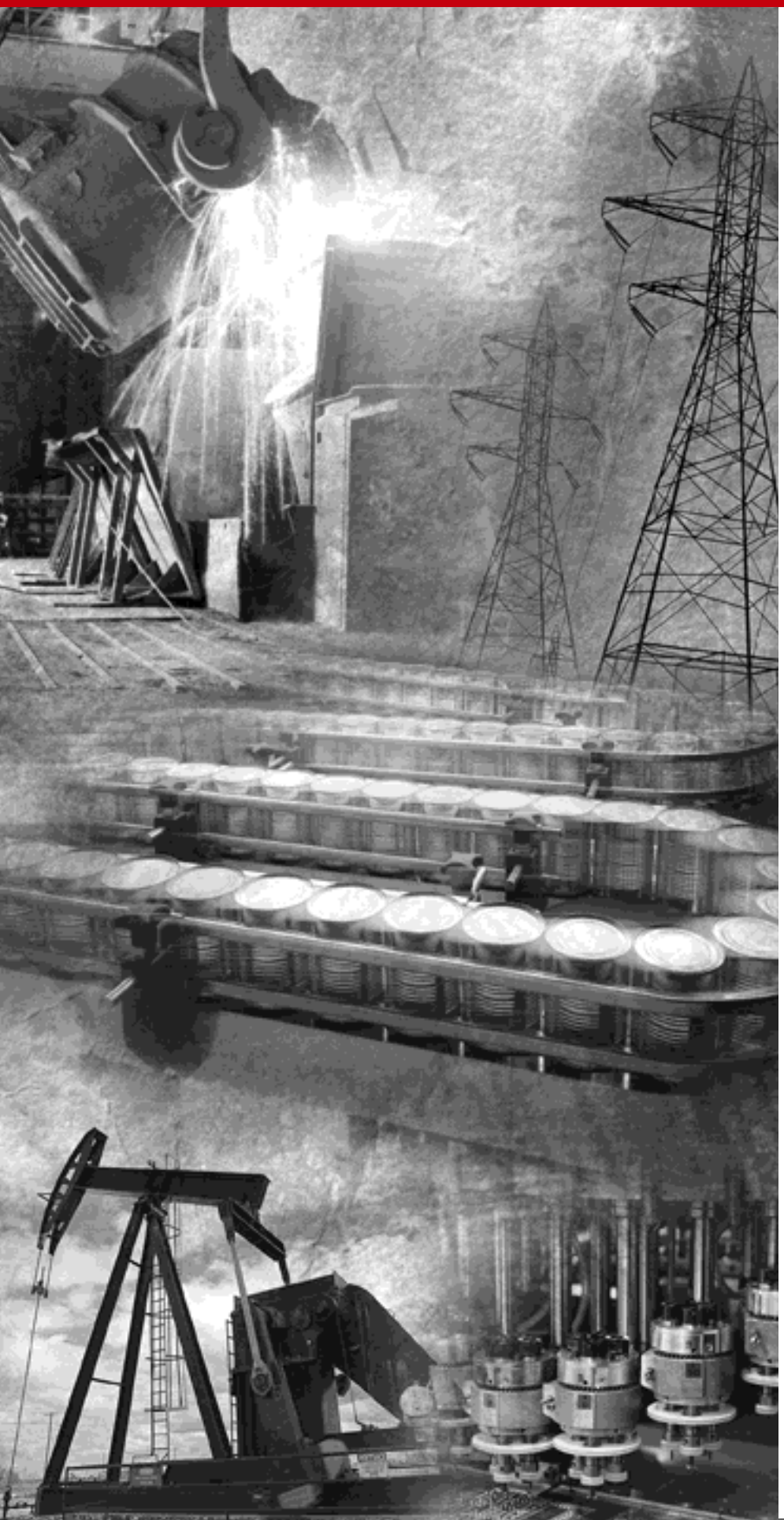


Allen-Bradley

# ControlLogix Programmable Limit Switch Module

1756-PLS

User Manual

Rockwell Automation

## Important User Information

Because of the variety of uses for the products described in this publication, those responsible for the application and use of this control equipment must satisfy themselves that all necessary steps have been taken to assure that each application and use meets all performance and safety requirements, including any applicable laws, regulations, codes and standards.

The illustrations, charts, sample programs and layout examples shown in this guide are intended solely for purposes of example. Since there are many variables and requirements associated with any particular installation, Allen-Bradley does not assume responsibility or liability (to include intellectual property liability) for actual use based upon the examples shown in this publication.

Allen-Bradley publication SGI-1.1, *Safety Guidelines for the Application, Installation and Maintenance of Solid-State Control* (available from your local Allen-Bradley office), describes some important differences between solid-state equipment and electromechanical devices that should be taken into consideration when applying products such as those described in this publication.



Throughout this manual we use notes to make you aware of safety considerations:

**ATTENTION**

Identifies information about practices or circumstances that can lead to personal injury or death, property damage or economic loss

Attention statements help you to:

- identify a hazard
- avoid a hazard
- recognize the consequences

**IMPORTANT**

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

## European Communities (EC) Directive Compliance

If this product has the CE mark it is approved for installation within the European Union and EEA regions. It has been designed and tested to meet the following directives.

### EMC Directive

This product is tested to meet the Council Directive 89/336/EC Electromagnetic Compatibility (EMC) by applying the following standards, in whole or in part, documented in a technical construction file:

- EN 50081-2 EMC — Generic Emission Standard, Part 2 — Industrial Environment
- EN 50082-2 EMC — Generic Immunity Standard, Part 2 — Industrial Environment

This product is intended for use in an industrial environment.

### Low Voltage Directive

This product is tested to meet Council Directive 73/23/EEC Low Voltage, by applying the safety requirements of EN 61131-2 Programmable Controllers, Part 2 - Equipment Requirements and Tests. For specific information required by EN 61131-2, see the appropriate sections in this publication, as well as the Allen-Bradley publication Industrial Automation Wiring and Grounding Guidelines For Noise Immunity, publication 1770-4.1.

This equipment is classified as open equipment and must be mounted in an enclosure during operation to provide safety protection.

## What This Preface Contains

This preface describes how to use this manual.



## Who Should Use This Manual

You must be able to program and operate an Allen-Bradley Control**Logix™** Controller (e.g. Logix5550) and an Allen-Bradley Resolver #846-SJxxxxR3-x (x=optional configuration) to efficiently use your ControlLogix Programmable Limit Switch module.

