 SIL 2 Inputs and Outputs with 1756 GuardLogix Controllers to Comply with EN 50156 | <p>Appendix G</p> <p>SIL 2 Dual-channel Inputs (standard side of 1756 GuardLogix controllers) 101</p> <p>SIL 2 Outputs Using SIL 3 Guard I/O Output Modules 103</p> <p>SIL 2 Outputs Using 1756 or 1794 SIL 2 Output Modules 103</p> <p>Safety Functions within the 1756 GuardLogix Safety Task..... 104</p> |
| Glossary | |
| Index | |

Safety Integrity Level (SIL) Concept

| Topic | Page |
|---|------|
| SIL 3 Certification | 11 |
| Functional Verification Tests | 12 |
| GuardLogix Architecture for SIL 3 Applications | 13 |
| GuardLogix System Components | 14 |
| GuardLogix Certifications | 17 |
| GuardLogix PFD and PFH Specifications | 17 |
| Safety Integrity Level (SIL) Compliance Distribution and Weight | 18 |
| System Reaction Time | 19 |
| Safety Task Period and Safety Task Watchdog | 20 |
| Contact Information if Device Failure Occurs | 20 |

SIL 3 Certification

1756 and 1768 GuardLogix controller systems are **type-approved** and certified for use in safety applications up to and including SIL CL3 according to IEC 61508 and IEC 62061, safety applications up to and including Performance Level PLe (Category 4) according to ISO 13849-1. SIL requirements are based on the standards current at the time of certification.

In addition, the standard tasks within 1756 GuardLogix controllers can be used either for standard applications or SIL 2 safety applications as described in the Using ControlLogix in SIL 2 Applications Reference Manual, publication [1756-RM001](#). In either case, do not use SIL 2 or standard tasks and variables to build up safety loops of a higher level. The safety task is the only task certified for SIL 3 applications.

The standard task in 1768 Compact GuardLogix controllers may not be used for SIL 2 safety applications.

RSLogix 5000® programming software is required to create programs for 1756 and 1768 GuardLogix controllers.

The TÜV Rheinland has approved GuardLogix controller systems for use in safety-related applications up to SIL CL 3, in which the de-energized state is considered to be the safe state. All of the examples related to I/O included in this manual are based on achieving de-energization as the safe state for typical Machine Safety and Emergency Shutdown (ESD) Systems.

IMPORTANT

The system user is responsible for:

- the setup, SIL rating, and validation of any sensors or actuators connected to the GuardLogix system.
 - project management and functional testing.
 - access control to the safety system, including password handling.
 - programming the application software and the device configurations in accordance with the information in this safety reference manual and the GuardLogix Controllers User Manual, publication [1756-UM020](#), or the 1768 Compact GuardLogix Controllers User Manual publication [1768-UM002](#).
-

When applying Functional Safety, restrict access to qualified, authorized personnel who are trained and experienced.

Functional Verification Tests

IEC 61508 requires the user to perform various functional verification tests of the equipment used in the system. Functional verification tests are performed at user-defined times. For example, functional verification test intervals can be once a year, once every 15 years, or whatever time frame is appropriate.

GuardLogix controllers have a functional verification test interval of up to 20 years. Other components of the system, such as safety I/O modules, sensors, and actuators may have shorter functional verification test intervals. The controller should be included in the functional verification testing of the other components in the safety system.

IMPORTANT

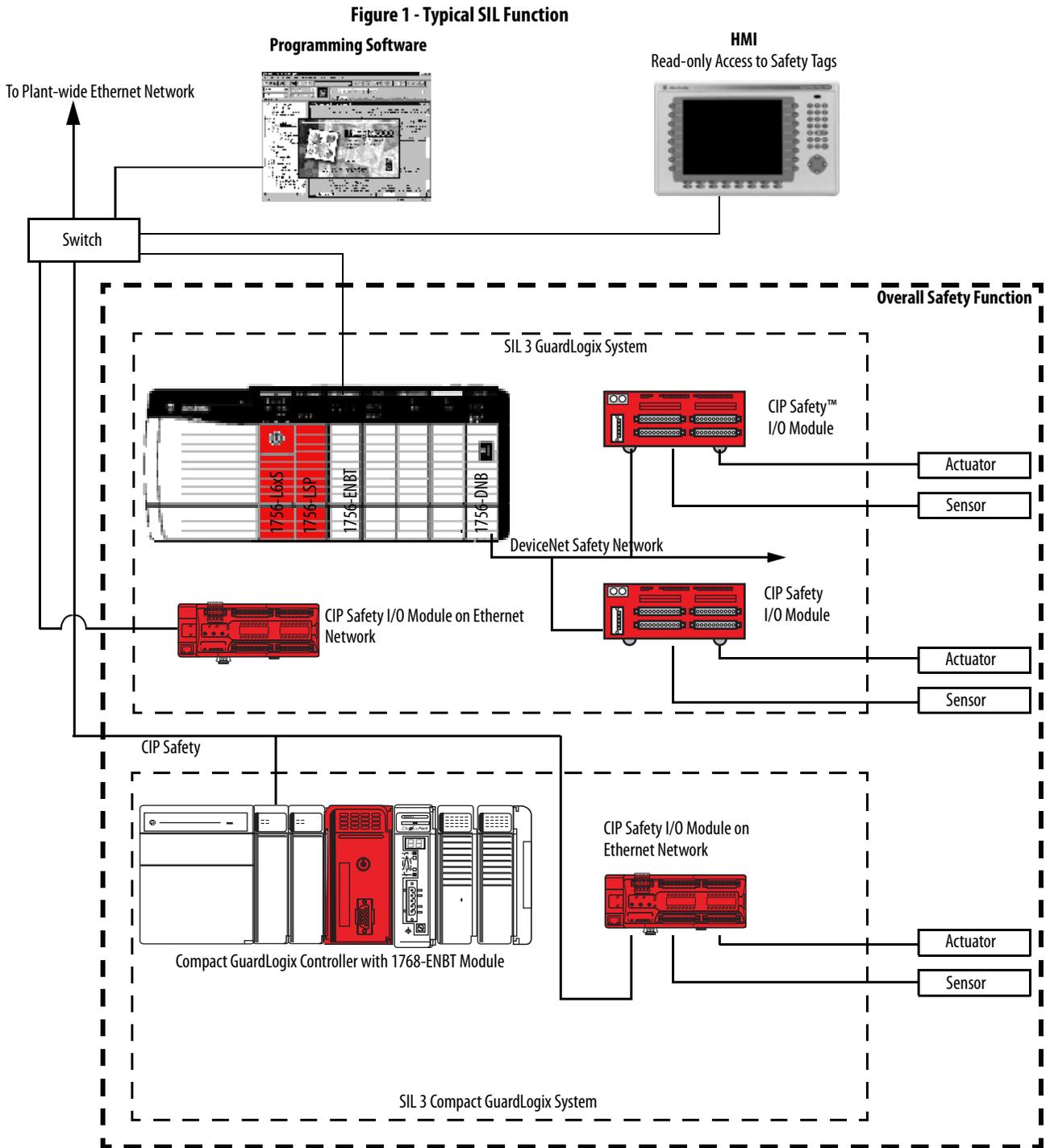
Your specific applications determine the time frame for the functional verification test interval. However, this is mainly related to safety I/O modules and field instrumentation.

For more information on the requirements of a functional verification test, see [Project Verification Test](#) on pages [56](#) and [56](#).

GuardLogix Architecture for SIL 3 Applications

The following illustration shows a typical SIL function, including:

- the overall safety function.
- the GuardLogix portion of the overall safety function.
- how other devices (for example, HMI) are connected, while operating outside the function.



GuardLogix System Components

The tables in this section list SIL 3-certified GuardLogix components for both 1756 and 1768 systems as well as non-SIL 3-certified components that may be used with SIL 3 GuardLogix systems.

For the most current list of GuardLogix controller and CIP Safety™ I/O modules certified series and firmware revisions, see <http://www.rockwellautomation.com/products/certification/safety/>. Firmware revisions are available at <http://support.rockwellautomation.com/ControlFLASH™/>.

Table 1 - SIL 3-certified GuardLogix Components

| Device Type | Cat. No. | Description | Related Documentation ⁽¹⁾ | |
|--|---|--|--------------------------------------|------------------------------|
| | | | Installation Instructions | User Manual |
| 1756 GuardLogix primary controller (ControlLogix556xS) | 1756-L61S | Controller with 2 MB standard, 1 MB safety memory | N/A ⁽²⁾ | 1756-UM020 |
| | 1756-L62S | Controller with 4 MB standard, 1 MB safety memory | | |
| | 1756-L63S | Controller with 8 MB standard, 3.75 MB safety memory | | |
| 1756 GuardLogix safety partner (ControlLogix55SP) | 1756-LSP | Safety partner (for L6xS) | | |
| 1756 GuardLogix primary controller (ControlLogix557xS) | 1756-L71S | Controller with 2 MB standard, 1 MB safety memory | N/A ⁽²⁾ | 1756-UM020 |
| | 1756-L72S | Controller with 4 MB standard, 2 MB safety memory | | |
| | 1756-L73S | Controller with 8 MB standard, 4 MB safety memory | | |
| | 1756-L73SXT | Controller (XT) with 8 MB standard, 4 MB safety memory | | |
| 1756 GuardLogix safety partner (ControlLogix557SP) | 1756-L7SP | Safety partner (for L7xS) | | |
| | 1756-L7SPXT | Safety partner (XT) (for L73SXT) | | |
| 1768 Compact GuardLogix Controller (CompactLogix4xS) | 1768-L43S | Controller with support for two 1768 modules | 1768-IN004 | 1768-UM002 |
| | 1768-L45S | Controller with support for four 1768 modules | | |
| CIP Safety I/O modules on DeviceNet networks | For the most current list of certified series and firmware revisions, see the safety certificate at http://www.rockwellautomation.com/products/certification/safety/ | | 1791DS-IN001 | 1791DS-UM001 |
| CIP Safety I/O modules on EtherNet/IP networks | | | 1791DS-IN002 | |
| | | | 1732DS-IN001 | |
| POINT Guard I/O modules | | | 1791ES-IN001 | 1791ES-UM001 |
| | | | N/A ⁽²⁾ | 1734-UM013 |

(1) These publications are available from Rockwell Automation by visiting <http://www.rockwellautomation.com/literature>.

(2) See user manual for installation instructions.

Table 2 - Components Suitable for Use with 1756 GuardLogix Controller Safety Systems

| Device Type | Cat. No. | Description | Series ⁽¹⁾ | Revision ⁽¹⁾ | Related Documentation ⁽⁴⁾ | |
|-----------------------|---|---|-----------------------|--------------------------------|--------------------------------------|----------------------|
| | | | | | Installation Instructions | User Manual |
| Chassis | 1756-A4 1756-A7 1756-A10 1756-A13 1756-A17 | 4-slot chassis 7-slot chassis 10-slot chassis 13-slot chassis 17-slot chassis | B | N/A | 1756-IN005 | N/A |
| | 1756-A4LXT 1756-A5XT 1756-A7XT 1756-A7LXT | 4-slot XT chassis 5-slot XT chassis 7-slot XT chassis 7-slot XT chassis | B | N/A | | N/A |
| Power supply | 1756-PA72 | Power supply, AC | C | N/A | 1756-IN005 | N/A |
| | 1756-PB72 | Power supply, DC | C | | | |
| | 1756-PA75 | Power supply, AC | B | | | |
| | 1756-PB75 | Power supply, DC | B | | | |
| | 1756-PAXT | XT power supply, AC | B | | | |
| | 1756-PBXT | XT power supply, DC | B | | | |
| Communication modules | 1756-ENBT 1756-EN2T 1756-EN2F 1756-EN2TR 1756-EN3TR | EtherNet/IP bridge module | A A A C C | 3.6 2.005 2.005 5.007 | ENET-IN002 | ENET-UM001 |
| | 1756-EN2TXT | XT EtherNet/IP bridge module (copper) | C | 5.007 | | |
| | 1734-AENT | POINT I/O™ Ethernet Adapter | A | 3.001 | 1734-IN590 | 1734-UM011 |
| | 1756-DNB | DeviceNet bridge module | A | 6.2 | DNET-IN001 | DNET-UM004 |
| | 1756-CN2 | ControlNet bridge module | A | 12.1 | CNET-IN005 | CNET-UM001 |
| | 1756-CN2R | ControlNet bridge module, redundant media | A | 12.1 | | |
| | 1756-CN2RXT | XT ControlNet bridge module, redundant media | B | 20.020 | | |
| Programming software | 9324-xxxx | RSLogix 5000 software for 1756-L6xS controllers | N/A | 14 ⁽²⁾ | N/A | Consult online help. |
| | | RSLogix 5000 software for 1756-L7xS and 1756-L73SXT controllers | | 20 ⁽³⁾ | | |
| Memory cards | 1784-CF128 | 128MB CompactFlash Card for 1756-L6xS controllers | N/A | N/A | N/A | N/A |
| | 1784-SD1 | 1 GB Secure Digital (SD) Card for 1756-L7xS controllers | | | | |
| | 1784-SD2 | 2 GB Secure Digital (SD) Card for 1756-L7xS controllers | | | | |

(1) This version or later.

(2) RSLogix 5000 software, version 15, does not support GuardLogix safety controllers (for 14, 16, 17, 18, 19, and 20).

(3) V20 only.

(4) These publications are available from Rockwell Automation by visiting <http://www.rockwellautomation.com/literature>.

Slots of a SIL 3 system chassis not used by the 1756 SIL 3 system may be populated with other ControlLogix® (1756) modules that are certified to the Low Voltage and EMC Directives.

IMPORTANT ControlLogix-XT™ system components are rated for extreme environmental conditions only when used properly with other Logix-XT system components. The use of ControlLogix-XT components with traditional ControlLogix or GuardLogix system components nullifies extreme environment ratings.

To find the certificates for the 'Programmable Control – ControlLogix Product Family' refer to <http://www.rockwellautomation.com/products/certification/ce/>.

Table 3 - Components Suitable for Use With 1768 Compact GuardLogix Controller Safety Systems

| Device Type | Cat. No. | Description | Series ⁽¹⁾ | Revision ⁽¹⁾ | Related Documentation ⁽²⁾ | |
|-----------------------|------------|----------------------------|-----------------------|-------------------------|--------------------------------------|----------------------------|
| | | | | | Installation Instructions | User Manual |
| Power supply | 1768-PA3 | Power supply, AC | N/A | N/A | 1768-IN001 | N/A |
| | 1768-PB3 | Power supply, DC | N/A | N/A | | |
| Communication modules | 1768-ENBT | EtherNet/IP bridge module | A | 3.1.1 | ENET-IN002 | ENET-UM001 |
| | 1734-AENT | POINT I/O Ethernet Adapter | A | 3.001 | 1734-IN590 | 1734-UM011 |
| | 1734-AENTR | POINT I/O Ethernet Adapter | A | 3.001 | 1734-IN040 | N/A |
| | 1768-CNB | ControlNet bridge module | A | 2.1.1 | CNET-IN005 | CNET-UM001 |
| Programming software | 9324-xxxx | RSLogix 5000 software | N/A | 18 | N/A | Consult online help. |
| Memory Cards | 1784-CF128 | 128 MB CompactFlash Card | N/A | N/A | N/A | N/A |

(1) This revision or version or later.

(2) These publications are available from Rockwell Automation by visiting <http://www.rockwellautomation.com/literature>.

Expansion slots of a SIL 3 system bus that are not used by the 1768 SIL 3 system may be populated with other CompactLogix™ (1768) modules that are certified to the Low Voltage and EMC Directives. To find the certificates for the ‘Programmable Control – CompactLogix Product Family’, refer to <http://www.rockwellautomation.com/products/certification/ce/>.

GuardLogix Certifications

The ControlLogix Controllers Technical Data, publication [1756-TD001](#) lists the product specifications and the agency certifications for which the products are approved. If a product has achieved agency certification, it is marked as such on the product labeling. See the Product Certification link at <http://www.rockwellautomation.com/products/certification/> for Declarations of Conformity, Certificates, and other certification details.

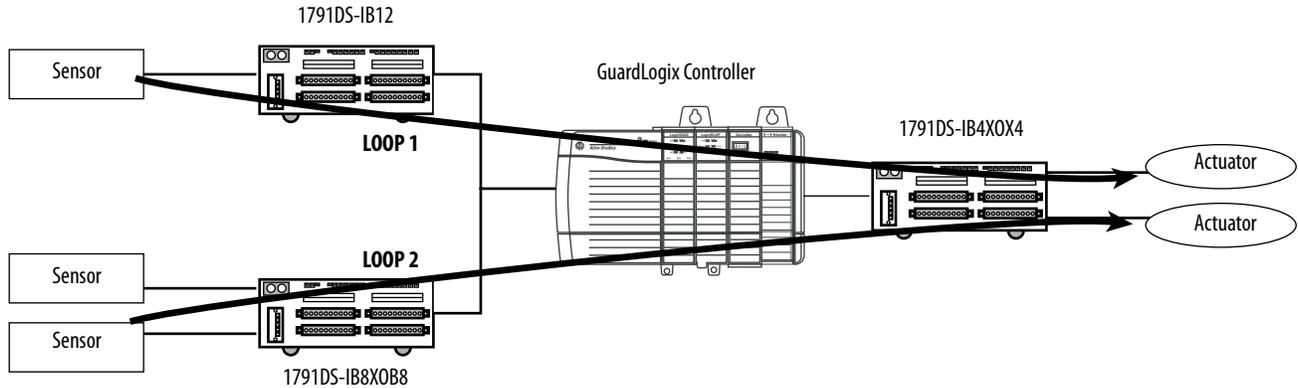
GuardLogix PFD and PFH Specifications

Safety-related systems can be classified as operating in either a Low Demand mode, or in a High Demand/Continuous mode. IEC 61508 quantifies this classification by stating that the frequency of demands for operation of the safety system is no greater than once per year in the Low Demand mode, or greater than once per year in High Demand/Continuous mode.

The Safety Integrity Level (SIL) value for a Low Demand safety-related system is directly related to order-of-magnitude ranges of its average probability of failure to satisfactorily perform its safety function on demand or, simply, probability of failure on demand (PFD). The SIL value for a High Demand/Continuous mode safety-related system is directly related to the probability of a dangerous failure occurring per hour (PFH).

PFH and PFH values are associated with each of the three primary elements making up a safety-related system (the sensors, the logic element, and the actuators). Within the logic element you also have input, processor, and output elements.

Figure 2 - PFH Example



To determine the logic element PFH for each safety loop in the simple example system shown in the PFH Example, sum the PFH values for each component in the loop. The [PFH Equations by Safety Loop](#) table provides a simplified example of PFH value calculations for each safety loop shown in the PFH Example illustration.

Table 4 - PFH Equations by Safety Loop

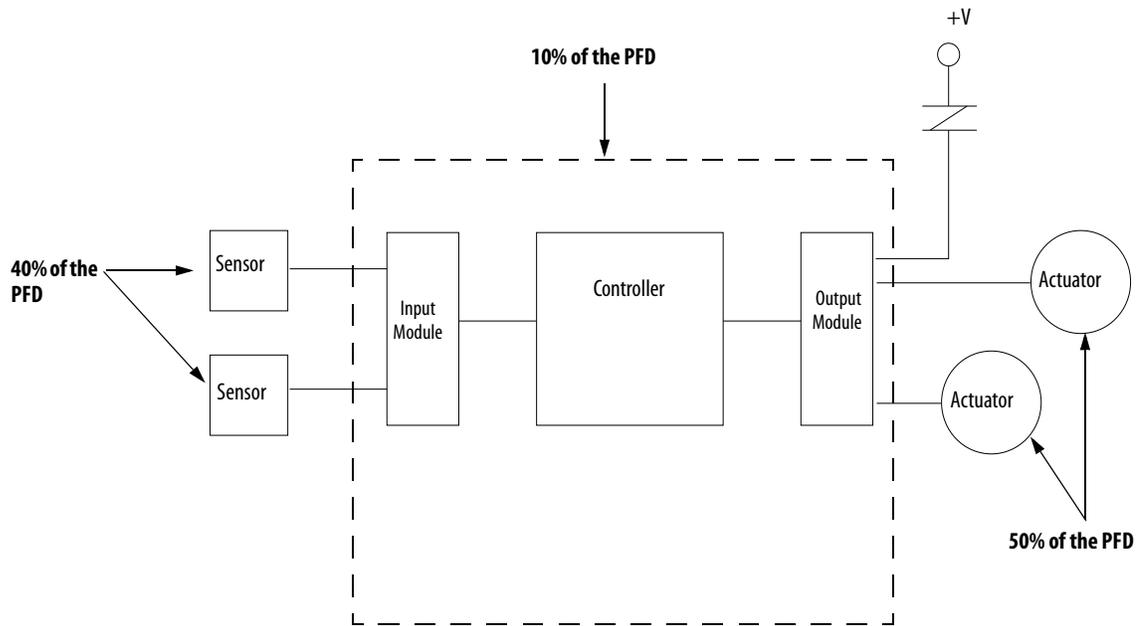
| For this loop | Sum the PFH values of these components |
|------------------------|---|
| Total PFH for loop 1 = | 1791DS-IB12 + GuardLogix controller + 1791DS-IB4XOX4 |
| Total PFH for loop 2 = | 1791DS-IB8XOB8 + GuardLogix controller + 1791DS-IB4XOX4 |

When calculating PFH values, you must take into account the specific requirements of your application, including proof test and diagnostic test intervals.

Safety Integrity Level (SIL) Compliance Distribution and Weight

The GuardLogix controller and I/O system may conservatively be assumed to contribute 10% of the reliability burden. A SIL 3 system may need to incorporate multiple inputs for critical sensors and input devices, as well as dual outputs connected in series to dual actuators dependent on SIL assessments for the safety related system.

Figure 3 - Reliability Burden



System Reaction Time

The system reaction time is the amount of time from a safety-related event as an input to the system until the system sets corresponding outputs to their safe state. Faults within the system can also have an effect upon the reaction time of the system. The system reaction time is the sum of the following reaction times.



Each of the times listed above is variably dependent on factors such as the type of I/O module and instructions used in the program.

Safety Task Reaction Time

The safety task reaction time is the worst-case delay from any input change presented to the controller until the processed output is set by the output producer. It is less than or equal to the sum of the safety task period and the safety task watchdog.

Safety Task Period and Safety Task Watchdog

The safety task period is the interval at which the safety task executes.

The safety task watchdog time is the maximum permissible time for safety task processing. If safety task processing time exceeds the safety task watchdog time, a non-recoverable safety fault occurs in the controller and outputs transition to the safe state (off) automatically.

You define the safety task watchdog time, which must be less than or equal to the safety task period.

The safety task watchdog time is set in the task properties window of RSLogix 5000 software. This value can be modified online, regardless of controller mode, but it cannot be changed when the controller is safety-locked or once a safety task signature is created.

Contact Information if Device Failure Occurs

If you experience a failure with any SIL 3-certified device, contact your local Rockwell Automation distributor to initiate the following actions:

- You can return the device to Rockwell Automation so the failure is appropriately logged for the catalog number affected and a record is made of the failure.
- You can request a failure analysis (if necessary) to try to determine the cause of the failure.

GuardLogix Controller System

| Topic | Page |
|---|------|
| 1756 GuardLogix Controller Hardware | 21 |
| 1768 Compact GuardLogix Controller Hardware | 23 |
| CIP Safety Protocol | 23 |
| Safety I/O | 23 |
| Communication Bridges | 24 |
| Programming Overview | 26 |

For a brief listing of components suitable for use in Safety Integrity Level (SIL) 3 applications, see the table on page 14. For more detailed and up-to-date information see <http://www.rockwellautomation.com/products/certification/safety/>.