In this manual, we assume that you know how to use the products mentioned above. If you do not, refer to the related user publications for each product before you attempt to use the Programmable Limit Switch (PLS) module.

## Purpose of This Manual

This manual describes how to install, configure, use, and troubleshoot your ControlLogix PLS module.

**IMPORTANT** In the rest of this manual, we refer to the ControlLogix 1756-Programmable Limit Switch module as the PLS module.

# Related Terms

This manual uses the following terms:

## Terms

| This term: | Means: |
|---|---|
| Connection | A communication mechanism from the controller to another module in the control system |
| ControlBus | The backplane used by the 1756 chassis |
| Data consumer | In the producer/consumer model, a device that consumes data. For example, an output module consumer output data from the owner-controller. |
| Direct connection | An individual I/O connection established between the owner-controller and an I/O module |
| Electronic keying | A feature where modules can be requested to perform an electronic check to make sure that the physical module is consistent with what was configured by the software |
| Field side | Interface between user field wiring and I/O module |
| Listen-only connection | An I/O connection where another controller owns/provides the configuration and any output data for the module. The listener merely connects and receives the data broadcast by the module. An owner-controller must exist to allow a listener. |
| Major revision | A module revision that is updated any time there is a significant functional change to the module resulting in an interface change with software |
| Messaging | The method used communicate configuration, input and output data between the controller and a PLS module |
| Minor revision | A module revision that is updated any time there is a change to the module that does not affect its function or software user interface |
| Module-defined data types | Data types used for configuration in RSLogix 5000 that the module itself creates |
| Network Update Time (NUT) | The smallest repetitive time interval in which data can be sent on a ControlNet network. The NUT ranges from 2ms to 100ms |
| Removable terminal block (RTB) | Field wiring connector for I/O modules |
| Removal and Insertion Under Power (RIUP) | A ControlLogix feature that allows a user to install or remove a module or RTB while power is applied. |
| Requested Packet Interval (RPI) | The maximum amount of time between broadcasts of I/O data for a specific connection |
| RSLogix 5000™ | ControlLogix programming software |
| Service | A system feature that is performed on user demand |
| System side | Backplane side of the interface to the I/O module |
| User-defined data types | Data types used for configuration in RSLogix 5000 that the user must creates |

## Related Products and Documentation

The following table lists related ControlLogix products and documentation:

| Cat. number: | Document title: | Pub. number: |
|---|---|---|
| 1756-PA72, -PB72 | ControlLogix Power Supply Installation Instructions | 1756-5.1 |
| 1756-PA72/B, -PB72/B | ControlLogix Power Supply Installation Instructions | 1756-5.67 |
| 1756-A4, -A7, -A10, -A13, -A17 | ControlLogix Chassis Installation Instructions | 1756-5.69 |
| 1756-L1, -L1M1, -L1M2 | Logix5550 Controller User Manual | 1756-6.5.12 |

If you need more information on these products, contact your local Rockwell Automation/Allen-Bradley distributor, integrator or sales office for assistance. For more information on the documentation, refer to the Allen-Bradley Publication Index, publication SD499.

## Rockwell Automation Support

Rockwell Automation offers support services worldwide, with over 75 sales/support offices, 512 authorized distributors and 260 authorized systems integrators located throughout the United States alone, as well as Rockwell Automation representatives in every major country in the world.

### Local Product Support

Contact your local Rockwell Automation representative for:

- sales and order support
- product technical training
- warranty support
- support service agreements

### Technical Product Assistance

If you need to contact Rockwell Automation for technical assistance, please review the troubleshooting information in Chapter 5 first. If the problem persists, then call your local Rockwell Automation representative.

## Your Questions or Comments on this Manual

If you find a problem with this manual, please notify us of it on the enclosed Publication Problem Report.

## Chapter 1

**What is the ControlLogix Programmable Limit Switch Module?**



## Chapter 2

**Programmable Limit Switch Module Operation Within the ControlLogix System**



## Chapter 3

**ControlLogix Programmable Limit Switch Module Features and I/O Operation**

## Chapter 4

**Installing and Wiring the ControlLogix Programmable Limit Switch Module**



## Chapter 5

**Configuring the ControlLogix Programmable Limit Switch Module**

**Notes:**

# What is the ControlLogix Programmable Limit Switch Module?

## What This Chapter Contains

This chapter describes the ControlLogix PLS module and what you must know and do before you begin to use it.

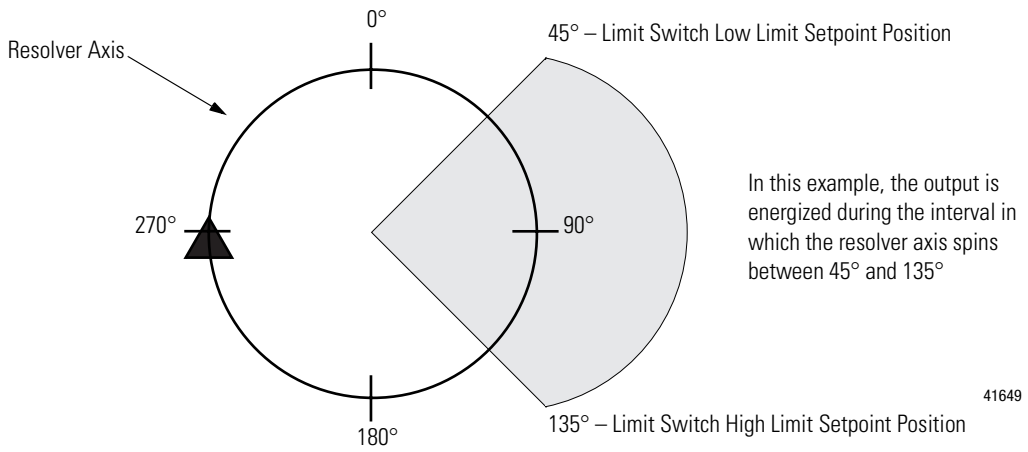


## What is the ControlLogix Programmable Limit Switch Module?

The ControlLogix PLS module energizes and deenergizes an output based on resolver position. For example, the PLS module can monitor machine position via a resolver and activate actuators at various limit switch Low Limit and High Limit setpoint positions.

When the axis position measured by a resolver reaches the predefined Low Limit setpoint, the PLS module energizes the output. As the axis continues its rotation, the resolver reaches the High Limit setpoint, and the PLS module deenergizes the output.

The graphic below shows a resolver axis with a limit switch ON setpoint of 45° and OFF setpoint of 135°.



The following is a list of the features available on the ControlLogix PLS module:

- 16 current-sourcing outputs at 10-31.2V dc with 1A maximum per output - 4 groups of 4 outputs with separate power and common
- 16 single-ended 10-31.2V dc inputs - 4 groups of 4 inputs with separate common
- Class I Division 2, UL, CSA and CE Agency Certification

## Using A Programmable Limit Switch Module in the ControlLogix System

A PLS module energizes and deenergizes outputs for industrial applications. The module interfaces with a ControlLogix Logix5550 controller.

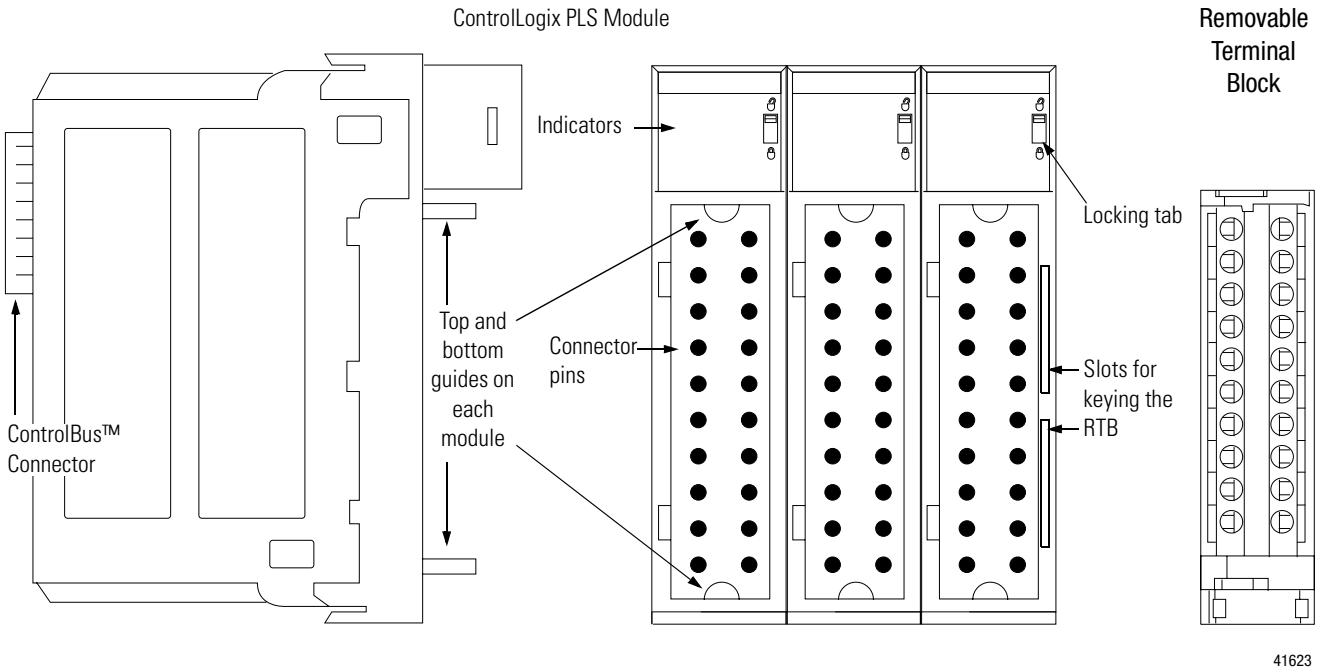
A ControlLogix PLS module mounts in a ControlLogix chassis and uses a Removable Terminal Block (RTB) to connect all field-side wiring.

Before you install and use your module you should have already:

- installed and grounded a 1756 chassis and power supply. To install these products, refer to publications 1756-5.1 and 1756-5.2.
- ordered and received an RTB and their components for your application.

**IMPORTANT** RTBs are not included with your module purchase and must be ordered separately.

## Features of the ControlLogix Programmable Limit Switch Modules




**ControlBus connector** - The backplane interface for the ControlLogix system connects the module to the ControlBus backplane.

**IMPORTANT** Although the PLS module is a 3-slot module, the module only connects to the ControlLogix backplane in the middle slot.

**Connectors pins** - Resolver, input/output, power and grounding connections are made to the module through these pins with the use of an RTB.

**Locking tab** - The locking tab anchors the RTB on the module, maintaining wiring connections.

**Slots for keying** - Mechanically keys the RTB to prevent inadvertently making the wrong wire connections to your module.

**Status indicators** - Indicators display the status of communication, module health and presence of input/output devices. Use these indicators to help in troubleshooting.

**Top and bottom guides** - Guides provide assistance in seating the RTB onto the module.

## Preventing Electrostatic Discharge

This module is sensitive to electrostatic discharge.

**ATTENTION**

Electrostatic discharge can damage integrated circuits or semiconductors if you touch backplane connector pins. Follow these guidelines when you handle the module:

- Touch a grounded object to discharge static potential
- Wear an approved wrist-strap grounding device
- Do not touch the backplane connector or connector pins
- Do not touch circuit components inside the module
- If available, use a static-safe work station
- When not in use, keep the module in its static-shield box

## Removal and Insertion Under Power

These modules are designed to be installed or removed while chassis power is applied.

**ATTENTION**

When you insert or remove a module while backplane power is applied, an electrical arc may occur. An electrical arc can cause personal injury or property damage by:

- sending an erroneous signal to your system's field devices causing unintended machine motion or loss of process control.
- causing an explosion in a hazardous environment.

Repeated electrical arcing causes excessive wear to contacts on both the module and its mating connectors. Worn contacts may create electrical resistance that can affect module operation.

Pulling the PLS module from the chassis while power is applied causes the message instructions carrying configuration, input and output data to fail with an error. Removal may also cause the module to lose any previous configuration.

If you remove the PLS module from the chassis while under power, in addition to considering the warnings above, you must also remember to reconfigure the module upon reinsertion.

## Chapter Summary and What's Next

In this chapter you learned about:

- what the ControlLogix PLS module is
- using the PLS module in the ControlLogix system
- preventing electrostatic discharge
- removing and inserting the module under power

Move on to Chapter 2 to learn about PLS module features and I/O operation.