When installing a GuardLogix controller, follow the information in the GuardLogix Controllers User Manual, publication [1756-UM020](#), or CompactLogix Controllers Installation Instructions, publication [1768-IN004](#).

1756 GuardLogix Controller Hardware

The 1756 GuardLogix controller consists of a primary controller (ControlLogix 556xS or ControlLogix 557xS) and a safety partner (ControlLogix 55SP or ControlLogix 557SP). These two modules work in a 1oo2 architecture to create the SIL 3-capable controller. They are described in the following sections.

Both the primary controller and safety partner perform power-up and run-time functional diagnostic tests of all safety-related components in the controller.

For details on status indicator operation, refer to the GuardLogix Controllers User Manual, publication [1756-UM020](#).

IMPORTANT

Status indicators are not reliable indicators for safety functions. They should be used only for general diagnostics during commissioning or troubleshooting. Do not attempt to use status indicators to determine operational status.

For a list of GuardLogix safety controller catalog numbers see [Table 1 on page 14](#). For a list of standard ControlLogix components suitable for safety applications, see [Table 2 on page 15](#).

Primary Controller

The primary controller is the processor that performs standard and safety control functions and communicates with the safety partner for safety-related functions in the GuardLogix control system. The primary controller consists of a central processor, I/O interface, and memory.

Safety Partner

To satisfy SIL 3 requirements, a safety partner must be installed in the slot immediately to the right of the primary controller. The safety partner is a co-processor that provides redundancy for safety-related functions in the system.

The safety partner is configured by the primary controller. Only a single download of the user program to the primary controller is required. The safety partner's operating mode is controlled by the primary controller.

Chassis

The chassis provides the physical connections between modules and the 1756 GuardLogix system. Any failure, though unlikely, would be detected as a failure by one or more of the active components of the system. Therefore, the chassis is not relevant to the safety discussion.

GuardLogix-XT™ controllers must use a ControlLogix-XT™ chassis to achieve the extreme environment rating.

Power Supplies

No extra configuration or wiring is required for SIL 3 operation of the ControlLogix power supplies. Any failure would be detected as a failure by one or more of the active components of the GuardLogix system. Therefore, the power supply is not relevant to the safety discussion.

GuardLogix-XT controllers must use a ControlLogix-XT power supply to achieve the extreme environment rating.

1768 Compact GuardLogix Controller Hardware

The 1768 Compact GuardLogix controllers combine the primary and safety partner controllers in a single controller hardware package to form a SIL 3 capable controller. Compact GuardLogix controllers feature a 1768 backplane and a 1769 backplane to support standard 1769 I/O modules.

Table 5 - Compact GuardLogix Controller Catalog Numbers

| Controller | Maximum 1768 Modules (local) | Maximum 1769 I/O Modules (local and remote) |
|------------|------------------------------|---|
| 1768-L435 | 2 | 16 |
| 1768-L455 | 4 | 30 |

The 1768 Compact GuardLogix controller is powered by a 1768-PA3 or 1768-PB3 power supply. A 1769-ECR end cap is also required.

For a list of 1768 Compact GuardLogix safety controllers and standard CompactLogix components suitable for safety applications, see [GuardLogix System Components on page 14](#).

CIP Safety Protocol

Safety-related communication between GuardLogix controllers takes place via produced and consumed safety tags. These safety tags use the CIP Safety protocol, which is designed to preserve data integrity during communication.

For more information on safety tags, see [Chapter 5, Characteristics of Safety Tags, the Safety Task, and Safety Programs](#).

Safety I/O

For information on CIP Safety I/O modules for use with GuardLogix controllers, see [Chapter 3](#).

Communication Bridges

Table 6 lists the communication interface modules available to facilitate communication over EtherNet/IP™, DeviceNet, and ControlNet networks via the CIP Safety protocol.

Table 6 - Communication Interface Modules by System

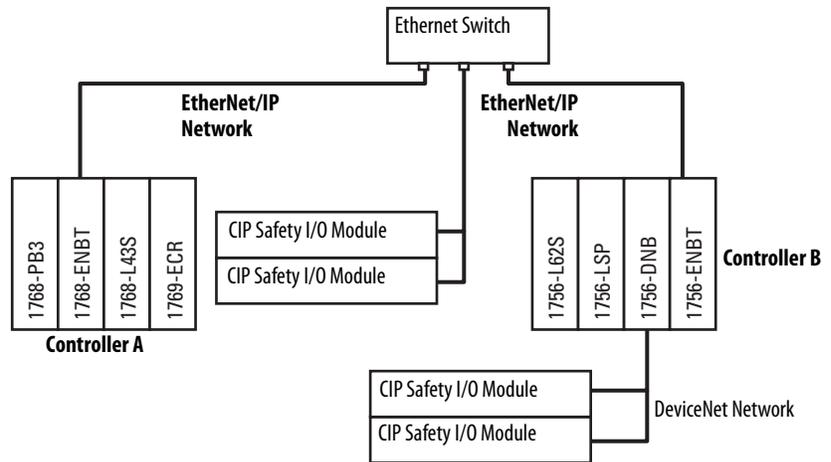
| GuardLogix System | Communication Modules |
|-------------------|--|
| 1756 | <ul style="list-style-type: none"> 1756-ENBT, 1756-EN2T(R), 1756-EN2F, or 1756-EN3TR EtherNet/IP bridge module 1734-AENT POINT I/O Ethernet Adapter 1756-DNB DeviceNet bridge module 1756-CN2 ControlNet bridge module 1756-CN2R Redundant ControlNet bridge module |
| 1756-XT | <ul style="list-style-type: none"> 1756-EN2TXT XT EtherNet/IP bridge module (copper) 1756-CN2RXT Redundant XT ControlNet bridge module |
| 1768 | <ul style="list-style-type: none"> 1768-ENBT 1734-AENT POINT I/O Ethernet Adapter 1768-CNB 1768-CNBR |

IMPORTANT Due to the design of the CIP Safety control system, CIP safety bridge devices, like those listed in the table, are not required to be SIL 3-certified.

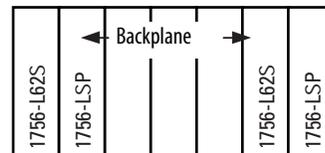
EtherNet/IP Network

Peer-to-peer safety communication between GuardLogix controllers is possible via the EtherNet/IP network through the use of EtherNet/IP bridge modules. An EtherNet/IP bridge module lets the GuardLogix controller control and exchange safety data with CIP Safety I/O modules on an EtherNet/IP network.

Figure 4 - Peer-to-peer Communication via EtherNet/IP Bridge Modules and the EtherNet/IP Network



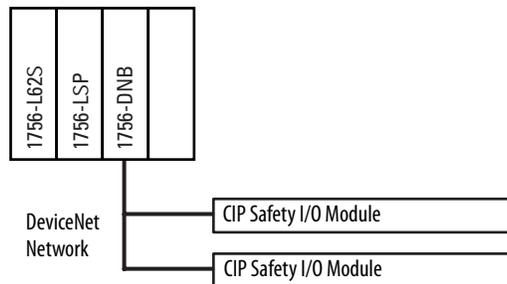
TIP Peer-to-peer safety communication between two 1756 GuardLogix controllers in the same chassis is also possible via the backplane.



DeviceNet Safety Network

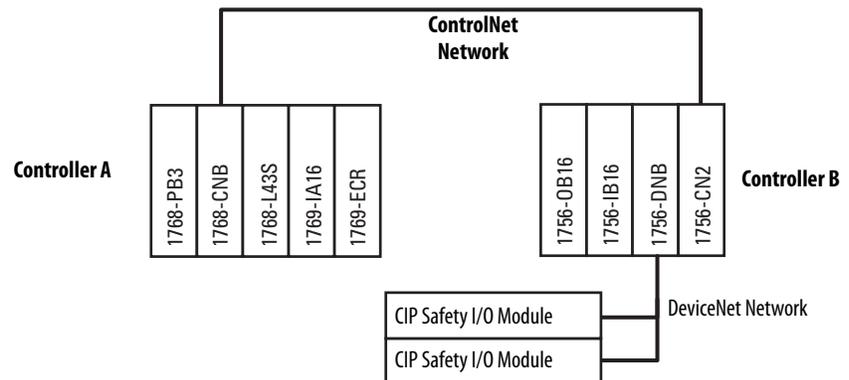
DeviceNet bridge modules let the 1756 GuardLogix controller control and exchange safety data with CIP Safety I/O modules on a DeviceNet network.

Figure 5 - Communication via a DeviceNet Bridge Module

*ControlNet Network*

ControlNet bridge modules let the GuardLogix controller produce and consume safety tags over ControlNet networks to other GuardLogix controllers or remote CIP Safety I/O networks.

Figure 6 - Communication via a ControlNet Bridge Module



Programming Overview

The programming software for the GuardLogix controller is RSLogix 5000 software.

RSLogix 5000 software is used to define the location, ownership, and configuration of I/O modules and controllers. The software is also used to create, test, and debug application logic. Initially, only relay ladder logic is supported in the GuardLogix safety task.

See [Appendix A](#) for information on the set of logic instructions available for safety applications.

Authorized personnel may change an application program, but only by using one of the processes described in [Editing Your Safety Application](#) on page [61](#).

IMPORTANT

When the GuardLogix controller is in the Run or Programming mode and you have not validated the application, you are responsible for maintaining safe conditions.

CIP Safety I/O for the GuardLogix Control System

| Topic | Page |
|--|------|
| Overview | 27 |
| Typical Safety Functions of CIP Safety I/O Modules | 27 |
| Reaction Time | 28 |
| Safety Considerations for CIP Safety I/O Modules | 29 |

Overview

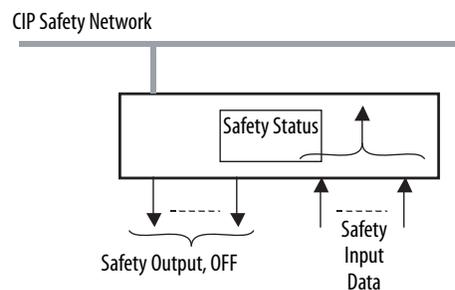
Before operating a GuardLogix safety system containing CIP Safety I/O modules, you must read, understand, and follow the installation, operation, and safety information provided in the publications listed in the [SIL 3-certified GuardLogix Components](#) tables on page 14.

CIP Safety I/O modules can be connected to safety input and output devices, allowing these devices to be monitored and controlled by the GuardLogix controller. For safety data, I/O communication is performed through safety connections using the CIP Safety protocol; safety logic is processed in the GuardLogix controller.

Typical Safety Functions of CIP Safety I/O Modules

The following is treated as the safe state by CIP Safety I/O modules:

- Safety outputs: OFF
- Safety input data to controller: OFF



The CIP Safety I/O modules should be used for applications that are in the safe state when the safety output turns OFF.

Diagnostics

CIP Safety I/O modules perform self-diagnostics when the power is turned ON and periodically during operation. If a diagnostic failure is detected, safety input data (to the controller) and local safety outputs are set to their safe state (OFF).

Status Data

In addition to safety input and output data, CIP Safety I/O modules support status data to monitor module and I/O circuit health. Refer to your module's product documentation for specific product capabilities.

Status Indicators

The CIP Safety I/O modules include status indicators. For details on status indicator operation, refer to the product documentation for your specific module.

On- or Off-delay Function

Some CIP Safety I/O modules may support On-delay and Off-delay functions for input signals. Depending upon your application, you may need to include Off-delay, On-delay, or both when calculating system reaction time. For example, the On to Off delay filter is helpful to filter out the pulse test from safety devices that affect the input logic level.

See [Appendix C](#) for information on system reaction time.

Reaction Time

The input reaction time is the time from when the signal changes on an input terminal to when safety data is sent to the GuardLogix controller.

The output reaction time is the time from when safety data is received from the GuardLogix controller to when the output terminal changes state.

For information on determining the input and output reaction times, refer to the product documentation for your specific CIP Safety I/O module.

See [Appendix C](#) for information on calculating the system reaction time.

Safety Considerations for CIP Safety I/O Modules

You must commission all devices with a node or IP address and communication rate, if necessary, before their installation on a safety network.

Ownership

Each CIP Safety I/O module in a GuardLogix system is owned by one GuardLogix controller. Multiple GuardLogix controllers and multiple CIP Safety I/O modules can be used without restrictions in chassis or on networks as needed. When a controller owns an I/O module, it stores the module's configuration data, as defined by the user. This controls how the modules operate in the system.

From a control standpoint, safety output modules can only be controlled by one controller. Each safety input module is also owned by a single controller; however, safety input data can be shared (consumed) by multiple GuardLogix controllers.

Safety I/O Configuration Signature

The configuration signature defines the module's configuration. It can be read and monitored. The configuration signature is used to uniquely identify a module's configuration. When using a GuardLogix controller, you do not have to monitor this signature. It is monitored automatically by the GuardLogix controller. If the configuration signature changes unexpectedly, the CIP safety connection between controller and I/O module is broken.

I/O Module Replacement

The replacement of safety devices requires that the replacement device be configured properly and that the replacement device's operation be user-verified.

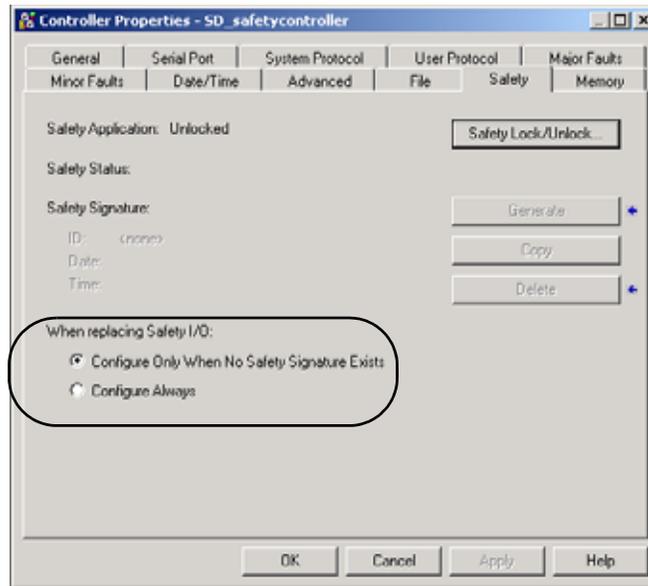


ATTENTION: During replacement or functional testing of a module, the safety of the system must not rely on any portion of the affected module.

Two options for I/O module replacement are available on the Safety tab of the Controller Properties dialog box in RSLogix 5000 software:

- Configure Only When No Safety Signature Exists
- Configure Always

Figure 7 - Safety I/O Replacement Options



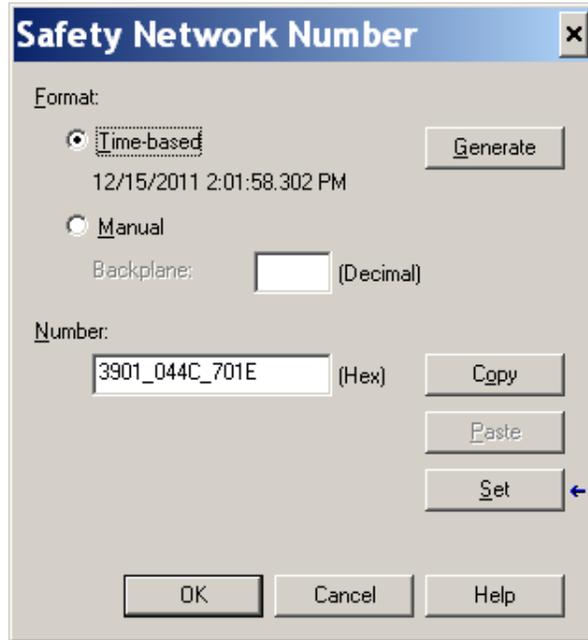
Configure Only When No Safety Signature Exists

This setting instructs the GuardLogix controller to automatically configure a safety module only when the safety task does not have a safety task signature, and the replacement module is in an out-of-box condition, meaning that a safety network number does not exist in the safety module.

If the safety task has a safety task signature, the GuardLogix controller only automatically configures the replacement CIP Safety I/O module if the following is true:

- The module already has the correct safety network number.
- The module electronic keying is correct.
- The node or IP address is correct.

To set the proper SNN when a safety signature exists, a manual action (typically SET), is required to download the proper SNN, after which the remainder of the configuration is automatically downloaded.



Configure Always

The GuardLogix controller will always attempt to automatically configure a replacement CIP Safety I/O module if the module is in an out-of-box condition, meaning that a safety network number does not exist in the replacement safety module, and the node number and I/O module keying matches the controller's configuration.



ATTENTION: Enable the Configure Always feature only if the entire routable CIP Safety control system is not being relied on to maintain SIL 3 behavior during the replacement and functional testing of a module.

If other parts of the CIP Safety control system are being relied upon to maintain SIL 3, make sure that the controller's Configure Always feature is disabled.

It is your responsibility to implement a process to make sure proper safety functionality is maintained during device replacement.



ATTENTION: Do not place any modules in the out-of-box condition on any CIP Safety network when the Configure Always feature is enabled, except while following the module replacement procedure in the GuardLogix Controllers User Manual, publication [1756-UM020](#), or the 1768 Compact GuardLogix Controllers User Manual, publication [1768-UM002](#).

Notes:

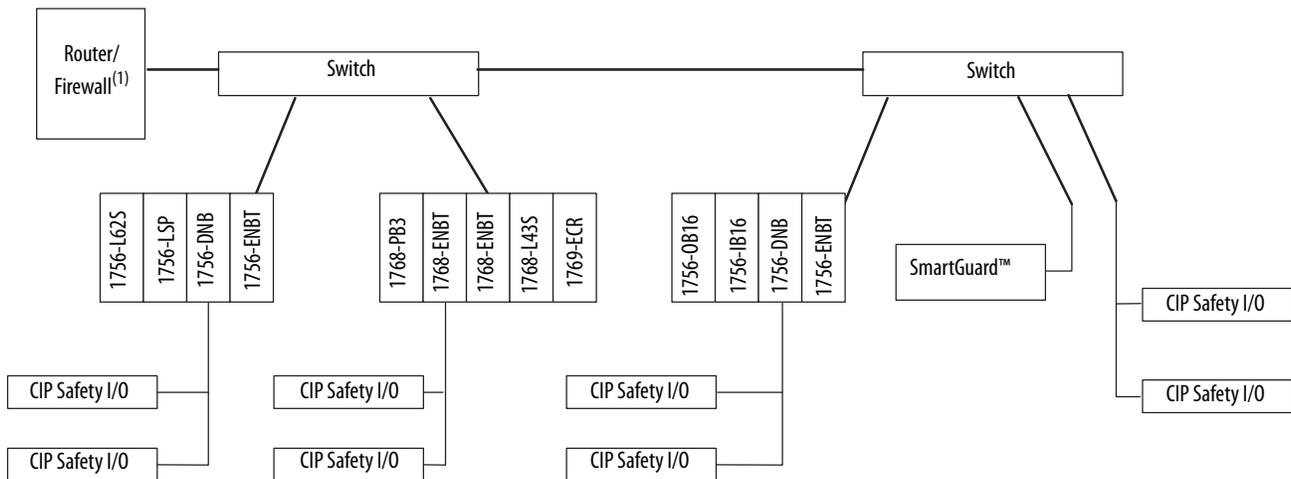
CIP Safety and the Safety Network Number

| Topic | Page |
|--|------|
| Routeable CIP Safety Control System | 33 |
| Considerations for Assigning the Safety Network Number (SNN) | 35 |

Routeable CIP Safety Control System

To understand the safety requirements of a CIP Safety control system, including the safety network number (SNN), you must first understand how communication is routeable in CIP control systems. The CIP Safety control system represents a set of interconnected CIP Safety devices. The routeable system represents the extent of potential mis-routing of packets from an originator to a target within the CIP Safety control system. The system is isolated such that there are no other connections into the system. For example, because the system below cannot be interconnected to another CIP Safety system through a larger, plant-wide Ethernet backbone, it illustrates the extent of a routeable CIP Safety system.

Figure 8 - CIP Safety System Example



(1) The router or firewall is set up to limit traffic.

Unique Node Reference

The CIP Safety protocol is an end-node to end-node safety protocol. The CIP Safety protocol allows the routing of CIP Safety messages to and from CIP Safety devices through non-certified bridges, switches, and routers.

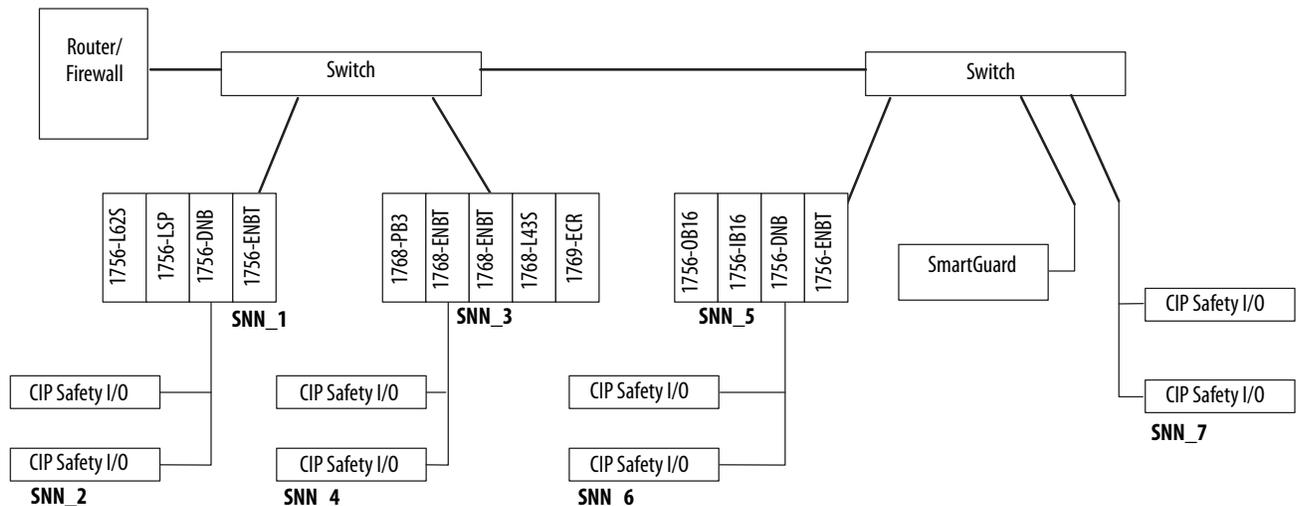
To prevent errors in non-certified bridges, switches, or routers from becoming dangerous, each end node within a routable CIP Safety control system must have a unique node reference. The unique node reference is a combination of a safety network number (SNN) and the node address of the node.

Safety Network Number

The safety network number (SNN) is assigned by software or by the user. Each CIP Safety network that contains Safety I/O nodes must have at least one unique SNN. Each ControlBus™ chassis that contains one or more safety devices must have at least one unique SNN. Safety network numbers assigned to each safety network or network subnet must be unique.

TIP More than one SNN can be assigned to a CIP Safety subnet or a ControlBus chassis that contains more than one safety device. However, for simplicity, we recommend that each CIP Safety subnet have one, and only one, unique SNN. This is also the case for each ControlBus chassis.