## Notes:

# Programmable Limit Switch Module Operation Within the ControlLogix System

## What This Chapter Contains

This chapter describes how the PLS module works in a ControlLogix system.



## Ownership and Connections

Every PLS module in the ControlLogix system must be owned by a Logix5550 Controller. This owner-controller stores configuration data for every PLS module that it owns. Other controllers may also talk to the PLS module through the owner-controller.

The owner-controller sends configuration data to the PLS module, defining the module's behavior. Each PLS module continuously maintains communication with its owner-controller during normal operation. When connections are severed or compromised, the PLS module performs as configured, either setting all outputs to reset (ON or OFF) or continuous operations. For more information on how to configure the PLS module, see Chapter 5.

## Using RSNetWorx and RSLogix 5000

When a PLS module is created in RSLogix 5000, the I/O configuration portion of the software generates the module-defined configuration data structures and tags for that PLS module. Additionally, you must generate the user-defined configuration data structures and tags for the PLS module. For more information on configuration data structures, see Chapter 5.

Data structures and tags are generated whether the module is located in a local or remote chassis. A remote chassis contains the PLS module but not the module's owner-controller.

**IMPORTANT** Application-specific configuration data is transferred to the controller during the program download and sent to the PLS module during the initial power-up. After PLS module operation has begun, you must use ladder logic and message instructions to make configuration changes.

### Enabling PLS Module Operation in a Remote Chassis

PLS modules in the same chassis as the controller are ready to run as soon as the program download is complete. But you must schedule the ControlNet network in RSNetWorx to enable PLS modules in the remote chassis.

RSNetWorx establishes a Network Update Time (NUT) for ControlNet that is compliant with the desired communications options specified for each module during configuration.

If you are not using PLS modules in a remote chassis, running RSNetWorx is not necessary. However, anytime a controller references an PLS module in a remote chassis, RSNetWorx must be run to configure ControlNet.

Follow these general guidelines when configuring PLS modules:

1. Configure all PLS modules for a given controller using RSLogix 5000 and download that information to the controller.

2. If the PLS configuration data references a module in a remote chassis, run RSNetWorx to schedule ControlNet.

RSNetWorx **must** be run whenever a new module is added to a remote chassis. When a module is permanently removed from a remote chassis, it is recommended that RSNetWorx be run to optimize the allocation of network bandwidth.

## Direct Connections

A **direct connection** is a real-time data transfer link between the controller and the device that occupies the slot that the configuration data references. When module configuration data is downloaded to an owner-controller, the controller attempts to establish a direct connection to each of the modules referenced by the data. One of the following events occurs:

- If the data is appropriate to the module found in the slot, a connection is made and operation begins.
- If the configuration data is not appropriate, the data is rejected and an error message displays in the software. In this case, the configuration data may be inappropriate for a number of reasons. For example, a module's configuration data may be appropriate except for a mismatch in electronic keying that prevents normal operation.

The controller monitors its connection with a module and detects any break in the connection, such as removal of the module from the chassis while under power. If RSLogix 5000 is connected to the controller, the failure is graphically displayed.

## Programmable Limit Switch Module Input Operation

In most traditional industrial applications, controllers poll inputs to obtain their status. Retrieving input status occurs during the normal I/O program scan.

ControlLogix PLS modules do not follow the traditional operational manner. A PLS module is not scanned by its owner-controller after a connection is established. Instead, the PLS module periodically multicasts its status to the controller.

PLS module communication behavior varies depending upon whether it operates in a local chassis or in a remote chassis. The following sections detail the differences in data transfers between these set-ups.

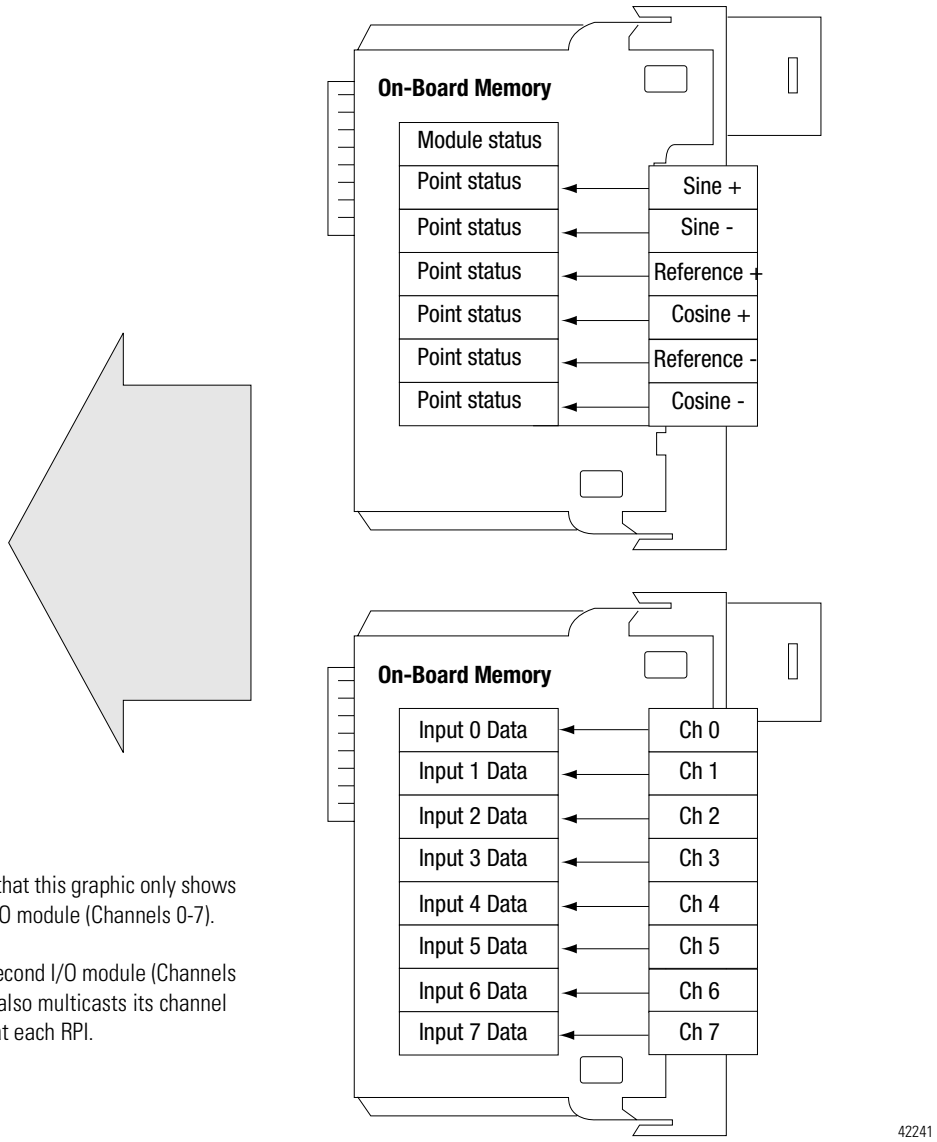
## PLS Inputs' Operation While Module is Located in a Local Chassis

A PLS module multicasts its data periodically, depending on the module configuration and location. The data consumer (i.e. an owner-controller) is responsible for knowing that the format of the new data is integers.

### *Requested Packet Interval (RPI)*

This configurable parameter instructs the module to multicast its channel and status data to the local chassis backplane at specific time intervals. Because the PLS module occupies 3 slots, the contents of each is multicast at the RPI.

The RPI instructs the module to multicast the **current contents** of its on-board memory when the RPI expires, (i.e. the module does not update its channels prior to the multicast).




Note that this graphic only shows one I/O module (Channels 0-7).

The second I/O module (Channels 8-15) also multicasts its channel data at each RPI.

**IMPORTANT** The RPI value is set during the initial module configuration using RSLogix 5000. This value can be adjusted when the controller is in Program mode.

## PLS Inputs' Operation While Module is Located in a Remote Chassis

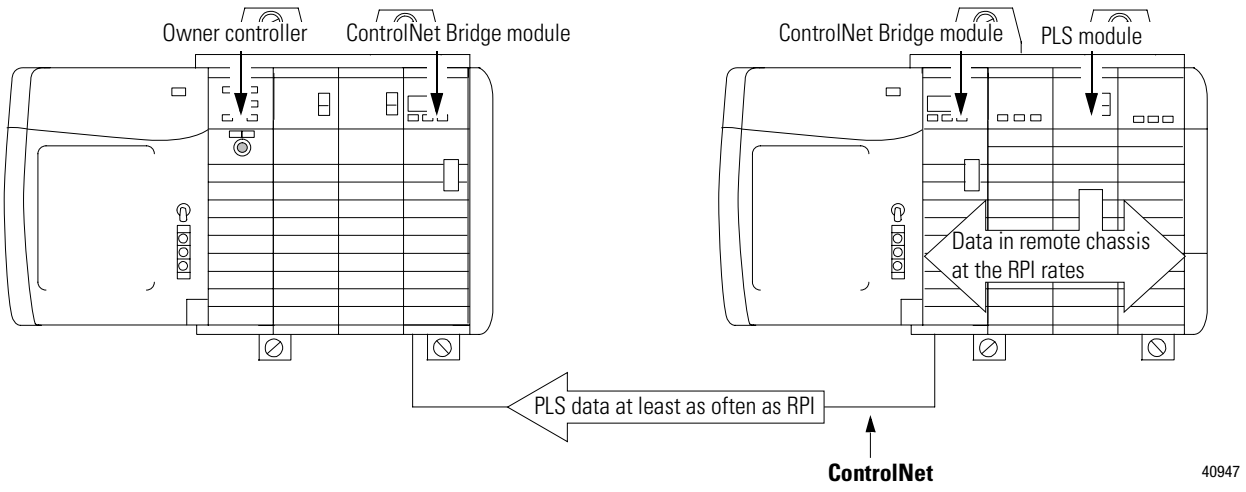
If a PLS module resides in a remote chassis, the role of the RPI changes slightly with respect to sending data to the owner-controller.

The RPI not only defines when the module multicasts data **within its own chassis** (as described in the previous section), but also determines how often the owner controller will receive it over the network.

When an RPI value is specified for an PLS module in a remote chassis, in addition to instructing the module to multicast data within its own chassis, the RPI also "reserves" a spot in the stream of data flowing across the ControlNet network.

The timing of this "reserved" spot may or may not coincide with the exact value of the RPI, but the control system guarantees that the owner controller receives data **at least as often** as the specified RPI.

**PLS Module in Remote Chassis with RPI Reserving a Spot in Flow of Data**



## Programmable Limit Switch Module Outputs' Operation

An owner-controller sends output data to PLS module outputs when either one of two things occur:

- at the end of every one of its program scans (local chassis only) and/or
- at the rate specified in the module's RPI

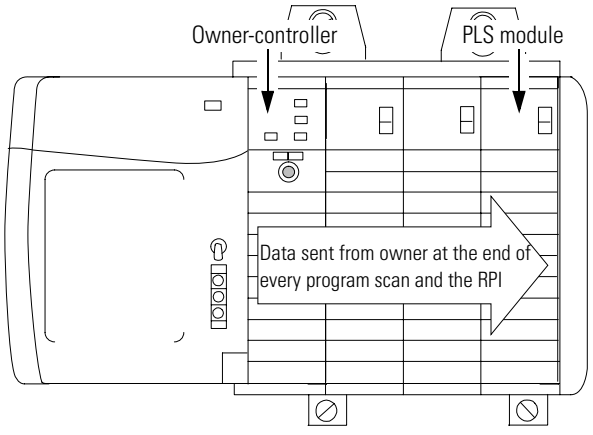
When the PLS module resides in a remote chassis, the owner-controller sends data to the PLS module outputs **only** at the RPI rate specified for the module. Updates are not performed at the end of the owner-controller's program scan.

When the PLS module receives data from the controller, it immediately multicasts the output commands it received to the rest of the system. The actual output data is echoed by the PLS module as input data and multicast back onto the network. This is called **Output Data Echo**. The Output Data Echo also may contain fault and diagnostic information, depending on the module type.

**IMPORTANT** In this Producer/Consumer model, the PLS module is the Consumer of the controller's output data and the Producer of the data echo.

### PLS Outputs' Operation While Module is Located in a Local Chassis

When specifying an RPI value for a PLS module, you are instructing the owner-controller when to broadcast the output data to the module. If the module resides in the same chassis as the owner-controller, the module will receive the data almost immediately after the owner-controller sends it (backplane transfer times are small).



Depending on the value of the RPI, with respect to the length of the program scan, the PLS module can receive and "echo" data multiple times during one program scan.