Figure 9 - CIP Safety Example with More Than One SNN

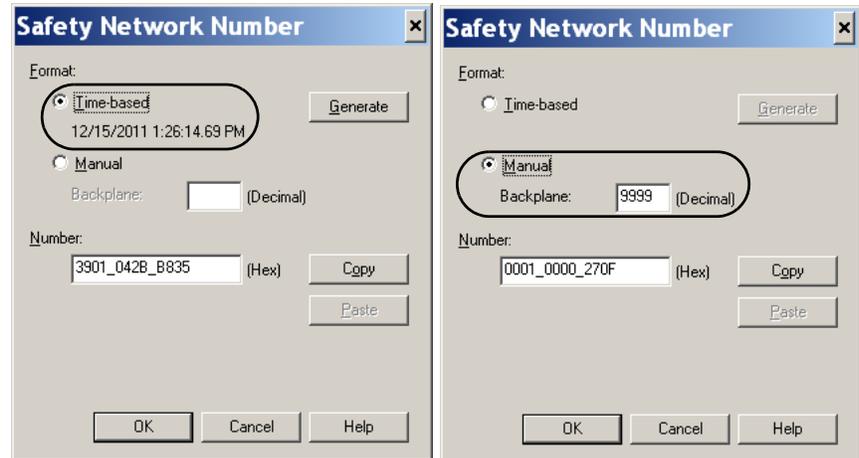


Each CIP Safety device must be configured with an SNN. Any device that originates a safety connection to another safety device must be configured with the SNN of the target device. If the CIP Safety system is in the start-up process prior to the functional safety testing of the system, the originating device may be used to set the unique node reference into the device.

The SNN used by the system is a 6-byte hexadecimal number. The SNN can be set and viewed in one of two formats: time-based or manual. When the time-

based format is selected, the SNN represents a localized date and time. When the manual format is selected, the SNN represents a network type and a decimal value from 1...9999.

Figure 10 - SNN Formats



The assignment of a time-based SNN is automatic when creating a new GuardLogix safety controller project and adding new Safety I/O modules.

Manual manipulation of an SNN is required in the following situations:

- If safety consumed tags are used.
- If the project will consume safety input data from a module whose configuration is owned by some other safety device
- If a safety project is copied to a different hardware installation within the same routable CIP Safety system

IMPORTANT

If you assign an SNN manually, take care to make sure that system expansion does not result in duplication of SNN and node address combinations.

Considerations for Assigning the Safety Network Number (SNN)

The assignment of the SNN is dependent upon factors including the configuration of the controller or CIP Safety I/O module.

Safety Network Number (SNN) for Safety Consumed Tags

When a safety controller that contains produced safety tags is added to the I/O Configuration tree, the SNN of the producing controller must be entered. The SNN may be copied from the producing controller's project and pasted into the new controller being added to the I/O Configuration tree.

Refer to the GuardLogix Controllers User Manual, publication [1756-UM020](#), or the 1768 Compact GuardLogix Controllers User Manual, publication [1768-UM002](#), for information on how to copy and paste an SNN.

Safety Network Number (SNN) for Out-of-box Modules

Out-of-box CIP Safety I/O modules do not have an SNN. The SNN is set when a configuration is sent to the module by the GuardLogix controller that owns the module.

IMPORTANT

To add a CIP Safety I/O module to a configured GuardLogix system (the SNN is present in the GuardLogix controller), the replacement CIP Safety module should have the correct SNN applied before it is added to the CIP Safety network. See page 30 for additional information.

Safety Network Number (SNN) for Safety Module with a Different Configuration Owner

When a CIP Safety I/O module is owned by a different GuardLogix controller (controller B), and then is added to another GuardLogix project (controller A project), RSLogix 5000 software assigns the SNN based on the current project. Because the current project (controller A project) is not the true configuration owner, you need to copy the original SNN (controller B project) into the configuration in controller A's project. This is easy to do with standard copy and paste commands. The result is that the CIP Safety I/O module produces data to two GuardLogix controllers at the same time. You can do this for a maximum of 16 controllers.

Refer to the GuardLogix Controllers User Manual, publication [1756-UM020](#), or the 1768 Compact GuardLogix Controllers User Manual, publication [1768-UM002](#), for information on changing, copying, and pasting safety network numbers.

Safety Network Number (SNN) When Copying a Safety Project



ATTENTION: If a safety project is copied for use in another project with different hardware or in a different physical location, and the new project is within the same routable CIP Safety system, every SNN must be changed in the second system. SNN values must not be repeated.

Refer to the GuardLogix Controllers User Manual, publication [1756-UM020](#), or the 1768 Compact GuardLogix Controllers User Manual, publication [1768-UM002](#), for information on changing the SNN.

Characteristics of Safety Tags, the Safety Task, and Safety Programs

| Topic | Page |
|---|------|
| Differentiate Between Standard and Safety | 37 |
| SIL 2 Safety Applications | 38 |
| SIL 3 Safety—the Safety Task | 41 |
| Use of Human-to-machine Interfaces | 44 |
| Safety Programs | 46 |
| Safety Routines | 46 |
| Safety Tags | 47 |
| Additional Resources | 49 |

Differentiate Between Standard and Safety

Because it is a Logix-series controller, both standard (non-safety-related) and safety-related components can be used in the GuardLogix control system.

You can perform standard automation control from standard tasks within a GuardLogix project. 1756 GuardLogix controllers provide the same functionality as other 1756 ControlLogix series controllers. 1768 Compact GuardLogix controllers provide the same functionality as other 1768-L4x CompactLogix controllers. What differentiates 1756 and 1768 GuardLogix controllers from standard controllers is that they provide a SIL 3-capable safety task.

However, a logical and visible distinction is required between the standard and safety-related portions of the application. RSLogix 5000 software provides this differentiation via the safety task, safety programs, safety routines, safety tags, and safety I/O modules. You can implement both SIL 2 and SIL 3 levels of safety control with the safety task of the GuardLogix controller.

SIL 2 Safety Applications

You can perform SIL 2 safety control by using the 1756 or 1768 GuardLogix controller's safety task.

Because 1756 GuardLogix controllers are part of the ControlLogix series of processors, you can perform SIL 2 safety control with a 1756 GuardLogix controller by using standard tasks or the safety task. This capability provides unique and versatile safety control options, as most applications have a higher percentage of SIL 2 safety functions than SIL 3 safety functions.

SIL 2 Safety Control in the Safety Task

The 1756 and 1768 GuardLogix safety task can be used to provide SIL 2 as well as SIL 3 safety functions. If SIL 3 safety functions need to be performed at the same time as SIL 2 safety functions, you must fulfill the requirements defined in the [SIL 3 Safety—the Safety Task, Safety Programs](#), and [Safety Routines](#) sections of this chapter, as well as the SIL 2 requirements listed in this section.

SIL 2 Safety Logic

From a GuardLogix safety control perspective, the biggest difference between SIL 2 and SIL 3 safety-rated devices is that SIL 2 is generally single-channel, while SIL 3 is typically dual-channel. When using Guard safety-rated I/O (red modules), which is required in the safety task, SIL 2 safety inputs can be single channel which can reduce complexity and the number of modules that are necessary.

It is up to the safety system designer to properly implement all safety functions. Consideration must be given to:

- field device selection (proper selection, identify and mitigate all device faults)
- consider safety demand requirements (low IEC 61511 or high ISO 13849)
- consider test intervals (diagnostics and proof testing needed to satisfy application requirements)
- identify, and justify with proper documentation any fault exclusions that are used

IMPORTANT

If a combination of SIL 2 and SIL 3 safety functions are used at the same time within the safety task, you must prevent SIL 2 input signals from directly controlling SIL 3 safety functions. This can be done by using specific safety task programs or routines to separate SIL 2 and SIL 3 safety functions.

Within the safety task, RSLogix 5000 software includes a set of safety-related ladder-logic instructions. In addition to these safety-rated ladder logic instructions, GuardLogix controllers feature application-specific SIL 3-rated safety instructions. All of these logic instructions may be used in Cat 1...4 and SIL 1...3 safety functions.

For SIL 2-only safety, a safety task signature is not required. However, if any SIL 3 safety functions are used within the safety task, a safety task signature is required.

Safety-locking the safety task once testing is completed is recommended for SIL 2 applications. Locking the safety task enables additional security features. You may also use FactoryTalk® Security and RSLogix 5000 routine source protection to limit access to safety-related logic.

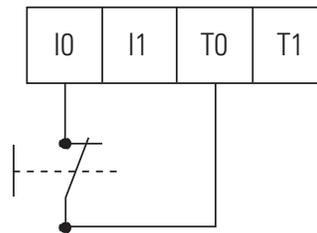
For more information on generating a safety task signature and safety-locking the safety task, refer to the GuardLogix Controllers User Manual, publication [1756-UM020](#), or the Compact GuardLogix Controllers User Manual, publication [1768-UM002](#).

SIL 2 Safety Inputs

CompactBlock™ Guard I/O™ (1791-series), ArmorBlock® Guard I/O™ (1732-series), and POINT Guard I/O™ (1734-series) safety input modules support single-channel SIL 2 safety input circuits. Because these modules are also rated for SIL 3 operation, mixing SIL 2 and SIL 3 circuits on the same module is allowed, provided you follow these guidelines.

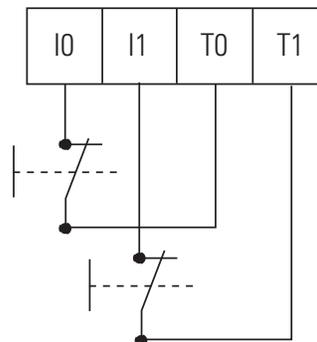
These two wiring examples show how to wire SIL 2 safety circuits to Guard I/O safety input modules. These examples make use of onboard test sources (T0...Tx) that are resident on all 1791 and 1732 safety input modules.

Figure 11 - Input Wiring



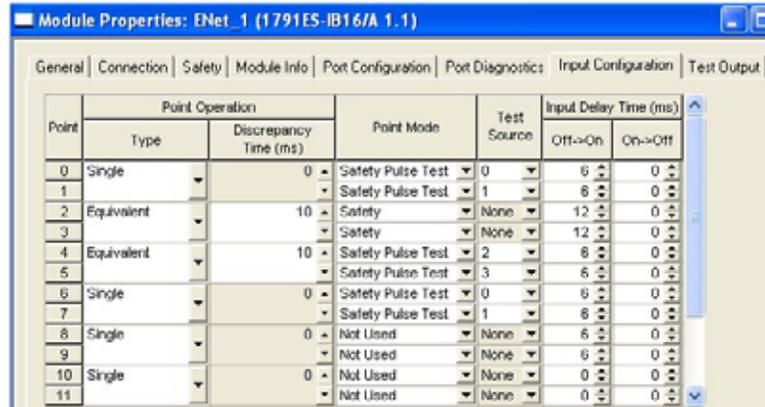
Guard I/O modules group inputs in pairs to facilitate Cat 3, Cat 4, and SIL 3 safety functions. For use in Cat 1, Cat 2, and SIL 2 safety functions, module inputs should still be used in pairs as illustrated. Two SIL 2 safety functions are shown wired to inputs I0 and I1 using test sources T0 and T1, respectively.

Figure 12 - Input Wiring in Pairs



For Cat 1, Cat 2, and SIL 2 safety functions, the Guard I/O safety modules need specific configurations within the GuardLogix project. In this example, inputs 0, 1, 6, 7, 8, 9, 10, and 11 are part of a CAT 1, 2 or SIL 2 safety function, because they are configured as Single. Inputs 2 and 3, as well as 4 and 5 are part of a CAT 3, CAT 4, or SIL 3 safety function, because they are configured as Equivalent.

Figure 13 - Input Configuration



| Field | Value |
|------------------|--|
| Type | Single |
| Discrepancy Time | N/A |
| Point Mode | Safety Pulse Test |
| Test Source | Set values based on how the field device is physically wired to the module. To make sure the test source is properly enabled, open and view settings on the Test Output tab. |
| Input Delay Time | User input based on field device characteristics. |

IMPORTANT The onboard pulse test outputs (TO...Tx) are typically used with field devices that have mechanical contacts. If a safety device that has electronic outputs is used (feeding safety inputs), they must have the appropriate safety ratings.

IMPORTANT If you are using GuardLogix Safety Application Instructions to detect discrepancy faults of dual-channel devices, be sure to configure your safety input modules as single, not equivalent or complementary. If configured for equivalent or complementary, discrepancy faults at the safety instruction are masked by the input module. These instructions provide all dual-channel functionality necessary for PLd (Cat. 3) or PLe (Cat. 4) safety functions. Either the instruction or the module can provide the discrepancy checking typically required for SIL3 safety functions.

Refer to the GuardLogix Safety Application Instruction Set Reference Manual, publication [1756-RM095](#).

SIL 2 Safety Control in Standard Tasks (1756 GuardLogix controllers only)

Because of the quality and amount of diagnostics built into the 1756 ControlLogix series of controllers, you can perform SIL 2 safety functions from within standard tasks. This is also true for 1756 GuardLogix controllers.

To perform SIL 2 safety control within a GuardLogix standard task, you must abide by requirements defined in the Using ControlLogix in SIL 2 Applications Safety Reference Manual, publication [1756-RM001](#).

IMPORTANT You may not use the standard task in a 1768 Compact GuardLogix controller for SIL-2 safety applications.

SIL 3 Safety—the Safety Task

Creation of a GuardLogix project automatically creates a single safety task. The safety task has these additional characteristics:

- GuardLogix controllers are the only controllers that support the safety task.
- The safety task cannot be deleted.
- GuardLogix controllers support a single safety task.
- Within the safety task, you can use multiple safety programs composed of multiple safety routines.
- You cannot schedule or execute standard routines from within the safety task.

The safety task is a periodic timed task with a user-selectable task priority and watchdog. In most cases, it is the controller's top priority and the user-defined program watchdog must be set to accommodate fluctuations in the execution of the safety task.

Safety Task Limitations

You specify both the safety task period and the safety task watchdog. The safety task period is the period at which the safety task executes. The safety task watchdog is the maximum time allowed from the start of safety task scheduled execution to its completion.

For more information on the safety task watchdog, see [Appendix C, Reaction Times](#).

The safety task period is limited to a maximum of 500 ms and cannot be modified online. Make sure that the safety task has enough time to finish before it is triggered again. Safety task watchdog timeout, a non-recoverable safety fault in the GuardLogix controller, occurs if the safety task is triggered while it is still executing from the previous trigger.

See [Chapter 7, Monitor Status and Handle Faults](#), for more information.

Safety Task Execution Details

The safety task executes in the same manner as standard periodic tasks, with the following exceptions:

- The safety task does not begin executing until the primary controller and safety partner have established their control partnership and the coordinated system time (CST) is synchronized. However, standard tasks begin executing as soon as the controller transitions to Run mode.
- Although the configurable range of the requested packet interval (RPI) for safety inputs and safety consumed tags is 6...500 ms, safety input tags and safety-consumed tags are updated only at the beginning of safety task execution. This means that even though the I/O RPI can be faster than the safety task period, the data does not change during safety task execution. The data is read only once at the beginning of the safety task execution.
- Safety input values are frozen at the start of safety task execution. As a result, timer-related instructions, such as TON and TOF, will not update during a single safety task execution. They will keep accurate time from one task execution to another, but the accumulated time will not change during safety task execution.



ATTENTION: This behavior differs from standard Logix task execution, but is similar to PLC or SLC™ behavior.

- For standard tags that are mapped to safety tags, the standard tag values are copied into safety memory at the start of the safety task and do not change during safety task execution.
- Safety output tag (output and produced) values are updated at the conclusion of safety task execution.

- The safety task responds to mode changes (for example, Run to Program or Program to Run) at timed intervals. As a result, the safety task may take more than one task period, but always less than two, to make a mode transition.

IMPORTANT

While safety-unlocked and without a safety task signature, the controller prevents simultaneous write access to safety memory from the safety task and communication commands. As a result, the safety task can be held off until a communication update completes. The time required for the update varies by tag size. Therefore, safety connection and/or safety watchdog timeouts could occur. (For example, if you make online edits when the safety task rate is set to 1 ms, a safety watchdog timeout could occur.)

To compensate for the hold-off time due to a communication update, add 2 ms to the safety watchdog time.

When the controller is safety-locked or a safety task signature exists, the situation described in this note cannot occur.

IMPORTANT

The safety task signature is required to operate in a SIL3 safety function.

Use of Human-to-machine Interfaces

Follow these precautions and guidelines for using HMI devices in SIL-rated GuardLogix systems.

Precautions

You must exercise precautions and implement specific techniques on HMI devices. These precautions include, but are not restricted to the following:

- Limited access and security
- Specifications, testing, and validation
- Restrictions on data and access
- Limits on data and parameters

For more information on how HMI devices fit into a typical SIL loop, see [Figure 1 on page 13](#).

Use sound techniques in the application software within the HMI and controller.

Accessing Safety-related Systems

HMI- related functions consist of two primary activities: reading and writing data.

Reading Parameters in Safety-related Systems

Reading data is unrestricted because reading doesn't affect the behavior of the safety system. However, the number, frequency, and size of the data being read can impact controller availability. To avoid safety-related nuisance trips, use good communication practices to limit the impact of communication processing on the controller. Do not set read rates to the fastest rate possible.

Changing Parameters in SIL-rated Systems

A parameter change in a safety-related loop via an external (that is, outside the safety loop) device (for example, an HMI) is allowed only with the following restrictions:

- Only authorized, specially-trained personnel (operators) can change the parameters in safety-related systems via HMIs.
- The operator who makes changes in a safety-related system via an HMI is responsible for the effect of those changes on the safety loop.
- You must clearly document variables that are to be changed.
- You must use a clear, comprehensive, and explicit operator procedure to make safety-related changes via an HMI.
- Changes can only be accepted in a safety-related system if the following sequence of events occurs:
 - a. The new variable must be sent twice to two different tags; that is, both values must not be written to with one command.
 - b. Safety-related code, executing in the controller, must check both tags for equivalency and make sure they are within range (boundary checks).
 - c. Both new variables must be read back and displayed on the HMI device.
 - d. Trained operators must visually check that both variables are the same and are the correct value.
 - e. Trained operators must manually acknowledge that the values are correct on the HMI screen that sends a command to the safety logic, which allows the new values to be used in the safety function.

In every case, the operator must confirm the validity of the change before they are accepted and applied in the safety loop.

- Test all changes as part of the safety validation procedure.

- Sufficiently document all safety-related changes made via the HMI, including the following:
 - Authorization
 - Impact analysis
 - Execution
 - Test information
 - Revision information
- Changes to the safety-related system must comply with IEC 61511 standard on process safety, section 11.7.1 Operator Interface requirements.
- Changes to the safety-related system must comply with IEC 62061 for machine safety.
- The developer must follow the same sound development techniques and procedures used for other application software development, including the verification and testing of the operator interface and its access to other parts of the program. In the controller application software, set up a table that is accessible by the HMI and limit access to required data points only.
- Similar to the controller program, the HMI software needs to be secured and maintained for SIL-level compliance after the system has been validated and tested.

Safety Programs

A safety program has all the attributes of a standard program, except that it can be scheduled only in the safety task. A safety program may also define program-scoped safety tags. A safety program may be scheduled or unscheduled.

A safety program can contain only safety components. All of the routines in a safety program are safety routines. A safety program cannot contain standard routines or standard tags.

Safety Routines

Safety routines have all the attributes of standard routines, except that they can exist only in safety programs. One safety routine may be designated as the main routine. Another safety routine may be designated as the fault routine. Only safety-certified instructions may be used in safety routines.

For a listing of safety instructions, see [Appendix A](#).



ATTENTION: To preserve SIL 3, you must make sure that your safety logic does not attempt to read or write standard tags.

Safety Tags

The GuardLogix control system supports the use of both standard and safety tags in the same project. However, the programming software operationally differentiates standard tags from safety tags.

Safety tags have all the attributes of standard tags with the addition of mechanisms to provide SIL 3 data integrity.

Table 7 - Valid Data Types for Safety Tags

| | | |
|-----------------------------|----------------------------|----------------------------|
| • AUX_VALVE_CONTROL | • DINT | • MUTING_FOUR_SENSOR_BIDIR |
| • BOOL | • DIVERSE_INPUT | • MUTING_TWO_SENSOR_ASYM |
| • CAM_PROFILE | • EIGHT_POS_MODE_SELECTOR | • MUTING_TWO_SENSOR_SYM |
| • CAMSHAFT_MONITOR | • EMERGENCY_STOP | • MOTION_INSTRUCTION |
| • CB_CONTINUOUS_MODE | • ENABLE_PENDANT | • PHASE |
| • CB_CRANKSHAFT_POS_MONITOR | • EXT_ROUTINE_CONTROL | • PHASE_INSTRUCTION |
| • CB_INCH_MODE | • EXT_ROUTINE_PARAMETERS | • REAL ⁽¹⁾ |
| • CB_SINGLE_STROKE_MODE | • FBD_BIT_FIELD_DISTRIBUTE | • REDUNDANT_INPUT |
| • CONFIGURABLE_ROUT | • FBD_CONVERT | • REDUNDANT_OUTPUT |
| • CONNECTION_STATUS | • FBD_COUNTER | • SAFETY_MAT |
| • CONTROL | • FBD_LOGICAL | • SERIAL_PORT_CONTROL |
| • COUNTER | • FBD_MASK_EQUAL | • SFC_ACTION |
| • DCA_INPUT | • FBD_MASKED_MOVE | • SFC_STEP |
| • DCI_MONITOR | • FBD_TIMER | • SFC_STOP |
| • DCI_START | • FIVE_POS_MODE_SELECTOR | • SINT |
| • DCI_STOP | • INT | • STRING |
| • DCI_STOP_TEST | • LIGHT_CURTAIN | • THRS_ENHANCED |
| • DCI_STOP_TEST_LOCK | • MAIN_VALVE_CONTROL | • TIMER |
| • DCI_STOP_TEST_MUTE | • MANUAL_VALVE_CONTROL | • TWO_HAND_RUN_STATION |

(1) Safety tags created by using the REAL data type are valid only on 1756-L7xS safety controllers.

RSLogix 5000 software prevents the direct creation of invalid tags in a safety program. In the event that invalid tags are imported, they cannot be verified.

In safety programs on 1756-L6xS and 1768-L4xS controllers, RSLogix 5000 software also prevents the following:

- The creation of a safety tag using a structured type (user-defined, Add-On defined, predefined) when they contain one or more members of the REAL data type, including nested structures.
- The modification of a user-defined, or Add-On defined type that would cause an invalid data type to be included when the user-defined or Add-On defined type is already referenced directly or indirectly by a safety tag.
- Invalid tags created by using with the New Tag or Tag Properties dialog box.

IMPORTANT Aliasing between standard and safety tags is prohibited in safety applications.

Tags classified as safety tags are either controller-scoped or program-scoped. Controller-scoped safety tags can be read by either standard or safety logic or other communication devices, but can only be written to by safety logic or another GuardLogix safety controller. Program-scoped safety tags are accessible only by local safety routines. These are routines that reside within the safety program.

Tags associated with Safety I/O and produced or consumed safety data must be controller-scoped safety tags.

IMPORTANT Any controller-scoped safety tag is readable by any standard routine, but the update rate is based on the execution of the safety task. This means that safety tags are updated at the safety task periodic rate, which is different from standard tag behavior.

Standard Tags in Safety Routines (tag mapping)

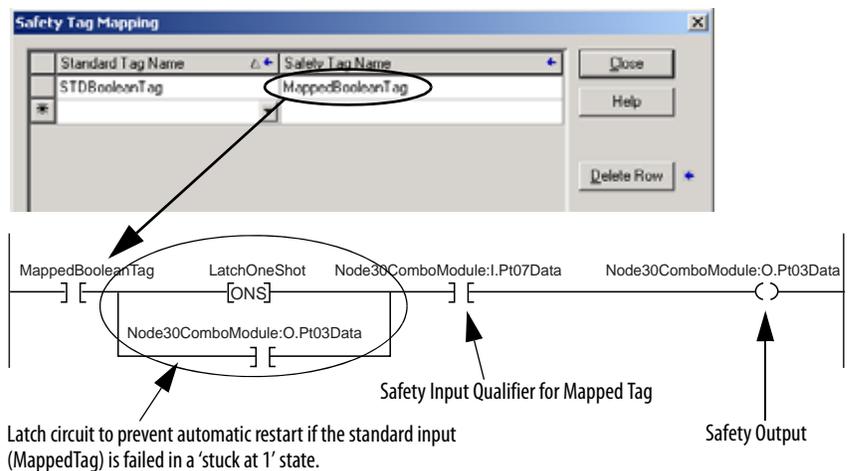
Controller-scoped standard tags can be mapped into safety tags, providing you with a mechanism to synchronize standard and safety actions.