## PLS Outputs' Operation While Module is Located in a Remote Chassis

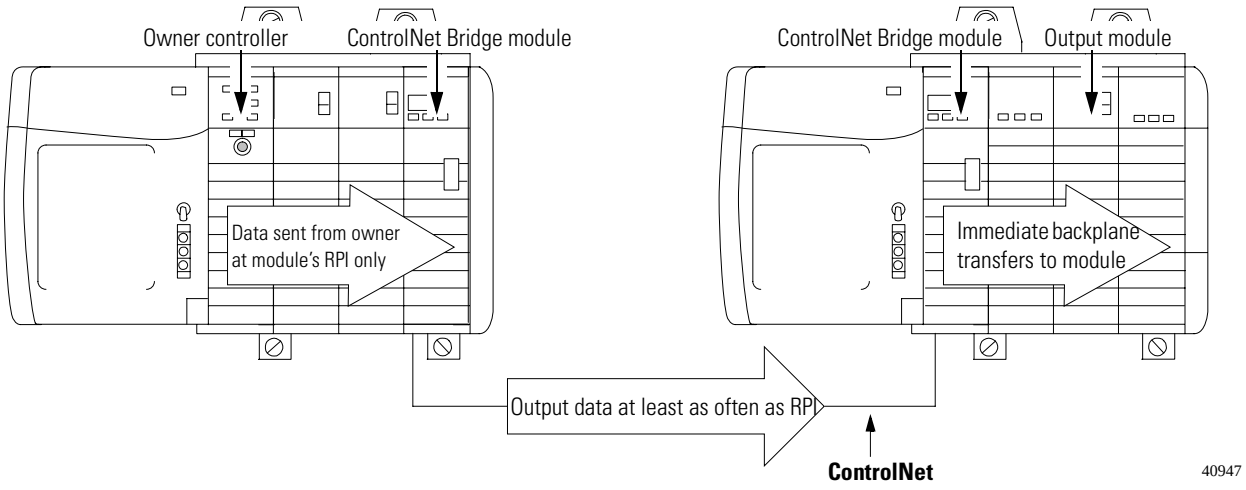
If a PLS module physically resides in a chassis other than that of the owner-controller (i.e. a remote chassis connected via ControlNet), the owner-controller sends data to the PLS module outputs **only** at the RPI rate specified. Updates are **not** performed at the end of the controller's program scan.

In addition, the role of the RPI for a remote PLS module changes slightly, with respect to getting data from the owner-controller.

When an RPI value is specified for a PLS module in a remote chassis, in addition to instructing the owner-controller to multicast the output data within its own chassis, the RPI also "reserves" a spot in the stream of data flowing across the ControlNet network.

The timing of this "reserved" spot may or may not coincide with the exact value of the RPI, but the control system will guarantee that the output module will receive data **at least as often** as the specified RPI.

**Output Module in Remote Chassis with Data Coming At Least as Often as RPI**



## Listen-Only Mode

Any controller in the system can **listen** to the data produced by any PLS module even if the controller does not own the module (i.e. it does not have to contain the module's configuration data to listen to the module).

During the PLS module creation process in RSLogix 5000, you can specify the 'Listen-Only' Communication Format. For more information on Communication Format, see page 5-7.

Choosing 'Listen-Only' mode allows the controller and module to establish communications without the controller sending any configuration data. In this instance, another controller owns the PLS module.

**IMPORTANT** Controllers using the Listen-Only mode continue to receive data multicast from the PLS module as long as a connection between an owner-controller and PLS module is maintained. If this connection between the owner-controller and the PLS module is broken, the PLS module stops multicasting data and connections to all 'Listening controllers' are also broken.

## Chapter Summary and What's Next

In this chapter you learned about:

- ownership and connections
- direct connections
- PLS module operations in a local chassis
- PLS module operations in a remote chassis
- listen-only mode

Move to Chapter 3 to learn about ControlLogix Programmable Limit Switch module features and I/O operation.

# ControlLogix Programmable Limit Switch Module Features and I/O Operation

## What this Chapter Contains

This chapter describes features of the ControlLogix PLS module.



## Determining Resolver Compatibility

ControlLogix PLS modules connect to resolvers and switch outputs based on that resolver's position. PLS modules can only be used with the following resolver:

- Allen-Bradley Resolver #846-SJxxxx-R3 (x=customer options)

The PLS module operates as a high speed resolver input module. The module converts the resolver SIN and COS signals to the axis position.

The PLS module calculates resolver RPM. Positive RPM indicates counterclockwise direction, and negative RPM indicates clockwise direction, as viewed from the resolver shaft. The PLS module is capable of detecting 0.088 degree changes in a resolver's position at 1800 RPM.

# General Features of the Programmable Limit Switch Module

The following features are available with the PLS module:

## Removal and Insertion Under Power (RIUP)

The PLS module may be inserted and removed from the chassis while power is applied. This feature allows greater availability of the overall control system because, while the module is being removed or inserted, there is no additional disruption to the rest of the controlled process.

**IMPORTANT**

Pulling the PLS module from the chassis while power is applied causes the message instructions carrying configuration, input and output data to fail with an error. Removal may also cause the module to lose any previous configuration.

If you remove the PLS module from the chassis while under power, in addition to considering the warning above, you must also reconfigure the module upon reinsertion.

## Module Fault Reporting

The PLS module provide both hardware and software indication when a module fault has occurred. Module status indicators and fault status registers notify the user of fault conditions.

This feature allows you to determine how your module has been affected and what action should be taken to resume normal operation.

## Sink Inputs and Source Outputs

The PLS module uses sinking input points and sourcing output points.

## LED Status Information

The ControlLogix PLS module has LED indicators on the front of the module that allow you to check the module health and operational status.

The following status can be checked with the LED indicators:

- **Input point status** - displays indicate the individual status of each input point and the status of the I/O module
- **Output point status** - displays indicate the individual status of output points and the status of the I/O module
- **Resolver module status** - display indicates the status of the resolver module

For examples of LED indicators on ControlLogix analog I/O modules, see chapter 5.

## Full Class I Division 2 Compliance

ControlLogix PLS modules maintain CSA Class I Division 2 system certification. This allows the ControlLogix system to be placed in an environment other than only a 100% hazard free environment.

**IMPORTANT** Modules should not be pulled under power, nor should a powered RTB be removed, when a hazardous environment is present.

## CE/CSA/UL Agency Certification

ControlLogix PLS modules that have obtained CE/CSA/UL agency certification are marked as such.