ATTENTION: When using standard data in a safety routine, you are responsible for providing a reliable means of ensuring that the data is used in an appropriate manner. Using standard data in a safety tag does not make it safety data. You must not directly control a safety output with standard tag data.

This example illustrates how to qualify the standard data with safety data.

Figure 14 - Qualify Standard Data with Safety Data



Additional Resources

These documents contain addition information about GuardLogix controllers.

| Resource | Description |
|--|--|
| Logix5000 Controllers Design Considerations Reference Manual, publication 1756-RM094 | Provides information on managing tasks and the effects of task execution and timing on user data |
| GuardLogix Controllers User Manual, publication 1756-UM020 | Contains information on how to map tags |
| 1768 Compact GuardLogix Controllers User Manual, publication 1768-UM002 | Contains information on how to map tags |

Notes:

Safety Application Development

| Topic | Page |
|---|------|
| Safety Concept Assumptions | 51 |
| Basics of Application Development and Testing | 52 |
| Commissioning Life Cycle | 53 |
| Downloading the Safety Application Program | 59 |
| Uploading the Safety Application Program | 59 |
| Online Editing | 59 |
| Storing and Loading a Project from Nonvolatile Memory | 60 |
| Force Data | 60 |
| Inhibit a Module | 60 |
| Editing Your Safety Application | 61 |

Safety Concept Assumptions

The safety concept assumes the following:

- If you are responsible for creating, operating, and maintaining the application, you are fully qualified, specially trained, and experienced in safety systems.
- You apply the logic correctly, meaning that programming errors can be detected. Programming errors can be detected by strict adherence to specifications, programming and naming rules.
- You perform a critical analysis of the application and use all possible measures to detect a failure.
- You confirm all application downloads via a manual check of the safety task signature.
- You perform a complete functional test of the entire system before the operational startup of a safety-related system.

Table 8 - Controller Modes

| Controller Mode | Safety Task Status | Safety ⁽¹⁾ (up to and including) | Comments (A valid program has been downloaded to the controller.) |
|-----------------|----------------------------|--|--|
| Program | Unlocked No signature | | <ul style="list-style-type: none"> I/O connections established Safety Task logic is not being scanned. |
| Run | Unlocked No signature | (Development purposes only) | <ul style="list-style-type: none"> Forcing allowed Online editing allowed Safety memory is isolated, but is unprotected (read/write). Safety Task logic is being scanned. Primary and partner controllers process logic, cross-compare logic outputs. Logic outputs are written to safety outputs. |
| Run | Locked No signature | PLd/Cat. 3 Control reliable SIL 2 | <ul style="list-style-type: none"> New forces are not allowed. Existing forces are maintained. Online editing is not allowed. Safety memory is protected (read only) Safety task logic is scanned. Primary and partner controllers process logic, cross-compare logic outputs. Logic outputs are written to safety outputs. |
| Run | Unlocked With signature | Ple/Cat. 4 Control reliable SIL 3 | <ul style="list-style-type: none"> Forces are not allowed. (They must be removed to generate a safety task signature.) Online editing is not allowed. Safety memory is protected (read only). Safety task logic is scanned. Primary and partner controllers process logic, cross-compare logic outputs. Logic outputs are written to safety outputs. <ul style="list-style-type: none"> Safety task signature is unprotected and can be deleted by anyone who has access to the controller. |
| Run | Locked With signature | Ple/Cat. 4 Control reliable SIL 3 | <ul style="list-style-type: none"> Forces are not allowed. (They must be removed to generate a safety task signature.) Online editing is not allowed. Safety memory is protected (read only). Safety task logic is scanned. Primary and partner controllers process logic, cross-compare logic outputs. Logic outputs are written to safety outputs. <ul style="list-style-type: none"> Safety task signature is protected. Users must enter the unlock password to unlock the controller before they can delete the safety task signature. |

(1) To achieved this level, you must adhere to the safety requirements defined in this publication.

Basics of Application Development and Testing

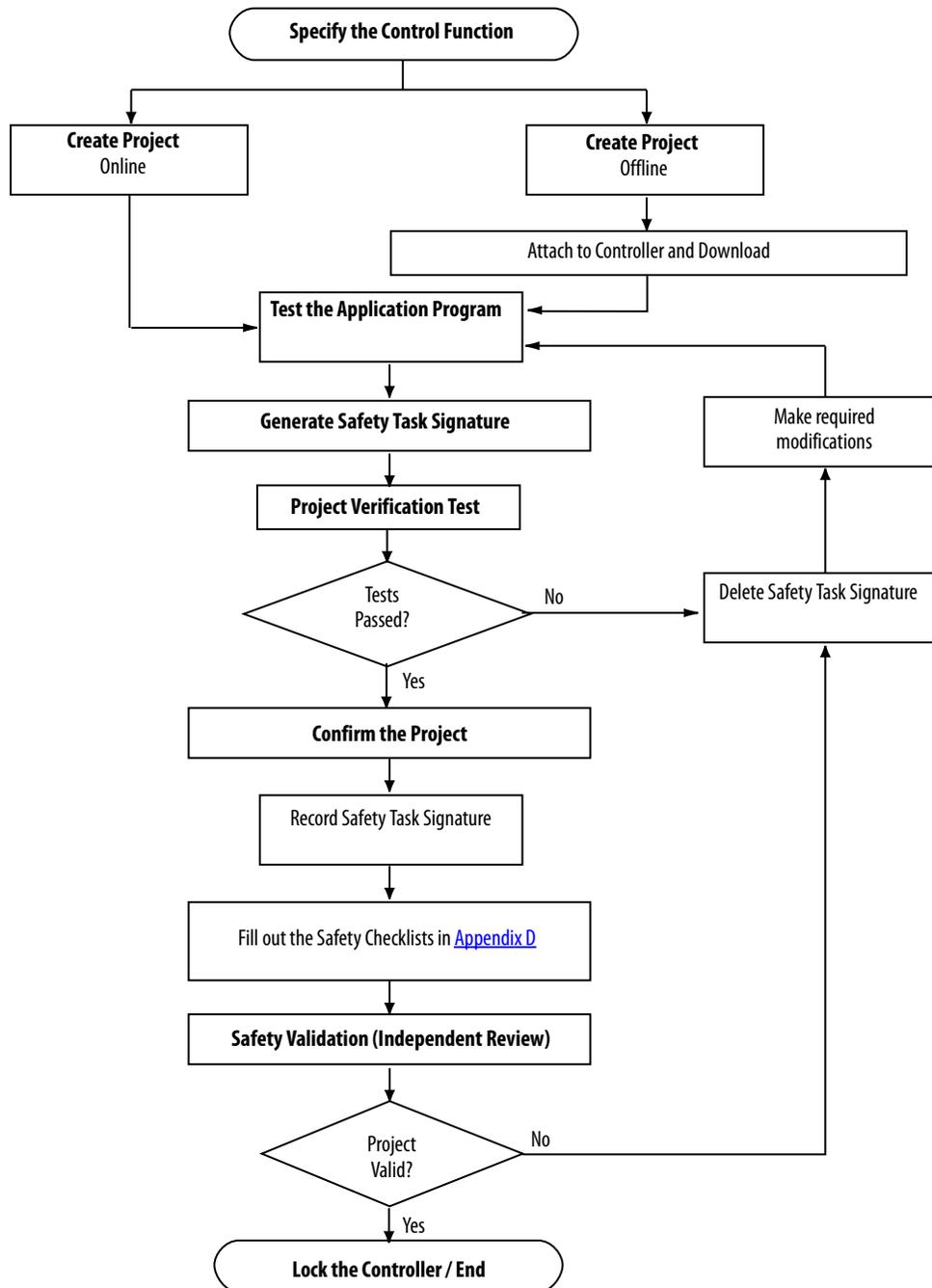
The application program for the intended SIL 3 system should be developed by the system integrator or a user trained and experienced in safety applications. The developer must follow good design practices:

- Use functional specifications, including flow charts, timing diagrams, and sequence charts.
- Perform a review of safety task logic.
- Perform application validation.

Commissioning Life Cycle

The flowchart below shows the steps required for commissioning a GuardLogix system. The items in bold text are explained in the following sections.

Figure 15 - Commission the System



Specification of the Control Function

You must create a specification for your control function. Use this specification to verify that program logic correctly and fully addresses your application's functional and safety control requirements. The specification may be presented in a variety of formats, depending on your application. However, the specification must be a detailed description that includes the following (if applicable):

- Sequence of operations
- Flow and timing diagrams
- Sequence charts
- Program description
- Program print out
- Written descriptions of the steps with step conditions and actuators to be controlled, including the following:
 - Input definitions
 - Output definitions
 - I/O wiring diagrams and references
 - Theory of operation
- Matrix or table of stepped conditions and the actuators to be controlled, including the sequence and timing diagrams
- Definition of marginal conditions, for example, operating modes and EMERGENCY STOP

The I/O portion of the specification must contain the analysis of field circuits, that is, the type of sensors and actuators.

- Sensors (Digital or Analog)
 - Signal in standard operation (dormant current principle for digital sensors, sensors OFF means no signal)
 - Determination of redundancies required for SIL levels
 - Discrepancy monitoring and visualization, including your diagnostic logic
- Actuators
 - Position and activation in standard operation (normally OFF)
 - Safe reaction/positioning when switching OFF or power failure
 - Discrepancy monitoring and visualization, including your diagnostic logic

Create the Project

The logic and instructions used in programming the application must be the following:

- Easy to understand
- Easy to trace
- Easy to change
- Easy to test

All logic should be reviewed and tested. Keep safety-related logic and standard logic separate.

Label the Program

The application program is clearly identified by one of the following:

- Name
- Date
- Revision
- Any other user identification

Test the Application Program

This step consists of any combination of Run and Program modes, online or offline edits, upload and download, and informal testing that is required to get an application running properly in preparation for the Project Verification test.

Generate the Safety Task Signature

The safety task signature is required to operate in a SIL3 safety function

The safety task signature uniquely identifies each project, including its logic, data, and configuration. The safety task signature is composed of an ID (identification number), date, and time.

You can generate the safety task signature if all of the following conditions are true:

- RSLogix 5000 software is online with the controller.
- The controller is in Program mode.
- The controller is safety-unlocked.
- The controller has no safety forces or pending online safety edits.
- The safety task status is OK.

Once application program testing is complete, you must generate the safety task signature. The programming software automatically uploads the safety task signature after it is generated.

IMPORTANT To verify the integrity of every download, you must manually record the safety task signature after initial creation and check the safety task signature after every download to make sure that it matches the original.

You can delete the safety task signature only when the GuardLogix controller is safety-unlocked and, if online, the keyswitch is in the REM or PROG position.

When a safety task signature exists, the following actions are not permitted within the safety task:

- Online or offline programming or editing of safety components
- Forcing Safety I/O
- Data manipulation (except through routine logic or another GuardLogix controller)

Project Verification Test

To check the application program for adherence to the specification, you must generate a suitable set of test cases covering the application. The set of test cases must be filed and retained as the test specification.

You must include a set of tests to prove the validity of the calculations (formulas) used in your application logic. Equivalent range tests are acceptable. These are tests within the defined value ranges, at the limits, or in invalid value ranges. The necessary number of test cases depends on the formulas used and must comprise critical value pairs.

Active simulation with sources (field devices) must also be included, as it is the only way to verify that the sensors and actuators in the system are wired correctly. Verify the operation of programmed functions by manually manipulating sensors and actuators.

You must also include tests to verify the reaction to wiring faults and network communication faults.

Project verification includes required functional verification tests of fault routines, and input and output channels, to be sure that the safety system operates properly.

To perform a functional verification test on the GuardLogix controller, you must perform a full test of the application. You must toggle each sensor and actuator involved in every safety function. From a controller perspective, this means toggling the I/O point going into the controller, not necessarily the actual activators. Be sure to test all shutdown functions, because these functions are not typically exercised during normal operation. Also, be aware that a functional verification test is valid only for the specific application tested. If the controller is

moved to another application, you must also perform start-up and functional verification testing on the controller in the context of its new application.

See [Functional Verification Tests](#) on page 12 for more information.

Confirm the Project

You must print or view the project, and compare the uploaded safety I/O and controller configurations, safety data, and safety task program logic to make sure that the correct safety components were downloaded, tested, and retained in the safety application program.

If your application program contains a safety Add-On Instruction that has been sealed with an instruction signature, you must also compare the instruction signature, date/time, and safety instruction signature to the values you recorded when you sealed the Add-On Instruction.

See [Appendix B, Safety Add-On Instructions](#) for information on creating and using safety Add-On Instructions in SIL 3 applications.

The steps below illustrate one method for confirming the project.

1. With the controller in Program mode, save the project.
2. Answer Yes to the Upload Tag Values prompt.
3. With RSLogix 5000 software offline, save the project with a new name, such as Offlineprojectname.ACD, where projectname is the name of your project.

This is the new tested master project file.

4. Close the project.
5. Move the original project archive file out of its current directory. You can delete this file or store it in an archival location. This step is required because if RSLogix 5000 software finds the projectname.ACD in this directory, it will correlate it with the controller project and will not perform an actual upload.
6. With the controller still in Program mode, upload the project from the controller.
7. Save the uploaded project as Onlineprojectname.ACD, where projectname is the name of your project.
8. Answer Yes to the Upload Tag Values prompt.

9. Use the RSLogix 5000 Program Compare utility to perform these comparisons:
 - Compare all of the properties of the GuardLogix controller and CIP Safety I/O modules.
 - Compare all of the properties of the safety task, safety programs and safety routines.
 - Compare all of the logic in the safety routines.

Safety Validation

An independent, third-party review of the safety system may be required before the system is approved for operation. An independent, third-party certification is required for IEC 61508 SIL 3.

Lock the GuardLogix Controller

The GuardLogix controller system can be safety-locked to help protect safety control components from modification. However, safety-locking the controller is not a requirement for SIL 3 applications. The safety-lock feature applies only to safety components, such as the safety task, safety programs, safety routines, safety tags, safety Add-On Instructions, safety I/O, and safety task signature. However, safety-locking alone does not satisfy SIL 3 requirements.

No aspect of safety can be modified while the controller is in the safety-locked state. When the controller is safety-locked, the following actions are not permitted in the safety task:

- Online or offline programming or editing
- Forcing safety I/O
- Data manipulation (except through routine logic or another GuardLogix controller)
- Creating or editing safety Add-On Instructions
- Generating or deleting the safety task signature
- If a safety task signature exists, only projects with a matching safety task signature can be downloaded to controller.

The default state of the controller is safety-unlocked. You may place the safety application in a safety-locked state regardless of whether you are online or offline, and regardless of whether you have the original source of the program. However, no safety forces or pending safety edits may be present. Safety-locked or -unlocked status cannot be modified when the keyswitch is in the RUN position.

To provide an additional layer of protection, separate passwords may be used for safety-locking or -unlocking the controller. Passwords are optional.

For information on using the safety-lock feature, refer to the GuardLogix Controllers User Manual, publication [1756-UM020](#), or the 1768 Compact GuardLogix Controllers User Manual, publication [1768-UM002](#).

Downloading the Safety Application Program

Upon download, application testing is required unless a safety task signature exists.

IMPORTANT To verify the integrity of every download, you must manually record the safety task signature after initial creation and check the safety task signature after every download to make sure that it matches the original.

Downloads to a safety-locked GuardLogix controller are allowed only if the safety task signature, the hardware series, and the operating system version of the offline project all match those contained in the target GuardLogix controller and the controller's safety task status is OK.

IMPORTANT If the safety task signature does not match and the controller is safety-locked, you must unlock the controller to download. In this case, downloading to the controller deletes the safety task signature. As a result, you must revalidate the application.



ATTENTION: The USB port is intended for temporary local programming purposes only and not intended for permanent connection.

Uploading the Safety Application Program

If the GuardLogix controller contains a safety task signature, the safety task signature will be uploaded with the project. This means that any changes to offline safety data will be overwritten as a result of the upload.

Online Editing

If there is no safety task signature and the controller is safety-unlocked, you can perform online edits to your safety routines.

TIP You cannot edit standard or safety Add-On Instructions while online.

Pending edits cannot exist when the controller is safety-locked or when there is a safety task signature. Online edits may exist when the controller is safety-locked. However, they may not be assembled or cancelled.

TIP Online edits in standard routines are unaffected by the safety-locked or -unlocked state.

See page [61](#) for more information on making edits to your application program.

Storing and Loading a Project from Nonvolatile Memory

In revision 18 or later, GuardLogix controllers support firmware upgrades and user program storage and retrieval by using a memory card. In a 1756 GuardLogix system, only the primary controller uses a memory card for nonvolatile memory.

When you store a safety application project on a memory card, Rockwell Automation recommends you select Remote Program as the Load mode, that is, the mode the controller should enter following the load. Prior to actual machine operation, operator intervention is required to start the machine.

You can initiate a load from nonvolatile memory only under these conditions:

- If the controller type specified by the project stored in nonvolatile memory matches your controller type
- If the major and minor revisions of the project in nonvolatile memory matches the major and minor revisions of your controller
- If your controller is not in Run mode

Loading a project to a safety-locked controller is allowed only when the safety task signature of the project stored in nonvolatile memory matches the project on the controller. If the signatures do not match or the controller is safety-locked without a safety task signature, you must first unlock the controller before attempting to update the controller via nonvolatile memory.

IMPORTANT

If you unlock the controller and initiate a load from nonvolatile memory, the safety-lock status, passwords, and safety task signature will be set to the values contained in nonvolatile memory once the load is complete.

Force Data

All data contained in an I/O, produced, or consumed safety tag, including CONNECTION_STATUS, can be forced while the project is safety-unlocked and no safety task signature exists. However, forces must be uninstalled, not just disabled, on all safety tags before the safety project can be safety-locked or a safety task signature can be generated. You cannot force safety tags while the project is safety-locked or when a safety task signature exists.

TIP

You can install and uninstall forces on standard tags regardless of the safety-locked or -unlocked state.

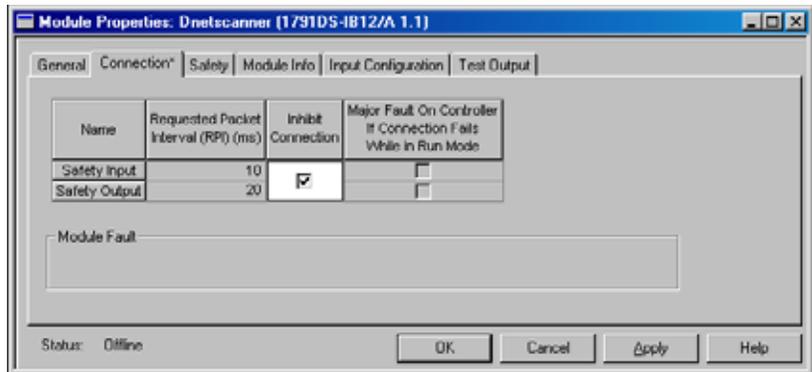
Inhibit a Module

You cannot inhibit or uninhibit safety I/O modules or producer controllers if the application is safety-locked or a safety task signature exists.

Follow these steps to inhibit a specific safety I/O module.

1. In RSLogix 5000 software, right-click the module and choose Properties.
2. On the Module Properties dialog box, click the Connection tab.

3. Check Inhibit Connection and click Apply.



The module is inhibited whenever the checkbox is checked. If a communication module is inhibited, all downstream modules are also inhibited.

Editing Your Safety Application

The following rules apply to changing your safety application in RSLogix 5000 software:

- Only authorized, specially-trained personnel can make program edits. These personnel should use all supervisory methods available, for example, using the controller keyswitch and software password protections.
- When authorized, specially-trained personnel make program edits, they assume the central safety responsibility while the changes are in progress. These personnel must also maintain safe application operation.
- When editing online, you must use an alternate protection mechanism to maintain the safety of the system.
- You must sufficiently document all program edits, including:
 - authorization.
 - impact analysis.
 - execution.
 - test information.
 - revision information.
- If online edits exist only in the standard routines, those edits are not required to be validated before returning to normal operation.
- You must make sure that changes to the standard routine, with respect to timing and tag mapping, are acceptable to your safety application.
- You **can** edit the logic portion of your program while offline or online, as described in the following sections.

Performing Offline Edits

When offline edits are made to only standard program elements, and the safety task signature matches following a download, you can resume operation.

When offline edits affect the safety program, you must revalidate all affected elements of the application, as determined by the impact analysis, before resuming operation.

The flowchart on page [63](#) illustrates the process for offline editing.

Performing Online Edits

If online edits affect the safety program, you must revalidate all affected elements of the application, as determined by the impact analysis, before resuming operation. The flowchart on page [63](#) illustrates the process for online editing.

TIP Limit online edits to minor program modifications such as setpoint changes or minor logic additions, deletions, and modifications.

Online edits are affected by the safety-lock and safety task signature features of the GuardLogix controller.

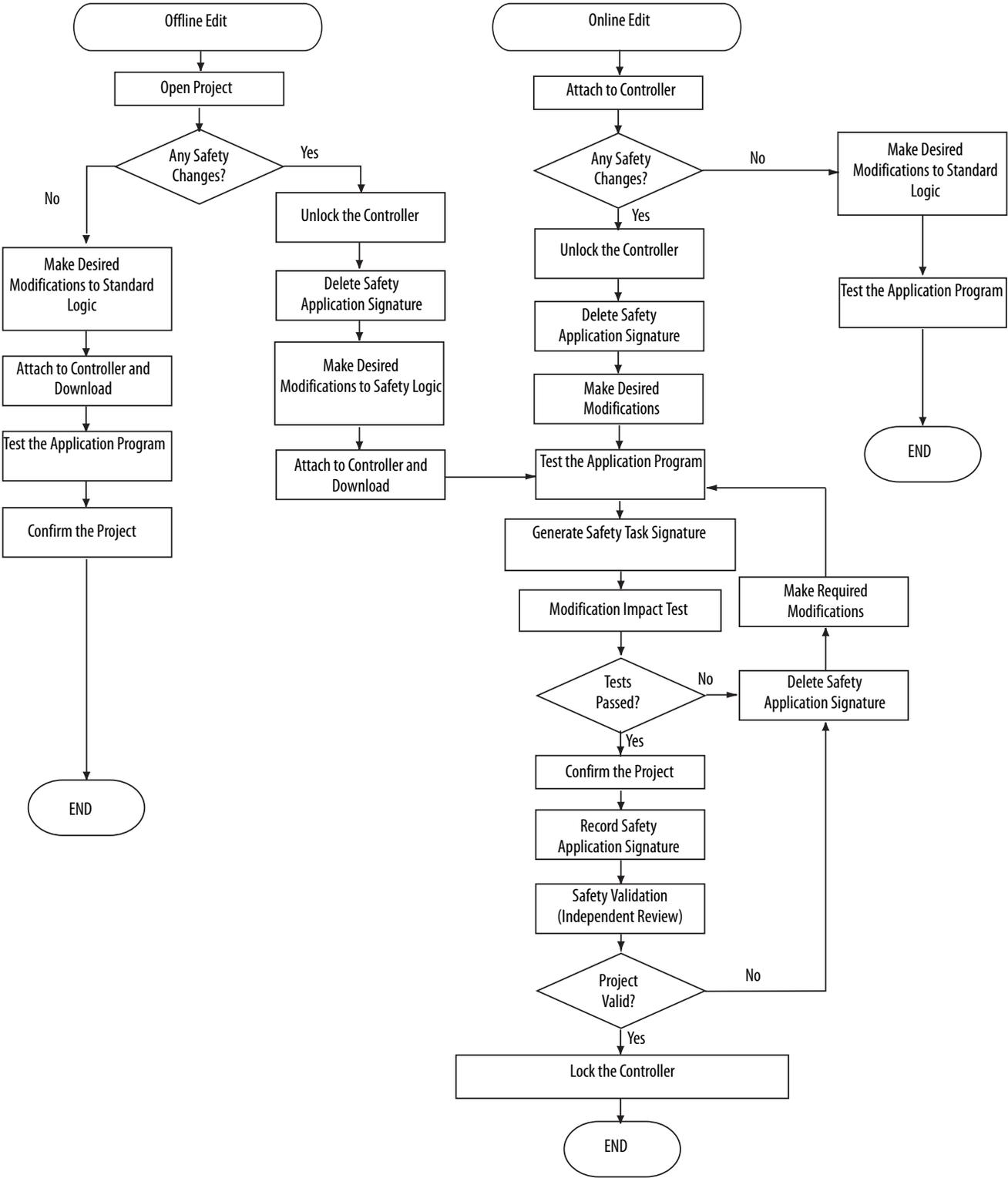
See [Generate the Safety Task Signature](#) on page [55](#) and [Lock the GuardLogix Controller](#) on page [58](#) for more information.

For detailed information on how to edit ladder logic in RSLogix 5000 software while online, see the Logix5000™ Controllers Quick Start, publication [1756-QS001](#).

Modification Impact Test

Any modification, enhancement, or adaptation of validated software must be planned and analyzed for any impact to the functional safety system. All appropriate phases of the software safety lifecycle need to be carried out as indicated by the impact analysis. At a minimum, functional testing of all impacted software must be carried out. All modifications to software specifications must be documented. Test results must also be documented. Refer to IEC 61508-3, Section 7.8 Software Modification, for detailed information.

Figure 16 - Online and Offline Edit Process



Notes:

Monitor Status and Handle Faults

| Topic | Page |
|--------------------------|------|
| Monitoring System Status | 65 |
| GuardLogix System Faults | 68 |

The GuardLogix architecture provides you with many ways of detecting and reacting to faults in the system. The first way that you can handle faults is to make sure you have completed the checklists for your application (see [Appendix D](#)).

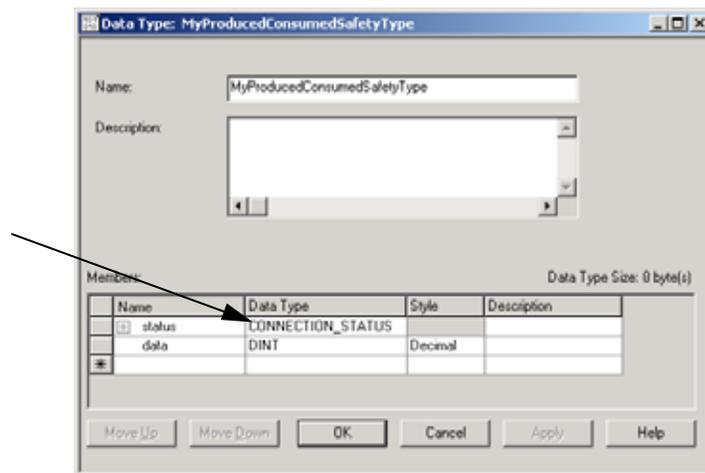
Monitoring System Status

You can view the status of safety tag connections. You can also determine current operating status by interrogating various device objects. It is your responsibility to determine what data is most appropriate to initiate a shutdown sequence.

CONNECTION_STATUS Data

The first member of the tag structure associated with safety input data and produced/consumed safety tag data contains the status of the connection. This member is a pre-defined data type called CONNECTION_STATUS.

Figure 17 - Data Type Dialog Box



The first two bits of the CONNECTION_STATUS data type contain a device's RunMode and ConnectionFaulted status bits. The following table describes the combinations of the RunMode and ConnectionFaulted states.

Table 9 - Safety Connection Status

| RunMode Status | ConnectionFaulted Status | Safety Connection Operation |
|----------------|--------------------------|---|
| 1 = Run | 0 = Valid | Data is actively being controlled by the producing device. The producing device is in Run mode. |
| 0 = Idle | 0 = Valid | The connection is active and the producing device is in the Idle state. The safety data is reset to zero. |
| 0 = Idle | 1 = Faulted | The safety connection is faulted. The state of the producing device is unknown. The safety data is reset to zero. |
| 1 | 1 | Invalid state. |



ATTENTION: Safety I/O connections and produced/consumed connections cannot be automatically configured to fault the controller if a connection is lost and the system transitions to the safe state. Therefore, if you need to detect a module fault to be sure that the system maintains SIL 3, you must monitor the Safety I/O CONNECTION_STATUS bits and initiate the fault via program logic.

Input and Output Diagnostics

Guard I/O modules provide pulse test and monitoring capabilities. If the module detects a failure, it sets the offending input or output to its safety state and reports the failure to the controller. The failure indication is made via input or output status and is maintained for a configurable amount of time after the failure is repaired.

IMPORTANT

You are responsible for providing application logic to latch these I/O failures and to make sure the system restarts properly.

I/O Module Connection Status

The CIP safety protocol provides status for each I/O module in the safety system. If an input connection failure is detected, the operating system sets all device inputs to their de-energized (safety) state, and the associated input status to faulted. If an output connection failure is detected, the operating system sets the associated output status to faulted. The output module de-energizes the outputs.

IMPORTANT

You are responsible for providing application logic to latch these I/O failures and to make sure the system restarts properly.

De-energize to Trip System

GuardLogix controllers are part of a de-energize to trip system, which means that zero is the safe state. Some, but not all, safety module faults cause all module inputs or outputs to be set to zero (safe state). Faults associated to a specific input channel result in that specific channel being set to zero; for example, a pulse test fault that is specific to channel 0 results in channel 0 input data being set to the safe state (0). If a fault is general to the module and not to a specific channel, the combined status bit displays the fault status and all module data is set to the safe state (0).

For information on how to use the RSLogix 5000 software, version 14 safety application instructions, see [Appendix F](#) of this manual and the GuardLogix Safety Application Instructions Safety Reference Manual, publication [1756-RM095](#).

Get System Value (GSV) and Set System Value (SSV) Instructions

The GSV and SSV instructions let you get (GSV) and set (SSV) controller system data stored in device objects. When you enter a GSV/SSV instruction, the programming software displays the valid object classes, object names, and attribute names for each instruction. Restrictions exist for using the GSV and SSV instructions with safety components.

IMPORTANT

The safety task cannot perform GSV or SSV operations on standard attributes.

The attributes of safety objects that can be written by the standard task are for diagnostic purposes only. They do not affect safety task execution.

For more information on which safety attributes are accessible via GSV and SSV instructions, refer to the following:

- GuardLogix Controllers User Manual, publication [1756-UM020](#)
- 1768 Compact GuardLogix Controllers User Manual, publication [1768-UM002](#)

For general information on using GSV and SSV instructions, refer to the Logix5000 Controllers General Instructions Reference Manual, publication [1756-RM003](#).

GuardLogix System Faults

Faults in the GuardLogix system fall into these three categories:

- Nonrecoverable controller faults
- Nonrecoverable safety faults
- Recoverable faults

For information on handling faults, refer to the GuardLogix Controllers User Manual, publication [1756-UM020](#), or the 1768 Compact GuardLogix Controllers User Manual, publication [1768-UM002](#).

Nonrecoverable Controller Faults

A nonrecoverable controller fault occurs if the controller's internal diagnostics fail. Partnership is lost when a nonrecoverable controller fault occurs in either the primary controller or the safety partner, causing the other to generate a nonrecoverable watchdog timeout fault. Standard task and safety task execution stops, and Safety I/O transitions to the safe state.

Recovery from a nonrecoverable controller fault requires a download of the application program.

Nonrecoverable Safety Faults

In the event of a non-recoverable safety fault, the controller logs the fault to the controller-scoped fault handler and shuts down the safety task, including Safety I/O and safety logic.

To recover from a nonrecoverable safety fault, safety memory is reinitialized either from the safety task signature (happens automatically when you clear the fault) or, if no safety task signature exists, via an explicit download of the safety project.

You can override the safety fault by clearing the fault log entry through the controller-scoped safety fault handler. This allows standard tasks to keep running.



ATTENTION: Overriding the safety fault does not clear it. If you override the safety fault, it is your responsibility to prove that doing so maintains SIL 3.

Recoverable Faults

Controller faults caused by user programming errors in a safety program trigger the controller to process the logic contained in the project's safety program fault handler. The safety program fault handler provides the application with the opportunity to resolve the fault condition and then recover.



ATTENTION: You must provide proof to your certifying agency that automatic recovery from recoverable faults maintains SIL 3.

When a safety program fault handler does not exist or the fault is not recovered by it, the controller processes the logic in the controller-scoped fault handler, terminating safety program logic execution and leaving safety I/O connections active, but idle.

IMPORTANT

When the execution of safety program logic is terminated due to a recoverable fault that is not handled by the safety program fault handler, the safety I/O connections are closed and reopened to reinitialize safety connections.

If user logic is terminated as a result of a recoverable fault that is not recovered, safety outputs are placed in the safe state and the producer of safety-consumed tags commands the consumers to place them in a safe state.

TIP

When using safety I/O for standard applications, safety I/O will be commanded to the safe state if user logic is terminated as a result of a recoverable fault that is not recovered.

If a recoverable safety fault is overridden in the controller-scoped fault handler, only standard tasks keep running. If the fault is not overridden, the standard tasks are also shut down.



ATTENTION: Overriding the safety fault does not clear it. If you override the safety fault, it is your responsibility to prove that doing so maintains SIL 3.

Notes:

Safety Instructions

| Topic | Page |
|--|------|
| Safety Application Instructions | 71 |
| Metal Form Safety Application Instructions | 72 |
| Safety Instructions | 73 |
| Additional Resources | 74 |

For the latest information, see our safety certificates at <http://www.rockwellautomation.com/products/certification/safety/>.

Safety Application Instructions

Table 10 - RSLogix 5000 Software, Version 20 and Later, Safety Application Instructions

| Mnemonic | Name | Purpose |
|----------|--|---|
| DCA | Dual Channel Input - Analog (integer version) | Monitors two analog values for deviation and range tolerance. |
| DCAF | Dual Channel Input - Analog (floating point version) | |

Table 11 - RSLogix 5000 Software, Version 17 and Later, Safety Application Instructions

| Mnemonic | Name | Purpose |
|----------|--|--|
| CROUT | Configurable Redundant Output | Controls and monitors redundant outputs. |
| DCS | Dual Channel Input - Stop | Monitors dual-input safety devices whose main purpose is to provide a stop function, such as an E-stop, light curtain, or gate switch. |
| DCST | Dual Channel Input - Stop With Test | Monitors dual-input safety devices whose main purpose is to provide a stop function, such as an E-stop, light curtain, or gate switch. It includes the added capability of initiating a functional test of the stop device. |
| DCSTL | Dual Channel Input - Stop With Test and Lock | Monitors dual-input safety devices whose main purpose is to provide a stop function, such as an E-stop, light curtain, or gate switch. It includes the added capability of initiating a functional test of the stop device and can monitor a feedback signal from a safety device and issue a lock request to a safety device. |
| DCSTM | Dual Channel Input - Stop With Test and Mute | Monitors dual-input safety devices whose main purpose is to provide a stop function, such as an E-stop, light curtain, or gate switch. It includes the added capability of initiating a functional test of the stop device and the ability to mute the safety device. |
| DCM™ | Dual Channel Input - Monitor | Monitors dual-input safety devices. |
| DCSRT | Dual Channel Input - Start | Energizes dual-input safety devices whose main function is to start a machine safely, for example an enable pendant. |

Table 11 - RSLogix 5000 Software, Version 17 and Later, Safety Application Instructions

| Mnemonic | Name | Purpose |
|----------|----------------------------------|---|
| SMAT | Safety Mat | Indicates whether or not the safety mat is occupied. |
| THRSe | Two-Hand Run Station – Enhanced | Monitors two diverse safety inputs, one from a right-hand push button and one from a left-hand push button, to control a single output. Features configurable channel-to-channel discrepancy time and enhanced capability for bypassing a two-hand run station. |
| TSAM | Two Sensor Asymmetrical Muting | Provides temporary, automatic disabling of the protective function of a light curtain, using two muting sensors arranged asymmetrically. |
| TSSM | Two Sensor Symmetrical Muting | Provides temporary, automatic disabling of the protective function of a light curtain, using two muting sensors arranged symmetrically. |
| FSBM | Four Sensor Bidirectional Muting | Provides temporary, automatic disabling of the protective function of a light curtain, using four sensors arranged sequentially before and after the light curtain's sensing field. |

Metal Form Safety Application Instructions

These instructions are available in RSLogix 5000 software, version 17 and later.

| Mnemonic | Name | Purpose |
|----------|----------------------------------|---|
| CBCM | Clutch Brake Continuous Mode | Used for press applications where continuous operation is desired. |
| CBIM | Clutch Brake Inch Mode | Used for press applications where minor slide adjustments are required, such as press setup. |
| CBSSM | Clutch Brake Single Stroke Mode | Used in single-cycle press applications. |
| CPM | Crankshaft Position Monitor | Used to determine the slide position of the press. |
| CSM | Camshaft Monitor | Monitors motion for the starting, stopping, and running operations of a camshaft. |
| EPMS | Eight-position Mode Selector | Monitors eight safety inputs to control one of the eight outputs corresponding to the active input. |
| AVC | Auxiliary Valve Control | Controls an auxiliary valve that is used in conjunction with a main valve. |
| MVC | Main Valve Control | Controls and monitors a main valve. |
| MMVC | Maintenance Manual Valve Control | Used to manually drive a valve during maintenance operations. |

Safety Instructions

Routines in the safety task may use these ladder logic safety instructions.

Table 12 - Ladder Logic Safety Instructions

| Type | Mnemonic | Name | Purpose | RSLogix 5000 Version ⁽⁶⁾ |
|--------------|---------------------|-----------------------------|--|---|
| Array (File) | FAL ⁽¹⁾ | File Arithmetic and Logic | Perform copy, arithmetic, logic, and function operations on data stored in an array | 20 |
| | FLL ⁽¹⁾ | File Fill | Fill the element of an array with the Source Value, while leaving the source value unchanged | 20 |
| | FSC ⁽¹⁾ | File Search and Compare | Compare the value in an array, element by element | 20 |
| | SIZE ⁽¹⁾ | Size In Elements | Find the size of a dimension of an array | 20 |
| Bit | XIC | Examine If Closed | Enable outputs when a bit is set | 14 |
| | XIO | Examine If Open | Enable outputs when a bit is cleared | 14 |
| | OTE | Output Energize | Set a bit | 14 |
| | OTL | Output Latch | Set a bit (retentive) | 14 |
| | OTU | Output Unlatch | Clear bit (retentive) | 14 |
| | ONS | One Shot | Triggers an event to occur one time | 14 |
| | OSR | One Shot Rising | Triggers an event to occur one time on the false-to-true (rising) edge of change-of-state | 14 |
| | OSF | One Shot Falling | Triggers an event to occur one time on the true-to-false (falling) edge of change-of-state | 14 |
| Timer | TON | Timer On Delay | Time how long a timer is enabled | 14 |
| | TOF | Timer Off Delay | Time how long a timer is disabled | 14 |
| | RTO | Retentive Timer On | Accumulate time | 14 |
| | CTU | Count Up | Count up | 14 |
| | CTD | Count Down | Count down | 14 |
| | RES | Reset | Reset a timer or counter | 14 |
| | Compare | CMP ⁽¹⁾⁽²⁾ | Compare | Perform a comparison on the arithmetic operations you specify in the expression |
| EQU | | Equal To | Test whether two values are equal | 14 |
| GEQ | | Greater Than Or Equal To | Test whether one value is greater than or equal to a second value | 14 |
| GRT | | Greater Than | Test whether one value is greater than a second value | 14 |
| LEQ | | Less Than Or Equal To | Test whether one value is less than or equal to a second value | 14 |
| LES | | Less Than | Test whether one value is less than a second value | 14 |
| MEQ | | Masked Comparison for Equal | Pass source and compare values through a mask and test whether they are equal | 14 |
| NEQ | | Not Equal To | Test whether one value is not equal to a second value | 14 |
| LIM | | Limit Test | Test whether a value falls within a specified range | 14 |
| Move | CLR | Clear | Clear a value | 14 |
| | COP ⁽³⁾ | Copy | Copy a value | 14 |
| | MOV | Move | Copy a value | 14 |
| | MVM | Masked Move | Copy a specific part of an integer | 14 |
| | SWPB ⁽¹⁾ | Swap Byte | Rearrange the bytes of a value | 20 |
| Logical | AND | Bitwise AND | Perform bitwise AND operation | 14 |
| | NOT | Bitwise NOT | Perform bitwise NOT operation | 14 |
| | OR | Bitwise OR | Perform bitwise OR operation | 14 |
| | XOR | Bitwise Exclusive OR | Perform bitwise exclusive OR operation | 14 |

Table 12 - Ladder Logic Safety Instructions

| Type | Mnemonic | Name | Purpose | RSLogix 5000 Version ⁽⁶⁾ |
|-----------------|--------------------|--------------------------|---|-------------------------------------|
| Program Control | JMP | Jump To Label | Jump over a section of logic that does not always need to be executed (skips to referenced label instruction) | 14 |
| | LBL | Label | Labels an instruction so that it can be referenced by a JMP instruction | 14 |
| | JSR | Jump to Subroutine | Jump to a separate routine | 14 |
| | RET | Return | Return the results of a subroutine | 14 |
| | SBR | Subroutine | Pass data to a subroutine | 14 |
| | TND | Temporary End | Mark a temporary end that halts routine execution | 14 |
| | MCR | Master Control Reset | Disable all the rungs in a section of logic | 14 |
| | AFI | Always False Instruction | Disable a rung | 14 |
| | NOP | No Operation | Insert a placeholder in the logic | 14 |
| | EVENT | Trigger Event Task | Trigger one execution of an event task ⁽⁵⁾ | 20 |
| Math/ Compute | ADD | Add | Add two values | 14 |
| | CPT ⁽¹⁾ | Compute | Perform the arithmetic operation defined in the expression | 20 |
| | SUB | Subtract | Subtract two values | 14 |
| | MUL | Multiply | Multiply two values | 14 |
| | DIV | Divide | Divide two values | 14 |
| | MOD | Modulo | Determine the remainder after one value is divided by a second value | 14 |
| | SQR | Square Root | Calculate the square root of a value | 14 |
| | NEG | Negate | Take the opposite sign of a value | 14 |
| | ABS | Absolute Value | Take the absolute value of a value | 14 |
| I/O | GSV ⁽⁴⁾ | Get System Value | Get controller status information | 14 |
| | SSV ⁽⁴⁾ | Set System Value | Set controller status information | 14 |

- (1) Supported only on 1756-L7xS and 1756-L7xSXT controllers. For the data type REAL, a floating point format is supported for safety routines on 1756-L7xS and 1756-L7xSXT controllers.
- (2) Advanced operands like SIN, COS, and TAN are not supported in safety routines.
- (3) The length operand must be a constant when the COP instruction is used in a safety routine. The length of the source and the destination must be the same.
- (4) Refer to the GuardLogix Controllers User Manual, publication [1756-UM020](#), for special considerations when using the GSV and SSV instructions.
- (5) The event instruction will trigger a scan of the standard task.
- (6) Available in this version and later.

Additional Resources

Refer to these publications for more information.

| Resource | Description |
|--|--|
| GuardLogix Safety Application Instruction Set Reference Manual, publication 1756-RM095 | Provides more information on the safety application instructions |
| Logix5000 Controllers General Instructions Reference Manual, publication 1756-RM003 | Contains detailed information on the Logix instruction set |

Safety Add-On Instructions

| Topic | Page |
|--|------|
| Creating and Using a Safety Add-On Instruction | 75 |
| Additional Resources | 80 |

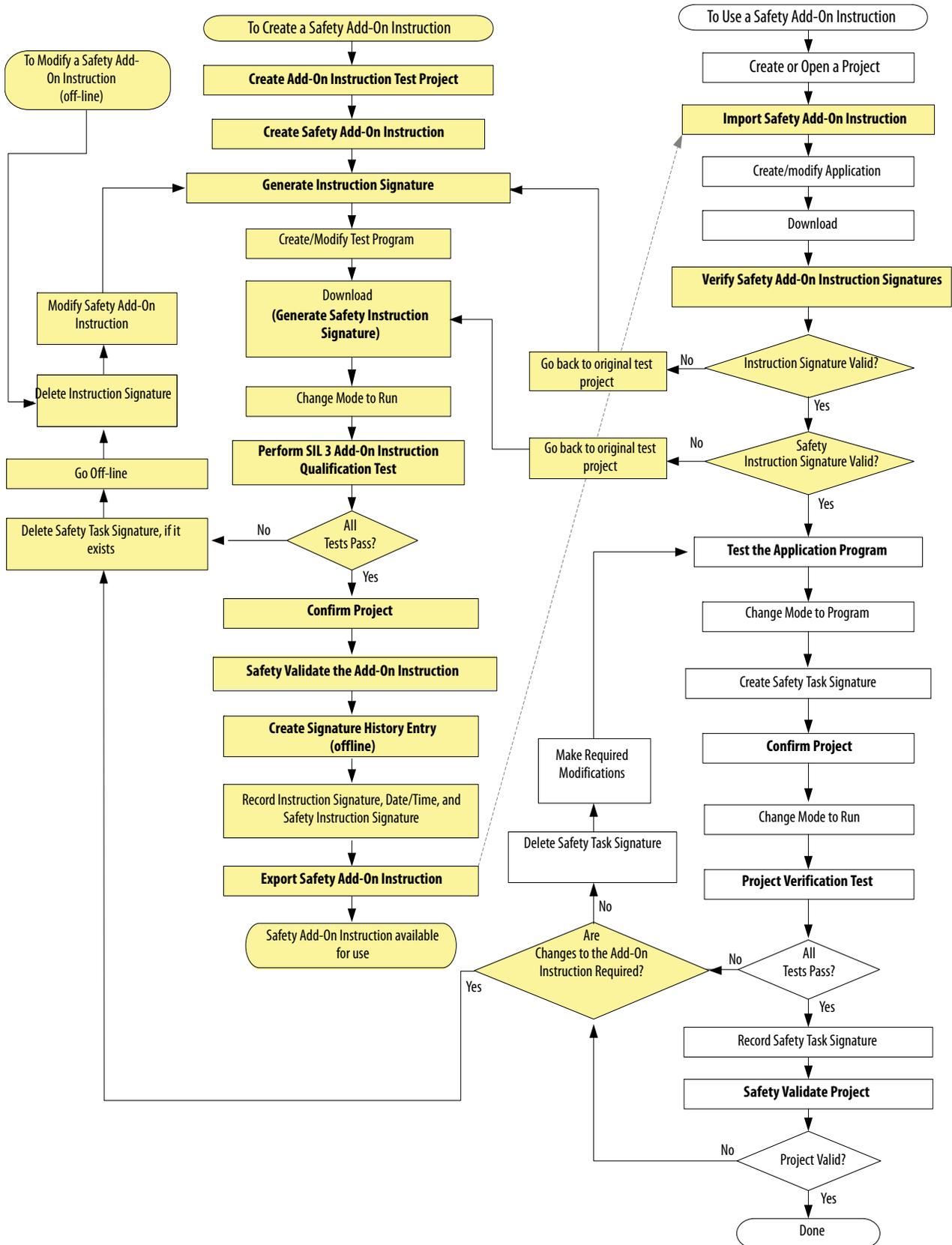
With RSLogix 5000 software, version 18 and later, you can create safety Add-On Instructions. Safety Add-On Instructions let you encapsulate commonly-used safety logic into a single instruction, making it modular and easier to reuse.