### Fully Software Configurable

Changing tag values and sending the tag to the PLS module via message instructions causes the PLS module to perform functions, such as setting a Limit Switch offset position or adjusting output behavior according to the resolver's RPM.

There are two categories of PLS data types.

- **Module-defined data types**
  - Input data structure
  - Output data structure
  - Configuration data structure
- **User-defined data types**
  - Limit switch data structure
  - Offset data structure
  - Registration data structure

For a full explanation of the PLS module data structures, see Chapter 5.

## Measuring Position

The PLS module measures position using a resolver. Resolver values range from 0 to 4095 for each rotation. Use scaling and home position to convert raw resolver position to machine position in desired engineering units. By using engineering units, you can configure the PLS module to report position and to operate using units meaningful to your application.

- Scaling
- Home Position

## Scaling

Scaling, also known as the unwind value, defines the rollover value for the resolver. You can configure the PLS module to operate in Engineering units, such as degrees.

### *Rollover Counts*

Rollover Counts represent the highest number of counts the PLS module can record before wrapping around to zero. The maximum number of counts per resolver rotation is 4095.

### *Rollover Position*

Rollover Position represents the highest position the PLS module can record before wrapping around to zero. For example, to measure in degrees, set the Rollover Position to 359.

**Table 3.A**
**Example Rollover Positions and Counts**

| Measurement Units: | Rollover Position: | Rollover Counts: |
|---|---|---|
| Degrees | 359 | 4095 |
| Tenths of degrees | 3599 | 4095 |
| Percent | 99 | 4095 |
| Hundredth of percent | 9999 | 4095 |
| Millimeters on a 150mm product | 149 | 4095 |

## Home Position

Three features determine a relative zero point, also called home position, for the resolver's axis. Use the following features to synchronize (calibrate or home) the PLS position measured from the resolver to the machine's position:

- Zero Offset
- Preset
- Nudge Up/Down

### *Zero Offset*

Use this feature when you know the starting point in the resolver's axis. For example, if the resolver zero position is at a machine's 10° point, to match the PLS zero position to the machine zero position, you can configure the PLS module Zero Offset value to 10°.

The Zero Offset value is limited to +/- the Rollover Position. For example, if you are using degrees as your engineering units, the maximum position is 359°. In this case, the Zero Offset can be configured for values between -359 and +359.

The PLS module does not save Zero Offset values after power-down. The value must be saved in the Logix5550 controller and downloaded to the PLS module with configuration data.

### *Preset*

Use the Preset to set the relative starting point for the resolver's axis by the energizing of an input.

For example, the PLS module can be configured so that when an input turns ON, the module records the resolver position in its axis at 40°. The PLS module subtracts 40 from zero. The resulting value (-40°) is the Preset Event. The PLS module replaces the current Zero Offset value with this number.

When the Preset occurs, the PLS changes the current Position to its Preset predefined value. The Zero Offset value is recalculated by the PLS module. The new zero offset value is temporary and remains until the next occurrence of the Preset.

**Preset Input** configures the PLS module to use a specific input to signal the Preset event.

You must perform the following steps to use the Preset:

1. Set the **Preset** value - For example, if you want the home position to equal zero when the input turns ON, set this value to zero.

2. Set the **Preset Input** - For example, if you want input 3 to trigger the Preset Event, set this value to 3.

3. Enable the Preset feature (i.e. set the **Arm Preset** bit to 1)

**IMPORTANT** The Preset value is limited to less than or equal to the Rollover Position. For example, if you are using degrees as your position units, the value of the Preset can range from 0° to 359°.

### *Nudge Up/Down*

The Nudge Up/Down functions allow you to precisely adjust the PLS setpoints to account for minor changes in the machine set-up.

For example, during a shift, a resolver may have changed its position by +5° due to slippage. Nudge Up allows you to correct the setpoints by nudging up to the original position.

The **Nudge Up Offset** value determines the number of units that are added to the Zero Offset when Nudge Up Input turns ON.

For example, if your application is using degrees for its position units, you can configure the Nudge Up Offset so that each time you energize the Nudge Up Input, the position change is +1°. Thus, to move the zero offset 5°, you press the Nudge Up button 5 times.

**Nudge Up Input** informs the PLS module which input on the module is adjusted when using the Nudge Up Offset.

You must perform the following steps to use the Nudge Up:

1. Set the **Nudge Up Offset** value
2. Set the **Nudge Up Input**
3. Enable the Nudge Up feature (i.e. set the **Arm Nudge Up** bit to 1)

The **Nudge Down Offset** value determines the number of units, engineering or raw, that are added to the Zero Offset when Nudge Down Input turns ON. This function works similarly to the Nudge Up Offset.

**IMPORTANT** Enter a negative number in the Nudge Down Offset to make the Nudge Down Offset work in the opposite direction as Nudge Up.

**Nudge Down Input** informs the PLS module which input on the module is adjusted when using the Nudge Up Offset.

You must perform the following steps to use the Nudge Down:

1. Set the **Nudge Down Offset** value
2. Set the **Nudge Down Input**
3. Enable the Nudge Down feature (i.e. set the **Arm Nudge Down** bit to 1).

## Configuring Limit Switch Operations

The PLS module provides up to 16 limit switches. There is a configuration profile associated with each limit switch. The PLS module must be enabled before a limit switch can work, even after limit switch configuration has been downloaded to the PLS module.

**IMPORTANT** The PLS module does not know when the last limit switch configuration was downloaded. In writing ladder logic to configure the module, make sure that the configuration data for each limit switch has been downloaded before enabling the module.