Safety Add-On Instructions use the instruction signature of high-integrity Add-On Instructions and also a SIL 3 safety instruction signature for use in safety-related functions up to and including SIL 3.

Creating and Using a Safety Add-On Instruction

The flowchart on page [76](#) shows the steps required for creating a safety Add-On Instruction and then using that instruction in a SIL 3 safety application program. The shaded items are steps unique to Add-On Instructions. The items in bold text are explained in the pages following the flowchart.

Figure 18 - Flowchart for Creating and Using Safety Add-On Instructions



Create Add-On Instruction Test Project

You need to create a unique test project, specifically for creating and testing the safety Add-On Instruction. This must be a separate and dedicated project to minimize any unexpected influences.

Follow the guidelines for projects described in [Create the Project on page 55](#).

Create a Safety Add-On Instruction

For guidance in creating Add-On Instructions, refer to the Logix5000 Controllers Add-On Instruction Programming Manual, publication [1756-PM010](#).

Generate Instruction Signature

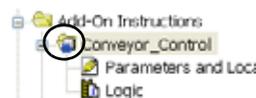
The instruction signature lets you quickly determine if the instruction has been modified. Each Add-On Instruction has the ability to have its own signature. The instruction signature is required when an Add-On Instruction is used in safety-related functions, and may be required for regulated industries. Use it when your application calls for a higher level of integrity.

The instruction signature consists of an ID number and timestamp that identifies the contents of the Add-On Instruction at a given point in time.

Once generated, the instruction signature seals the Add-On Instruction, preventing it from being edited while the signature is in place. This includes rung comments, tag descriptions, and any instruction documentation that was created. When the instruction is sealed, you can perform only these actions:

- Copy the instruction signature
- Create or copy a signature history entry
- Create instances of the Add-On Instruction
- Download the instruction
- Remove the instruction signature
- Print reports

When an instruction signature has been generated, RSLogix 5000 software displays the instruction definition with the seal icon.



IMPORTANT

If you plan to protect your Add-On Instruction by using the source protection feature in RSLogix 5000 software, you must enable source protection prior to generating the instruction signature.

Download and Generate Safety Instruction Signature

When a sealed safety Add-On Instruction is downloaded for the first time, a SIL 3 safety instruction signature is automatically generated. The safety instruction signature is an ID number that identifies the execution characteristics of the safety Add-On Instruction.

SIL 3 Add-On Instruction Qualification Test

Safety Add-On Instruction SIL 3 tests must be performed in a separate, dedicated application to make sure unintended influences are minimized. The developer must follow a well-designed test plan and perform a unit test of the safety Add-On Instruction that exercises all possible execution paths through the logic, including the valid and invalid ranges of all input parameters.

Development of all safety Add-On Instructions must meet IEC 61508 - 'Requirements for software module testing', which provides detailed requirements for unit testing.

Confirm the Project

You must print or view the project, and manually compare the uploaded safety I/O and controller configurations, safety data, safety Add-On Instruction definitions, and safety task program logic to make sure that the correct safety components were downloaded, tested, and retained in the safety application program.

See [Confirm the Project on page 57](#) for a description of one method for confirming a project.

Safety Validate Add-On Instructions

An independent, third-party review of the safety Add-On Instruction may be required before the instruction is approved for use. An independent, third-party validation is required for IEC 61508 SIL 3.

Create Signature History Entry

The signature history provides a record for future reference. A signature history entry consists of the instruction signature, the name of the user, the timestamp value, and a user-defined description. Up to six history entries may be stored. You must be offline to create a signature history entry.

TIP The Signature Listing report in RSLogix 5000 software prints the instruction signature, the timestamp, and the safety instruction signature. Print the report by right-clicking Add-On Instruction in the Controller Organizer and choosing Print>Signature Listing.

Export and Import the Safety Add-On Instruction

When you export a safety Add-On Instruction, choose the option to include all referenced Add-On Instructions and User-Defined Types in the same export file. By including referenced Add-On Instructions, you make it easier to preserve the signatures.

When importing Add-On Instructions, consider these guidelines:

- You cannot import a safety Add-On Instruction into a standard project.
- You cannot import a safety Add-On Instruction into a safety project that has been safety-locked or one that has a safety task signature.
- You cannot import a safety Add-On Instruction while online.
- If you import an Add-On Instruction with an instruction signature into a project where referenced Add-On Instructions or User-Defined Types are not available, you may need to remove the signature.

Verify Safety Add-On Instruction Signatures

After you download the application project containing the imported safety Add-On Instruction, you must compare the instruction signature value, the date and timestamp, and the safety instruction signature values with the original values you recorded prior to exporting the safety Add-On Instruction. If they match, the safety Add-On Instruction is valid and you can continue with the validation of your application.

Test the Application Program

This step consists of any combination of Run and Program mode, online or offline program edits, upload and download, and informal testing that is required to get an application running properly.

Project Verification Test

Perform an engineering test of the application, including the safety system.

See [Functional Verification Tests on page 12](#) and [Project Verification Test on page 56](#) for more information on requirements.

Safety Validate Project

An independent, third-party review of the safety system may be required before the system is approved for operation. An independent, third-party validation is required for IEC 61508 SIL 3.

Additional Resources

For more information on using Add-On Instructions, refer to these publications.

| Resource | Description |
|--|---|
| Logix5000 Controllers Add-On Instructions Programming Manual, publication 1756-PM010 | Provides information on planning, creating, using, importing and exporting Add-On Instructions in RSLogix 5000 applications |
| Import/Export Project Components Programming Manual, publication 1756-PM019 | Contains detailed information on importing and exporting |

Reaction Times

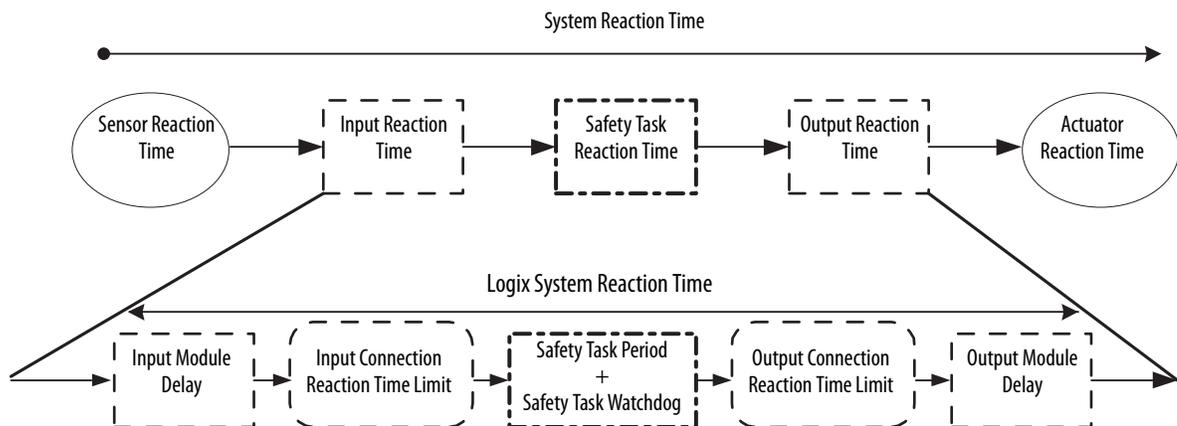
| Topic | Page |
|----------------------------|------|
| System Reaction Time | 81 |
| Logix System Reaction Time | 81 |

System Reaction Time

To determine the system reaction time of any control chain, you must add up the reaction times of all of components of the safety chain.

$$\text{System Reaction Time} = \text{Sensor Reaction Time} + \text{Logix System Reaction Time} + \text{Actuator Reaction Time}$$

Figure 19 - System Reaction Time

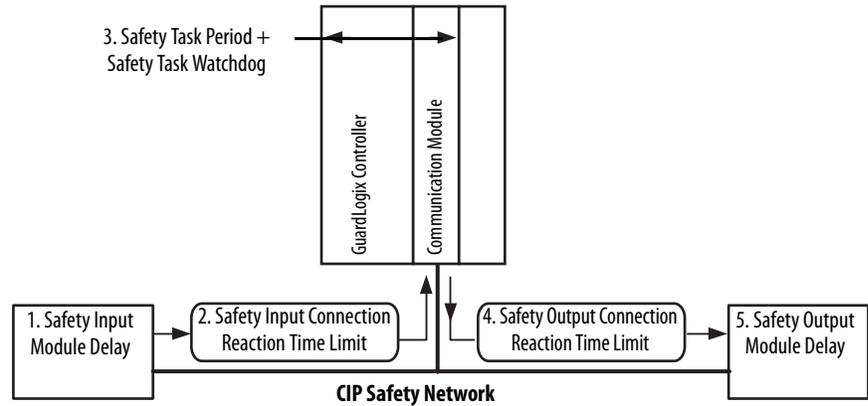


Logix System Reaction Time

The following sections provide information on calculating the Logix System Reaction Time for a simple input-logic-output chain and for a more complex application using produced/consumed safety tags in the logic chain.

Simple Input-logic-output Chain

Figure 20 - Logix System Worst-case Reaction Time for Simple Input to Logic to Output



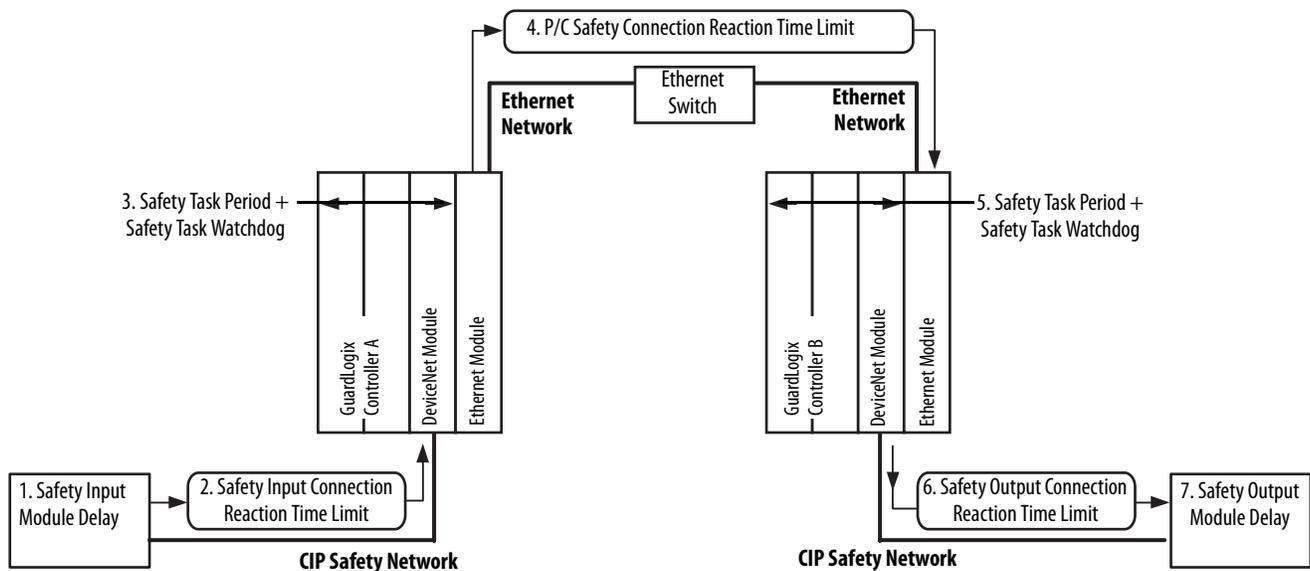
The Logix System Reaction Time for any simple input to logic to output chain consists of these five components:

1. Safety input module reaction time plus input delay time
2. Safety Input Connection Reaction Time Limit
(Read from the Module Properties dialog box in RSLogix 5000 software, this value is a multiple of the safety input module connection RPI.)
3. Safety Task Period plus Safety Task Watchdog time
4. Safety Output Connection Reaction Time Limit
(Read from the Module Properties dialog box in RSLogix 5000 software, this value is a multiple of the safety task period.)
5. Safety output module reaction time

To aid you in determining the reaction time of your particular control loop, a Microsoft Excel spreadsheet is available in the Tools folder of the RSLogix 5000 software CD.

Logic Chain Using Produced/Consumed Safety Tags

Figure 21 - Logix System Reaction Time for Input to Controller A Logic to Controller B Logic to Output Chain



The Logix System Reaction Time for any input to controller A logic to controller B logic to output chain consists of these seven components:

1. Safety input module reaction time plus input delay time
2. Safety Input Connection Reaction Time Limit
3. Safety Task Period plus Safety Task Watchdog time for Controller A
4. Produced/Consumed Safety Connection Reaction Time Limit
5. Safety Task Period plus Safety Task Watchdog time for Controller B
6. Safety Output Connection Reaction Time Limit
7. Safety output module reaction time

To aid you in determining the reaction time of your particular control loop, a Microsoft Excel spreadsheet is available in the Tools folder of the RSLogix 5000 software CD.

Factors Affecting Logix Reaction-time Components

The Logix Reaction Time components described in the previous sections can be influenced by a number of factors.

Table 13 - Factors Affecting Logix System Reaction-time

| These reaction time components | Are influenced by the following factors |
|---|--|
| Input module delay | Input module reaction time |
| | Each input channels On-Off and Off-On delay settings |
| Safety Input Connection Reaction Time Limit | Input module settings for: <ul style="list-style-type: none"> Requested Packet Interval (RPI) Timeout Multiplier Delay Multiplier |
| | The amount of network communication traffic |
| | The system's EMC environment |
| Safety Task Period and Safety Task Watchdog | Safety Task Period setting |
| | Safety Task Watchdog setting |
| | The number and execution time of instructions in the safety task |
| | Any higher priority tasks that may preempt safety task execution |
| Produced/Consumed Safety Connection Reaction Time Limit | Consumed tag settings for: <ul style="list-style-type: none"> RPI Timeout Multiplier Delay Multiplier |
| | The amount of network communication traffic |
| | The system's EMC environment |
| Output Connection Reaction Time Limit | Safety Task Period setting |
| | Output module's settings for: <ul style="list-style-type: none"> Timeout Multiplier Delay Multiplier |
| | The amount of network communication traffic |
| | The system's EMC environment |
| Output module delay | Output module reaction time |

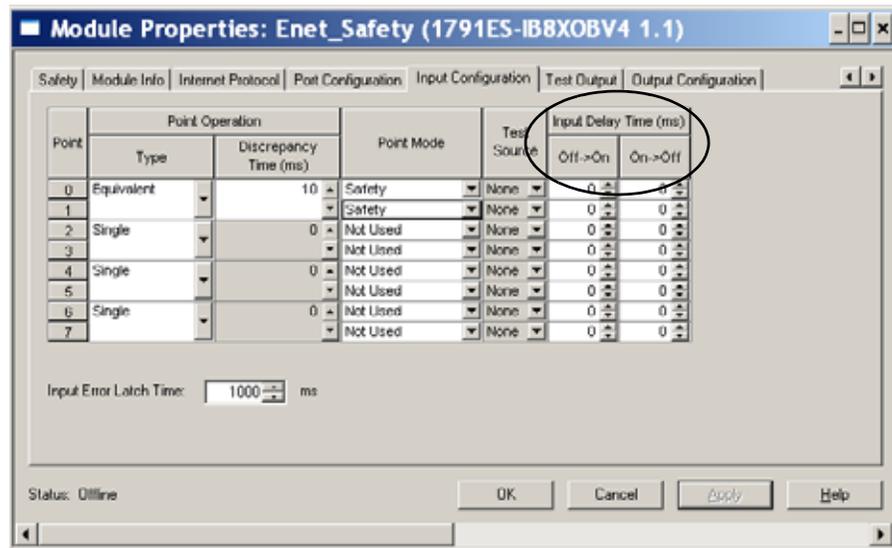
The following sections describe how to access data or settings for many of these factors.

Accessing Input Module Delay Time Settings

To configure input module delay time in RSLogix 5000 software, follow these steps.

1. In the configuration tree, right-click your I/O module and choose Properties
2. Click the Input Configuration tab.

3. Adjust the input delay time as required for your application.



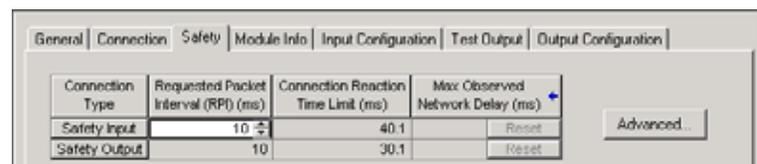
Accessing Input and Output Safety Connection Reaction Time Limit

The Connection Reaction Time Limit is defined by these three values:

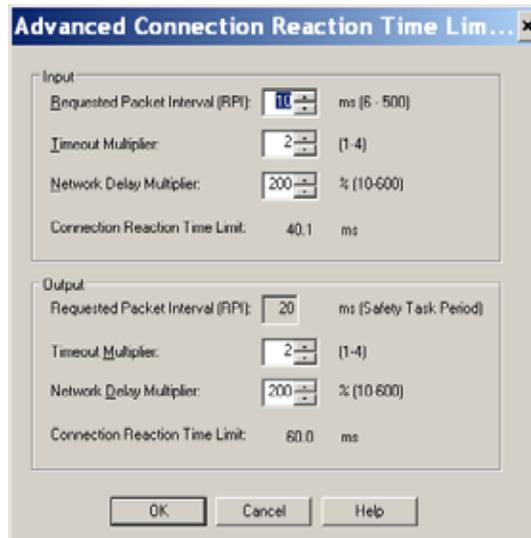
| Value | Description |
|---------------------------------|--|
| Requested Packet Interval (RPI) | This is how often the input and output packets are placed on the wire (network). |
| Timeout Multiplier | The Timeout Multiplier is essentially the number of retries before timing out. |
| Network Delay Multiplier | The Network Delay Multiplier accounts for any known delays on the wire. When these delays occur, timeouts can be avoided using this parameter. |

By adjusting these values, you can adjust the Connection Reaction Time Limit. If a valid packet is not received within the CRTL, the safety connection times out and the input and output data are placed in the safe state (OFF). To view or configure these settings, follow these steps.

1. In the configuration tree, right-click your I/O module and choose Properties.
2. Click the Safety tab.



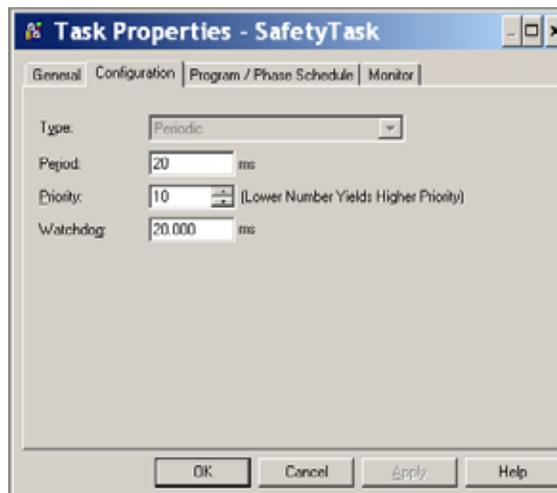
3. Click Advanced to open the Advanced Connection Reaction Time Limit dialog box.



Configuring the Safety Task Period and Watchdog

The safety task is a periodic timed task. You select the task priority and watchdog time via the Task Properties - Safety Task dialog box in your RSLogix 5000 project.

To access the safety task period and watchdog time settings, right-click the Safety Task and choosing Properties.

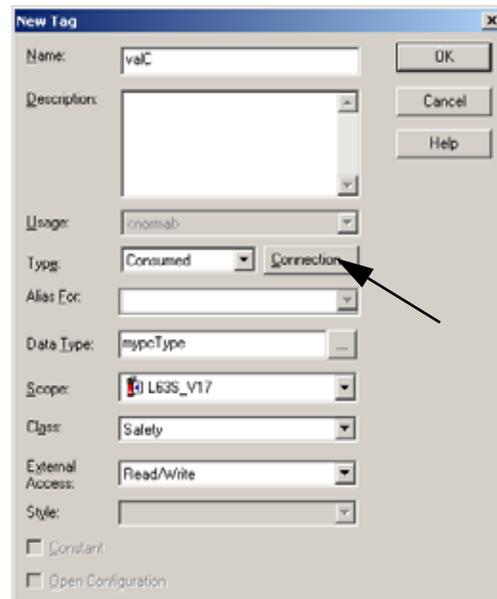


The priority of the safety task is not a safety concern, as the safety task watchdog monitors if the task is interrupted by higher priority task.

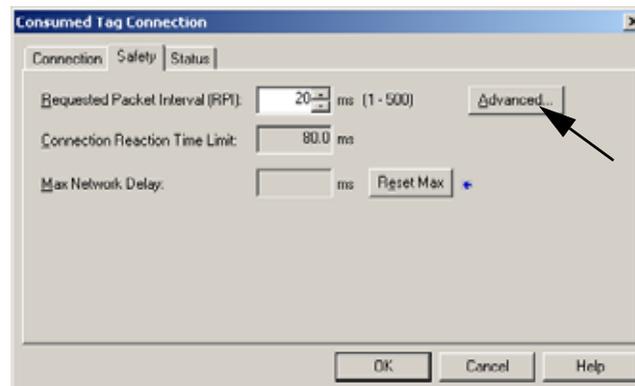
Accessing Produced/Consumed Tag Data

To view or configure safety tag connection data, follow these steps.

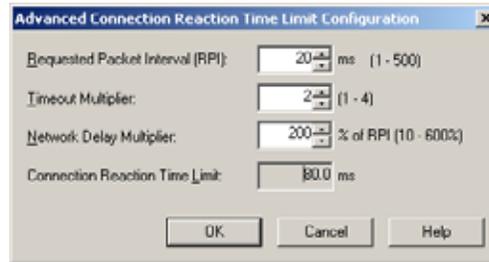
1. In the configuration tree, right-click Controller Tags and choose Edit tags.
2. In the Tag Editor, right-click the name of the tag and choose Edit Properties.
3. Click Connection.



4. Click the Safety tab.



- Click Advanced to view or edit the current settings.



Additional Resources

Refer to these publications for more information. Also, consult the product documentation for your specific module for reaction times associated with CIP Safety I/O modules.

| Resource | Description |
|---|---|
| GuardLogix Controllers User Manual, publication 1756-UM020 | Contains information on configuring delay times and reaction time limits for the input connection, safety task, and output connection |
| 1768 Compact GuardLogix Controllers User Manual, publication 1768-UM002 | |

Checklists for GuardLogix Safety Applications

| Topic | Page |
|---|-------------|
| Checklist for GuardLogix Controller System | 90 |
| Checklist for Safety Inputs | 91 |
| Checklist for Safety Outputs | 92 |
| Checklist for Developing a Safety Application Program | 93 |

The checklists in this appendix are required for planning, programming, and startup of a SIL 3-certified GuardLogix application. They may be used as planning guides as well as during functional verification testing. If used as planning guides, the checklists can be saved as a record of the plan.

The checklists on the following pages provide a sample of safety considerations and are not intended to be a complete list of items to verify. Your particular safety application may have additional safety requirements, for which we have provided space in the checklists.

TIP Make copies of the checklists and keep these pages for future use.

Checklist for GuardLogix Controller System

Checklist for GuardLogix System

Company

Site

Safety Function Definition

| Number | System Requirements | Fulfilled | | Comment |
|--------|--|--------------------------|--------------------------|---------|
| | | Yes | No | |
| 1 | Are you using only the components listed in SIL 3-certified GuardLogix Components on page 14 and on the http://www.rockwellautomation.com/products/certification/safety/ site, with the corresponding firmware release? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 2 | Have you calculated the system's safety response time for each safety chain? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 3 | Does the system's response time include both the user-defined safety task program watchdog (software watchdog) time and the safety task rate/period? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 4 | Is the system response time in proper relation to the process tolerance time? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 5 | Have probability (PFD/PFH) values been calculated according to the system's configuration? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 6 | Have you performed all appropriate functional verification tests? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 7 | Have you determined how your system will handle faults? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 8 | Does each network in the safety system have a unique SNN? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 9 | Is each CIP safety device configured with the correct SNN? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 10 | Have you generated a safety task signature? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 11 | Have you uploaded and recorded the safety task signature for future comparison? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 12 | Following a download, have you verified that the safety task signature in the controller matches the recorded safety task signature? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 13 | Do you have an alternate mechanism in place to preserve the safety integrity of the system when making online edits? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 14 | Have you taken into consideration the checklists for using SIL inputs and outputs listed on pages 91 and 92? | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |

Checklist for Safety Inputs

For programming or startup, an individual checklist can be filled in for every single SIL input channel in a system. This is the only way to make sure that the requirements are fully and clearly implemented. This checklist can also be used as documentation on the connection of external wiring to the application program.

Input Checklist for GuardLogix System

Company

Site

Safety Function Definition

SIL Input Channels

| Number | Input Module Requirements | Fulfilled | | Comment |
|--------|--|--------------------------|--------------------------|---------|
| | | Yes | No | |
| 1 | Have you followed installation instructions and precautions to conform to applicable safety standards? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 2 | Have you performed functional verification tests on the system and modules? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 3 | Are control, diagnostics, and alarming functions performed in sequence in application logic? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 4 | Have you uploaded and compared the configuration of each module to the configuration sent by configuration tool? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 5 | Are modules wired in compliance with PLe/Cat. 4 according to ISO 13849-1? ⁽¹⁾ | <input type="checkbox"/> | <input type="checkbox"/> | |
| 6 | Have you verified that the electrical specifications of the sensor and input are compatible? | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |

(1) For information on wiring your CIP Safety I/O module, refer to the product documentation for your specific module.

Checklist for Safety Outputs

For programming or startup, an individual requirement checklist must be filled in for every single SIL output channel in a system. This is the only way to make sure that the requirements are fully and clearly implemented. This checklist can also be used as documentation on the connection of external wiring to the application program.

Output Checklist for GuardLogix System

Company _____

Site _____

Safety Function Definition _____

SIL Output Channels _____

| Number | Output Module Requirements | Fulfilled | | Comment |
|--------|--|--------------------------|--------------------------|---------|
| | | Yes | No | |
| 1 | Have you followed installation instructions and precautions to conform to applicable safety standards? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 2 | Have you performed functional verification tests on the modules? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 3 | Have you uploaded and compared the configuration of each module to the configuration sent by configuration tool? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 4 | Have you verified that test outputs are not used as safety outputs? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 5 | Are modules wired in compliance with PLe/Cat. 4 according to ISO 13849-1? ⁽¹⁾ | <input type="checkbox"/> | <input type="checkbox"/> | |
| 6 | Have you verified that the electrical specifications of the output and the actuator are compatible? | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |

(1) For information on wiring your CIP Safety I/O module, refer to the product documentation for your specific module.

Checklist for Developing a Safety Application Program

Use the following checklist to help maintain safety when creating or modifying a safety application program.

Checklist for GuardLogix Application Program Development

Company _____

Site _____

Project Definition

| Number | Application Program Requirements | Fulfilled | | Comment |
|--------|--|--------------------------|--------------------------|---------|
| | | Yes | No | |
| 1 | Are you using version 14, or version 16 or later of RSLogix 5000 software ⁽¹⁾ , the GuardLogix system programming software? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 2 | Were the programming guidelines in Chapter 6 followed during creation of the safety application program? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 3 | Does the safety application program contain only relay ladder logic? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 4 | Does the safety application program contain only those instructions listed in Appendix A as suitable for safety application programming? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 5 | Does the safety application program clearly differentiate between safety and standard tags? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 6 | Are only safety tags used for safety routines? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 7 | Have you verified that safety routines do not attempt to read from or write to standard tags? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 8 | Have you verified that no safety tags are aliased to standard tags and vice versa? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 9 | Is each safety output tag correctly configured and connected to a physical output channel? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 10 | Have you verified that all mapped tags have been conditioned in safety application logic? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 11 | Have you defined the process parameters that are monitored by fault routines? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 12 | Have you sealed any safety Add-On Instructions with an instruction signature and recorded the safety instruction signature? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 13 | Has the program been reviewed by an independent safety reviewer (if required)? | <input type="checkbox"/> | <input type="checkbox"/> | |
| 14 | Has the review been documented and signed? | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |
| | | <input type="checkbox"/> | <input type="checkbox"/> | |

(1) RSLogix 5000 software, version 18 or later, supports 1768 Compact GuardLogix controllers.

Notes:

GuardLogix Systems Safety Data

| Topic | Page |
|------------|------|
| PFD Values | 95 |
| PFH Values | 96 |

The following examples show probability of failure on demand (PFD) and probability of failure per hour (PFH) values for GuardLogix 1oo2 SIL 3 systems.

Mission time for GuardLogix controllers is 20 years.

For safety data, including PFD and PFH values for Guard I/O modules, see the manuals for those products.

- POINT Guard I/O Safety Modules User Manual, publication [1734-UM013](#)
- Guard I/O EtherNet/IP Safety Modules User Manual, publication [1791ES-UM001](#)
- Guard I/O DeviceNet Safety Modules User Manual, publication [1791DS-UM001](#)

Data for Rockwell Automation machinery safety products is now available in the form of a library file to be used with the Safety Integrity Software Tool for the Evaluation of Machine Applications (SISTEMA). The library file is available for download at: http://www.marketing.rockwellautomation.com/safety-solutions/en/MachineSafety/ToolsAndDownloads/sistema_download.

PFD Values

Table 14 - Calculated PFD by Proof Test Interval

| Cat. No. | Description | Calculated PFD | | | |
|-----------------------------|-------------------------------|---------------------------|---------------------------|----------------------------|-----------------------------|
| | | 2 Years (17,520 hours) | 5 Years (43,800 hours) | 10 Years (87,600 hours) | 20 Years (175,200 hours) |
| 1756-L6xS and 1756-LSP | GuardLogix Controller | Not applicable | | 5.5E-06 | 1.2E-05 |
| 1756-L7xS and 1756-L7SP | GuardLogix controller | 5.7E-06 | 1.5E-05 | 3.5E-05 | 8.9E-05 |
| 1756-L73SXT and 1756-L7SPXT | GuardLogix XT controller | 5.7E-06 | 1.5E-05 | 3.5E-05 | 8.9E-05 |
| 1768-L43S and 1768-L45S | Compact GuardLogix Controller | 1.1E-06 | 2.7E-06 | 5.7E-06 | 1.2E-05 |

PFH Values

The data in [Table 15](#) applies to proof test intervals up to and including 20 years.

Table 15 - PFH Calculations

| Cat. No. | Description | PFH (1/Hour) |
|-----------------------------|-------------------------------|--------------|
| 1756-L6xS and 1756-LSP | GuardLogix controller | 2.0E-10 |
| 1756-L7xS and 1756-L7SP | GuardLogix controller | 1.2E-09 |
| 1756-L7xSXT and 1756-L7SPXT | GuardLogix XT controller | 1.2E-09 |
| 1768-L43S and 1768-L45S | Compact GuardLogix controller | 2.0E-10 |

RSLogix 5000 Software, Version 14 and Later, Safety Application Instructions

| Topic | Page |
|---|------|
| De-energize to Trip System | 97 |
| Use Connection Status Data to Initiate a Fault Programmatically | 97 |

De-energize to Trip System

When using RSLogix 5000 software, version 14 safety application instructions, all inputs and outputs are set to zero when a fault is detected. As a result, any inputs being monitored by one of the diverse input instructions (Diverse Inputs or Two-hand Run Station) should have normally-closed inputs conditioned by logic similar to the logic in Rung 4 of [Ladder Logic Example 2](#) and [Ladder Logic Example 3](#) on pages [100](#) and [101](#). The exact logic required is both application and input-module dependent. However, the logic must create a safety state of 1 for the normally-closed input of the diverse input instructions.

Use Connection Status Data to Initiate a Fault Programmatically

The following diagrams provide examples of the application logic required to latch and reset I/O failures. The examples show the logic necessary for input only modules, as well as input and output combination modules. The examples use a feature of the I/O modules called Combined Status, which presents the status of all of the input channels in a single boolean variable. Another boolean variable represents the status of all the output channels. This approach reduces the amount of I/O conditioning logic required and forces the logic to shut down all input or output channels on the affected module.

Use the [Input Fault Latch and Reset Flow Chart](#) on page [98](#) to determine which rungs of logic are required for different application situations. [Ladder Logic Example 1](#) shows logic that overwrites the actual input tag variables while a fault condition exists. If the actual input state is required for troubleshooting while the input failure is latched, use the logic shown in [Ladder Logic Example 2](#). This logic uses internal tags that represent the inputs to be used in the application logic. While the input failure is latched, the internal tags are set to their safety state. While the input failure is not latched, the actual input values are copied to the internal tags.

Use the [Output Fault Latch and Reset Flowchart](#) to determine which rungs of application logic in [Ladder Logic Example 3](#) on page [101](#) are required.

Figure 22 - Input Fault Latch and Reset Flow Chart

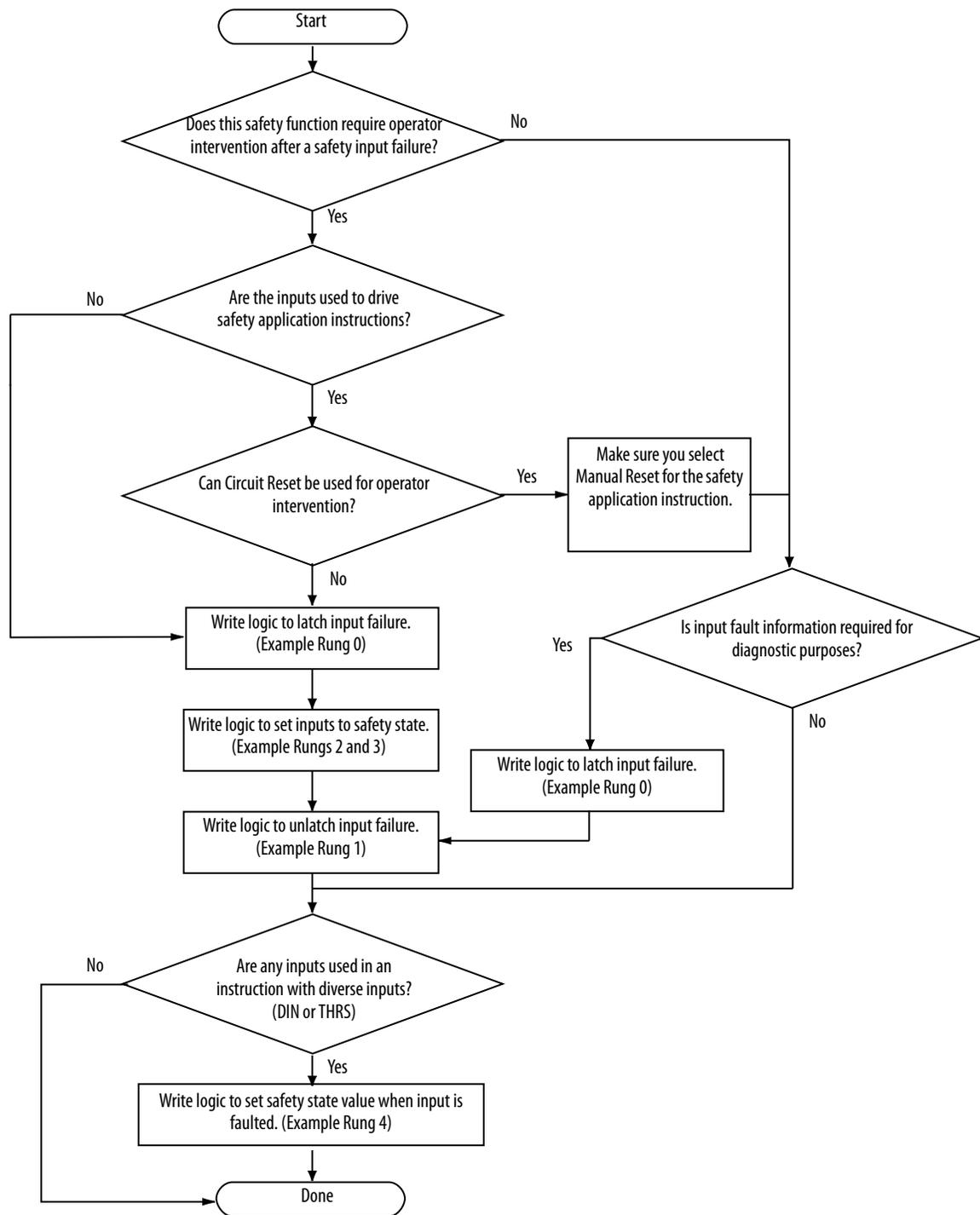


Figure 23 - Ladder Logic Example 1

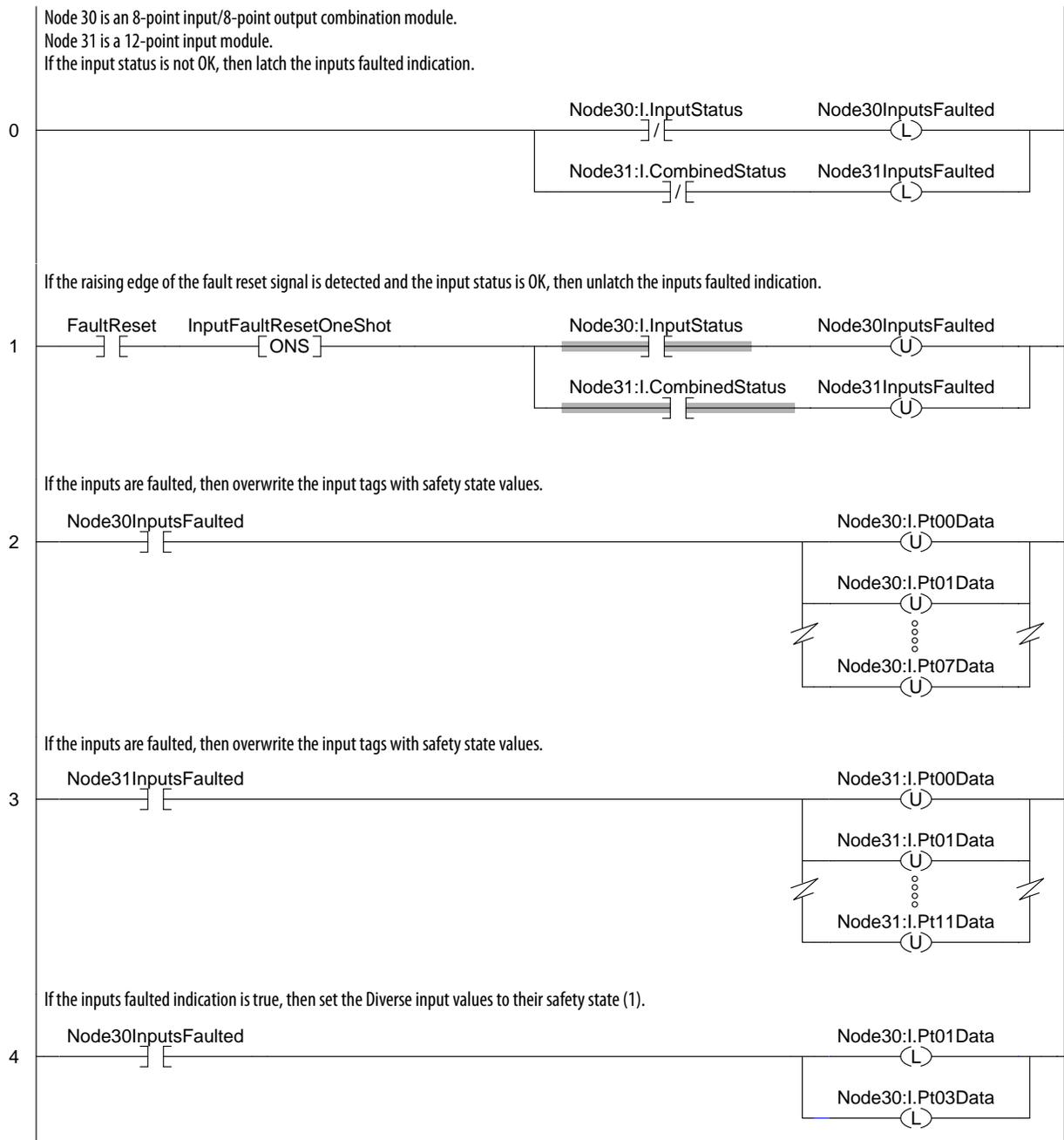


Figure 24 - Ladder Logic Example 2

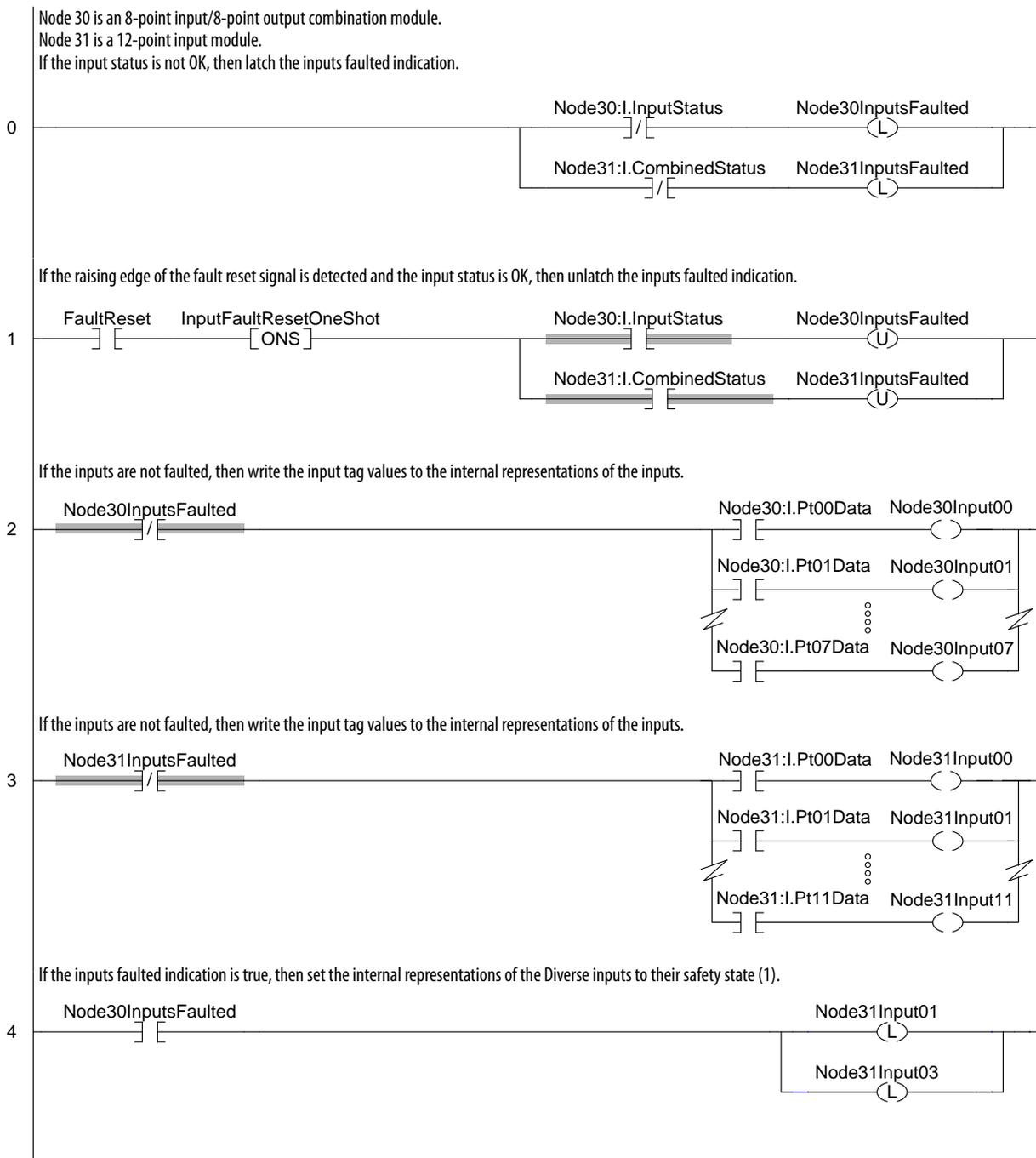


Figure 25 - Output Fault Latch and Reset Flowchart

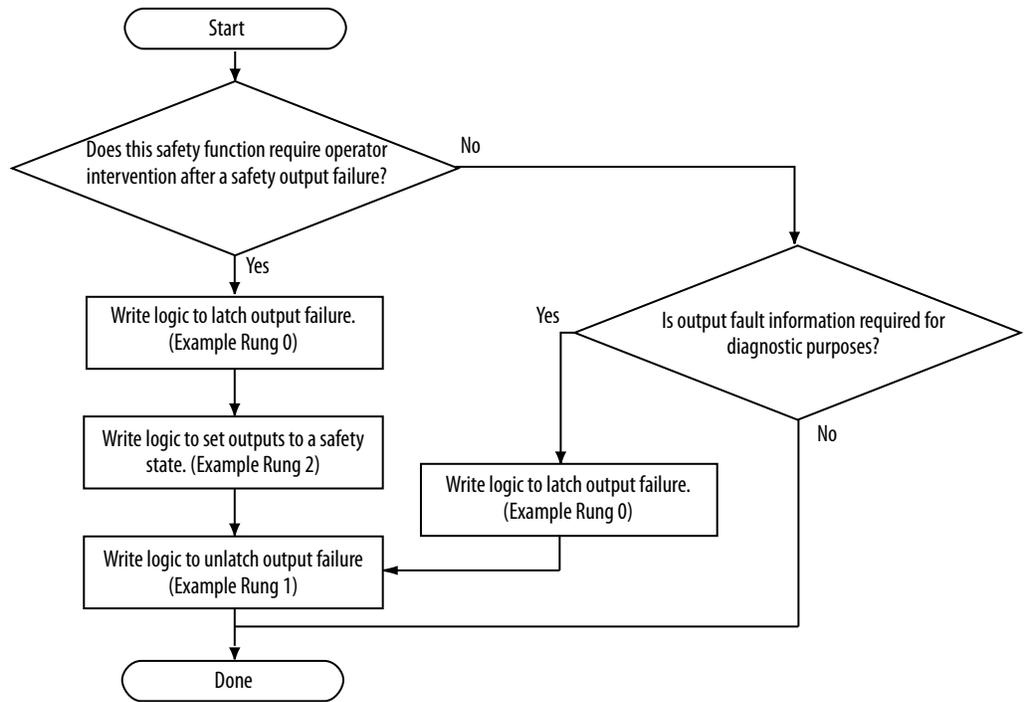
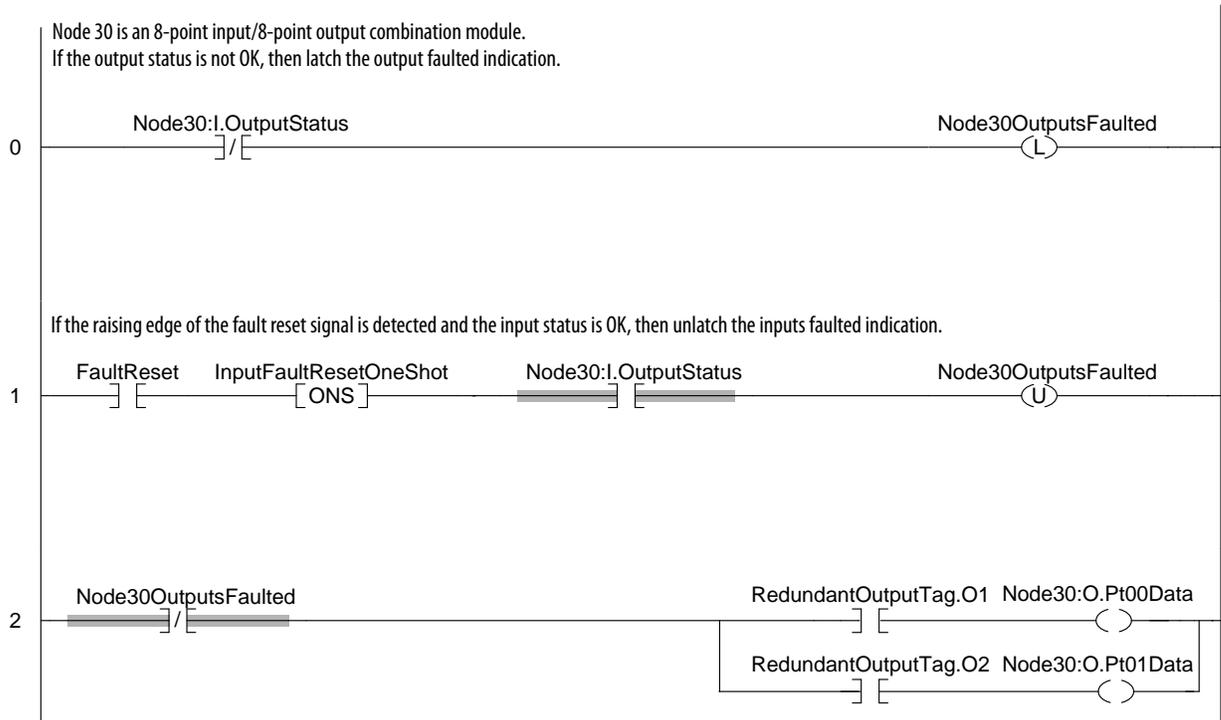


Figure 26 - Ladder Logic Example 3



Notes:

Using 1794 FLEX™ I/O Modules and 1756 SIL 2 Inputs and Outputs with 1756 GuardLogix Controllers to Comply with EN 50156

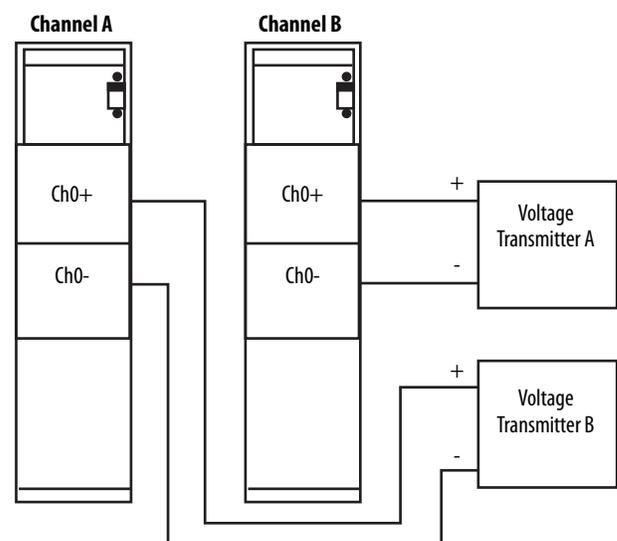
| Topic | Page |
|--|------|
| SIL 2 Dual-channel Inputs (standard side of 1756 GuardLogix controllers) | 103 |
| SIL 2 Outputs Using SIL 3 Guard I/O Output Modules | 105 |
| SIL 2 Outputs Using 1756 or 1794 SIL 2 Output Modules | 105 |
| Safety Functions within the 1756 GuardLogix Safety Task | 106 |

Dual-channel configuration is required for compliance in certain safety-related applications, including burner-related safety functions. These examples provide guidelines for satisfying EN50156 SIL 2 dual-channel requirements with 1- and 2-year proof test intervals.

SIL 2 Dual-channel Inputs (standard side of 1756 GuardLogix controllers)

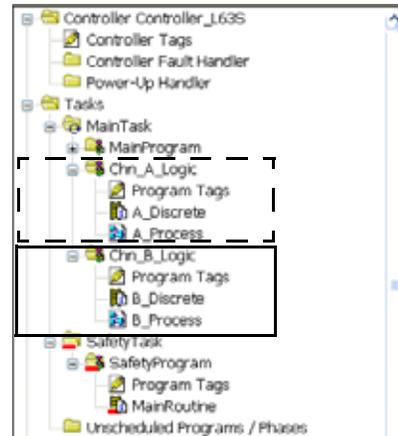
You must implement clear and easily-identifiable separation between both input channels and adhere to all existing SIL 2 requirements as defined in Using ControlLogix in SIL 2 Applications, publication [1756-RM001](#).

Figure 27 - SIL 2 Dual-channel Inputs Example F



SIL 2 Input Data

Keep channel A and channel B input data separate at all times. This example illustrates one method for separating channel A and channel B data in your application. Any logic processing that needs to occur must follow ControlLogix SIL 2 guidelines.

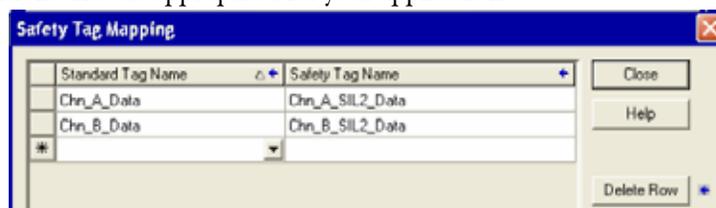


- Follow all rules for 1756 I/O modules as defined in the Using ControlLogix in SIL 2 Applications Safety Reference Manual, publication [1756-RM001](#).
- Follow all rules for 1794 FLEX™ I/O modules as defined in the FLEX I/O System with ControlLogix for SIL 2 Safety Reference Manual, publication [1794-RM001](#).

IMPORTANT Do not perform safety-specific functions within these routines. Safety evaluation must be handled within the 1756 GuardLogix safety task.

Transferring SIL 2 Data Into the Safety Task

To transfer channel A and channel B SIL 2 safety data into the GuardLogix safety task, use the safety tag mapping functionality in RSLogix 5000 software. The tag names used here are for example purposes. Implement and follow naming conventions that are appropriate for your application.



TIP To use the safety tag mapping feature, select Map Safety Tags from the Logic menu in RSLogix 5000 software.

SIL 2 Outputs Using SIL 3 Guard I/O Output Modules

Follow these guidelines for SIL 2 outputs:

- The following list of Guard I/O output modules used for SIL 2 safety outputs must be configured for dual-channel operation (all Guard I/O output modules are approved for use in SIL 2 applications):
 - 1732DS-IB8XOBV4
 - 1791ES-IB8XOBV4
 - 1791DS-IB8XOBV4
 - 1791ES-IB8XOBV4
 - 1791DS-IB4XOW4
 - 1791DS-IB8XOB8
- The following output modules are approved for use in SIL 2 applications, using a single channel safety output:
 - 1734-OB8S
 - 1732ES-IB12XOB4
 - 1732ES-IB8XOB8

SIL 2 Outputs Using 1756 or 1794 SIL 2 Output Modules

When using these SIL 2-rated output modules, you are required to configure your SIL 2 safety outputs as GuardLogix-produced safety tags to comply with the dual-channel requirements of EN 50156.

Create produced safety tags with the SIL 2 outputs that your application requires. GuardLogix produced/consumed safety tags require the first member to be allocated for diagnostics. The first member of a produced/consumed safety connection must be a data type called CONNECTION_STATUS. This example shows a SIL 2 tag with two INT and two BOOL members. Use these SIL 2 safety tags to directly control the 1756 or 1794 SIL 2 outputs.

| Name | Alias For | Base Tag | Data Type | Class | Description | External Access | Constant | Style |
|--------------------------------|-----------|----------|----------------|--------|-------------|-----------------|--------------------------|---------|
| SIL2_Outputs | | | SIL_2_Produced | Safety | | Read/Write | <input type="checkbox"/> | |
| SIL2_Outputs.Connection_Status | | | CONNECTION_STA | Safety | | Read/Write | | |
| SIL2_Outputs.SIL2_TempA | | | INT | Safety | | Read/Write | | Decimal |
| SIL2_Outputs.SIL2_TempB | | | INT | Safety | | Read/Write | | Decimal |
| SIL2_Outputs.SIL2_Valve1 | | | BOOL | Safety | | Read/Write | | Binary |
| SIL2_Outputs.SIL2_Valve2 | | | BOOL | Safety | | Read/Write | | Binary |

TIP In this example, a consumer for the produced tag is not shown. The connection status will show faulted if you don't configure a consumer. However, in this type of configuration, you are not required to monitor the connection status of the produced tag so the fault is not a concern.

- Follow all rules for 1756 I/O modules as defined in the Using ControlLogix in SIL 2 Applications Safety Reference Manual, publication [1756-RM001](#).
- Follow all rules for 1794 FLEX I/O modules as defined in the FLEX I/O System with ControlLogix for SIL 2 Safety Reference Manual, publication [1794-RM001](#).

Safety Functions within the 1756 GuardLogix Safety Task

Follow these guidelines for using SIL 2 and SIL 3 safety functions within the safety task:

- All available safety application instructions may be used.
- SIL 3 safety input modules (that is, Guard I/O modules) may be used with single-channel configuration for SIL 2 safety functions.
- Use of the safety task signature and safety-locking the application is recommended.

IMPORTANT The safety task signature is required to operate in a SIL3 safety function.

IMPORTANT You must not use SIL 2 data to directly control a SIL 3 output.

The following terms and abbreviations are used throughout this manual. For definitions of terms not listed here, refer to the Allen-Bradley Industrial Automation Glossary, publication [AG-7.1](#).

- Add-On Instruction** An instruction that you create as an add-on to the Logix instruction set. Once defined, an Add-On Instruction can be used like any other Logix instruction and can be used across various projects. An Add-On Instruction is composed of parameters, local tags, logic routine, and optional scan mode routines.
- Assemble Edits** You assemble edits when you have made online edit changes to the controller program and want the changes to become permanent because you can test, un-test, or cancel the edits.
- Cancel Edits** Action taken to reject any unassembled online edit changes.
- CIP Safety Protocol** A network communication method designed and certified for transport of data with high integrity.
- Configuration Signature** A unique number that identifies a device's configuration. The configuration signature is made up of an ID number, date, and time.
- Instruction Signature** The instruction signature consists of an ID number and date/timestamp that identifies the contents of the Add-On Instruction definition at a given point in time.
- Nonrecoverable Controller Fault** A fault that forces all processing to be terminated and requires controller power to be cycled from off to on. The user program is not preserved and must be redownloaded.
- Nonrecoverable Safety Fault** A fault, which even though properly handled by the fault handling mechanisms provided by the safety controller and implemented by the user, terminates all safety task processing, and requires external user action to restart the safety task.
- Online** Situation where you are monitoring/modifying the program in the controller.
- Overlap** When a task (periodic or event) is triggered while the task is still executing from the previous trigger.
- Partnership** The primary controller and safety partner must both be present, and the hardware and firmware must be compatible for partnership to be established.
- Pending Edit** A change to a routine that has been made in RSLogix 5000 software, but has not yet been communicated to the controller by accepting the edit.

- Periodic Task** A task that is triggered by the operating system at a repetitive period of time. Whenever the time expires, the task is triggered and its programs are executed. Data and outputs established by the programs in the task retain their values until the next execution of the task or until they are manipulated by another task. Periodic tasks always interrupt the continuous task.
- Primary Controller** The processor in a dual-processor controller that performs standard controller functionality and communicates with the safety partner to perform safety-related functions.
- Recoverable Fault** A fault, which when properly handled by implementing the fault handling mechanisms provided by the controller, does not force user logic execution to be terminated.
- Requested Packet Interval (RPI)** When communicating over a network, this is the maximum amount of time between subsequent production of input data.
- Routine** A set of logic instructions in a single programming language, such as a ladder diagram. Routines provide executable code for the project in a controller. Each program has a main routine. You can also specify optional routines.
- Safety Add-On Instruction** An Add-On Instruction that can use safety application instructions. In addition to the instruction signature used for high-integrity Add-On Instructions, safety Add-On Instructions feature a SIL 3 safety instruction signature for use in safety-related functions.
- Safety Application Instructions** Safety Instructions that provide safety-related functionality. They have been certified to SIL 3 for use in safety routines.
- Safety Component** Any object, task, program, routine, tag, or module that is marked as a safety-related item.
- Safety Instruction Signature** The safety instruction signature is an ID number that identifies the execution characteristics of the safety Add-On Instruction. It is used to verify the integrity of the safety Add-On Instruction during downloads to the controller.
- Safety I/O** Safety I/O has most of the attributes of standard I/O except it features mechanisms certified to SIL 3 for data integrity.
- Safety Network Number (SNN)** Uniquely identifies a network across all networks in the safety system. The end user is responsible for assigning a unique number for each safety network or safety subnet within a system. The safety network number makes up part of the Unique Node Identifier (UNID).
- Safety Partner** The processor in a dual-processor controller that works with the primary controller to perform safety-related functions.

| | |
|----------------------------------|--|
| Safety Program | A safety program has all the attributes of a standard program, except that it can be scheduled only in a safety task. The safety program consists of zero or more safety routines. It cannot contain standard routines or standard tags. |
| Safety Routine | A safety routine has all the attributes of a standard routine except that it is valid only in a safety program and that it consists of one or more instructions suitable for safety applications. (See Appendix A for a list of Safety Application Instructions and standard Logix Instructions that may be used in safety routine logic.) |
| Safety Tags | A safety tag has all the attributes of a standard tag except that the GuardLogix controller provides mechanisms certified to SIL 3 to ensure the integrity of their associated data. They can be program-scoped or controller-scoped. |
| Safety Task | A safety task has all the attributes of a standard task except that it is valid only in a GuardLogix controller and that it may schedule only safety programs. Only one safety task can exist in a GuardLogix controller. The safety task must be a periodic/timed task. |
| Safety Task Period | The period at which the safety task executes. |
| Safety Task Reaction Time | The sum of the safety task period plus the safety task watchdog. This time represents the worst case delay from any input change presented to the GuardLogix controller until the processed output is available to the producing connection. |
| Safety Task Signature | A value, calculated by the firmware, that uniquely represents the logic and configuration of the safety system. It is used to verify the integrity of the safety application program during downloads to the controller. |
| Safety Task Watchdog | The maximum time allowed from the start of safety task execution to its completion. Exceeding the safety task Watchdog triggers a nonrecoverable safety fault. |
| Standard Component | Any object, task, tag, program, and so on, that is not marked as being a safety-related item. |
| Standard Controller | As used in this document, standard controller refers generically to a ControlLogix controller. |
| Symbolic Addressing | A method of addressing that provides an ASCII interpretation of the tag name. |
| System Reaction Time | The worst case time from a safety-related event as input to the system or as a fault within the system, until the time that the system is in the safe state. System Reaction Time includes sensor and activator Reaction Times as well as the Controller Reaction Time. |

Task A scheduling mechanism for executing a program. A task provides scheduling and priority information for a set of one or more programs that execute based on a certain criteria. Once a task is triggered (activated), all of the programs assigned (scheduled) to the task execute in the order in which they are displayed in the controller organizer.

Timeout Multiplier This value determines the number of messages that may be lost before declaring a connection error.

Valid Connection Safety connection is open and active, with no errors.

Numerics

1734-AENT 15, 16, 24
1734-AENTR 16
1756-A10 15
1756-A13 15
1756-A17 15
1756-A4 15
1756-A5XT 15
1756-A7 15
1756-A7XT 15
1756-CN2 15, 24
1756-CN2R 15, 24
1756-CN2RXT 15, 24
1756-DNB 15, 24
1756-EN2F 15, 24
1756-EN2T 15, 24
1756-EN2TR 24
1756-EN2TXT 15, 24
1756-EN3TR 24
1756-ENBT 15, 24
1756-PB72 15
1756-PB75 15
1768-CNB 16, 24
1768-CNBR 24
1768-ENBT 16, 24
1768-PA3 16
1768-PB3 16
1784-CF128 15, 16
1784-SD1 15
1784-SD2 15

A

Add-On Instruction
 certify 73
 instruction signature 75
 safety instruction signature 76
agency certifications 16
application development basics 50
application program
 See program

B

burner-related safety functions 101

C

certifications 16
chassis
 catalog numbers 15
 hardware overview 22

checklist

GuardLogix controller system 26, 88
 program development 91
 SIL 3 inputs 89
 SIL 3 outputs 90

CIP Safety protocol

definition 105
 overview 23
 routable system 33

commissioning life cycle

 51

communication modules

catalog numbers 15
 hardware overview 24

CompactFlash card

 16

configuration signature

 29

connection status

 64

CONNECTION_STATUS

data type 63

control and information protocol

Definition 9

control function

specification 52

ControlNet bridge module

hardware overview 24

D

DeviceNet Safety

communication overview 25

DeviceNet scanner interface module

hardware overview 24

diagnostic coverage

definition 9

E

EN50156 101

EN954-1

CAT 4 9, 11

EtherNet/IP

communication overview 24

EtherNet/IP communication interface module

hardware overview 24

European norm.

definition 9

F

faults

nonrecoverable controller faults 66
 nonrecoverable safety faults 66
 overriding 66
 recoverable 67, 106

firmware revisions

 15, 16

forcing

 58

functional verification tests

 12

G

- get system value (GSV)**
 - definition 9
- GSV instructions** 65
- Guard I/O modules**
 - SIL 2 applications 103

H

- hard faults**
 - recovery 66
- human-to-machine interfaces**
 - use and application 43-45

I

- I/O modules**
 - replacement 29-31
- IEC 61508**
 - Safety Integrity Level 3 (SIL 3) certification 9, 11, 76
- inhibiting a module** 58
- installing a controller** 21
- instruction signature** 75
 - definition 105
- instructions**
 - safety 71
 - safety application 69
- interface**
 - HMI use and application 43-45
- ISO 13849-1** 9, 11

L

- ladder logic safety instructions** 71
- Logix components**
 - SIL 3-certified 14
- Logix system reaction time**
 - calculating 80

M

- mapping tags** 47
- memory card** 15, 16
- metal form instructions** 70

N

- nonrecoverable controller faults** 66, 105
- nonrecoverable safety faults** 66, 105
 - restarting the safety task 66

O

- offline edits** 60
- online**
 - definition 105
- online editing** 57, 60

- output delay time** 28
- overlap**
 - definition 105
- ownership** 29

P

- partnership**
 - definition 105
- peer-to-peer communication** 24
- pending edits** 57
- Performance Level**
 - definition 9
- period task**
 - definition 106
- PLe** 9, 11
- power supplies** 15
 - hardware overview 22
- primary controller**
 - definition 106
 - hardware overview 22
- probability of failure on demand (PFD)** 16-17
 - definition 9
- probability of failure per hour (PFH)** 16-17
 - definition 9
- program**
 - checklist 91
 - download 57
 - editing life cycle 61
 - identification 53
 - offline editing 60
 - online editing 60
 - upload 57
 - verification 54
- programming software** 11
- project**
 - confirmation 55
- proof tests** 12
 - see functional verification tests

Q

- qualifying standard data** 47

R

- reaction time**
 - calculating for system 79
 - safety task 18
 - system 18, 107
- recoverable faults** 67, 106
- reliability burden** 17
- requested packet interval**
 - definition 106
 - range 42
- RSLogix 5000 software**
 - changing your application program 59
 - commissioning life cycle 51
 - version 15, 16

S

safety application instructions 69
 definition 106

safety certifications and compliances 16

safety concept
 assumptions 49

safety consumed tags
 safety network number 35

safety functions
 CIP Safety I/O 27
 Safety Output 28

safety instruction signature 76
 definition 106

Safety Integrity Level (SIL)
 compliance distribution and weight 17
 function example 14
 policy 11-19

Safety Integrity Level (SIL) 3 certification
 Logix components 14
 TÜV Rheinland 12
 user responsibilities 12

Safety Integrity Level 3 (SIL 3) certification 9,
 11, 76

safety network number 34
 definition 106
 manual assignment 34
 out-of-box modules 36
 safety consumed tags 35

safety partner
 configuration 22
 definition 106
 hardware overview 22
 location 22

safety program 45
 definition 107

safety routine 45
 definition 107

safety tags 46
 definition 107
 valid data types 46

safety task
 definition 107
 execution 42
 overview 41
 priority 84
 reaction time 18, 107
 watchdog time 84

safety task period 19
 definition 107
 limitations 41
 overview 19

safety task signature
 definition 107
 deleting 54
 generating 53
 restricted operations 54

safety task watchdog 19
 definition 107
 modifying 19
 overview 19
 setting via RSLogix 5000 19
 timeout 41

safety-locking 56
 default 56
 passwords 56
 restricted operations 56

Secure Digital (SD) card 15

set system variable (SSV) instruction 65

signature history 77

SIL 2
 EN50156 101

software
 changing your application program 59
 commissioning life cycle 51

system reaction time 18
 calculating 79

T

tags
 produced/consumed safety data 46
 Safety I/O 46
 see also safety tags

terminology 9

timeout multiplier 82
 definition 108

U

unique node reference
 defined 34

W

watchdog time 84

X

XT-components 15

Rockwell Automation Support

Use the following resources to access support information.

| | | |
|---|---|---|
| Technical Support Center | Knowledgebase Articles, How-to Videos, FAQs, Chat, User Forums, and Product Notification Updates. | https://rockwellautomation.custhelp.com/ |
| Local Technical Support Phone Numbers | Locate the phone number for your country. | http://www.rockwellautomation.com/global/support/get-support-now.page |
| Direct Dial Codes | Find the Direct Dial Code for your product. Use the code to route your call directly to a technical support engineer. | http://www.rockwellautomation.com/global/support/direct-dial.page |
| Literature Library | Installation Instructions, Manuals, Brochures, and Technical Data. | http://www.rockwellautomation.com/global/literature-library/overview.page |
| Product Compatibility and Download Center (PCDC) | Get help determining how products interact, check features and capabilities, and find associated firmware. | http://www.rockwellautomation.com/global/support/pcdc.page |

Documentation Feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete the How Are We Doing? form at http://literature.rockwellautomation.com/idc/groups/literature/documents/du/ra-du002_-en-e.pdf.

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

Allen-Bradley, ArmorBlock Guard I/O, CompactBlock Guard I/O, CompactLogix, ControlBus, ControlFLASH, ControlLogix, ControlLogix-XT, DCM, FactoryTalk Security, FLEX I/O, Guard I/O, GuardLogix, GuardLogix-XT, Logix5000, POINT Guard I/O, POINT I/O, RSLogix 5000, Rockwell Automation, Rockwell Software, SLC, and SmartGuard are trademarks of Rockwell Automation, Inc.

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

CIP Safety, ControlNet, DeviceNet, and EtherNet/IP are trademarks of ODVA, Inc.

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 1756-RM093I-EN-P - May 2016

Supersedes Publication 1756-RM093H-EN-P - August 2012

Copyright © 2016 Rockwell Automation, Inc. All rights reserved. Printed in the U.S.A.