The following factors affect limit switch operation:

- Position-Based Limit Switch Operation
- Enable Logic
- Speed Compensation
- Limit Switch-Specific Home Position

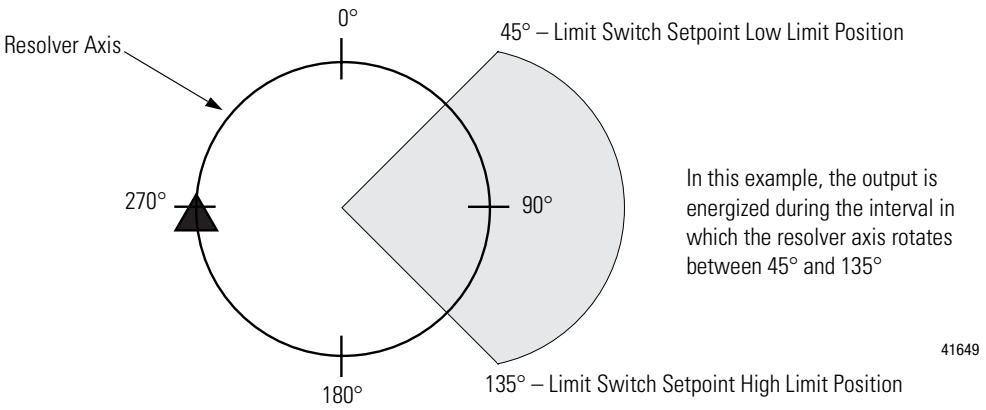
The **Output ID** informs the PLS module which output point you are configuring. When configuration information is downloaded from the Logix5550 controller to the PLS, the Output ID designates which output uses the message's configuration information.

### Position-Based Limit Switch Operation

You can configure a single limit switch with setpoints to turn ON and OFF once, multiple times, or cyclically during a single resolver rotation.

#### *Low and High Limit Setpoints*

You can configure Low Limit and High Limit setpoints to determine when an output is energized. This feature allows you to set particular points in the resolver's rotation where the output specified by the Output ID turns ON and OFF, as shown below.

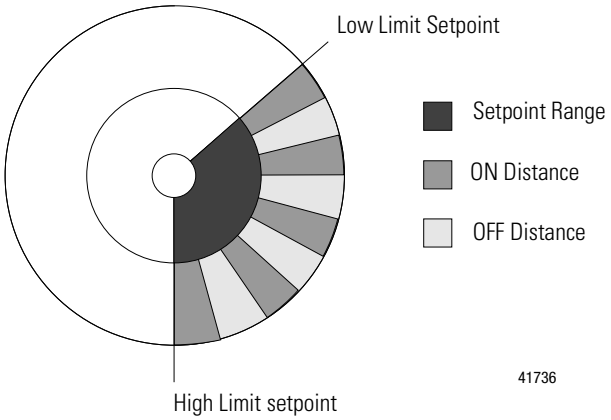


You can configure multiple pairs of setpoints in a single resolver rotation for a single output. For example, you can configure an output to turn ON at 15°, OFF at 45°, ON at 60°, OFF at 100° and so forth with each Low Limit and High Limit pair representing a single setpoint.

### *Stitching Setpoints*

Stitch Setpoints allow you to turn outputs ON for a series of specific distance intervals without configuring separate ON-OFF setpoints for each interval. When using this feature, the series of OFF to ON to OFF transitions occur within the configured Low Limit-High Limit
setpoint region.

For example, starting at 45°, you may want to turn an output ON for 15°, then turn it OFF for 15° and repeat this sequence until 180°, as shown
below. You can configure ON-OFF setpoints for each transition or use Cyclic Setpoints.




When using  Stitching setpoints, you configure the following fields:

- Enable the **Enable Stitching** setpoints - enables the feature
- Setpoints **Low Limit** Position - the beginning of the cycle window
- Setpoints **High Limit** Position - the ending of the cycle window
- Setpoints **ON Stitch Distance** - the distance that the output remains ON
- Setpoints **OFF Stitch Distance** - the distance that the output remains OFF before turning ON again

### *Output Invert*

The Output Invert function causes the PLS module to invert the state of an output channel.

For example, you can configure an output to turn ON while the resolver passes between two positions. If Output Invert is enabled, the channel that has been configured to turn ON will turn OFF while between Low Limit and High Limit.

You must set the Output Invert bit to 1 to enable this feature.

## Enable Logic

Limit switches can be combined with Enable Logic to govern limit switch operation based on the occurrence of specific conditions. For example, the AND Enable feature can be used to prevent a limit switch output from turning ON unless the AND Enable input is ON.

The following Enable Logic can be combined with limit switches:

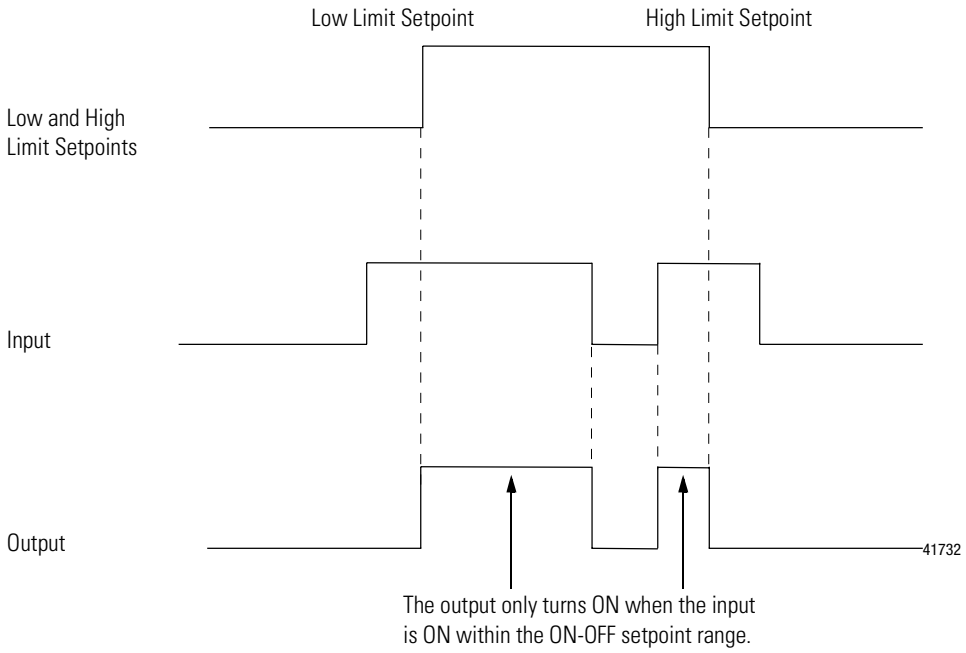
- AND Enable
- Pulse Enable
- Windowed Enable
- Timer Enable
- RPM Enable

These features can be disabled so as not to affect limit switch operation.

### *AND Enable*

This feature enables an output to turn ON and OFF if the resolver is within the Low Limit-High Limit setpoints and the designated AND input point is ON.

The graphic below shows an output, with configured Limit setpoints, turn ON and OFF based on the operation of an AND input.




You must perform the following steps to use the AND Enable:

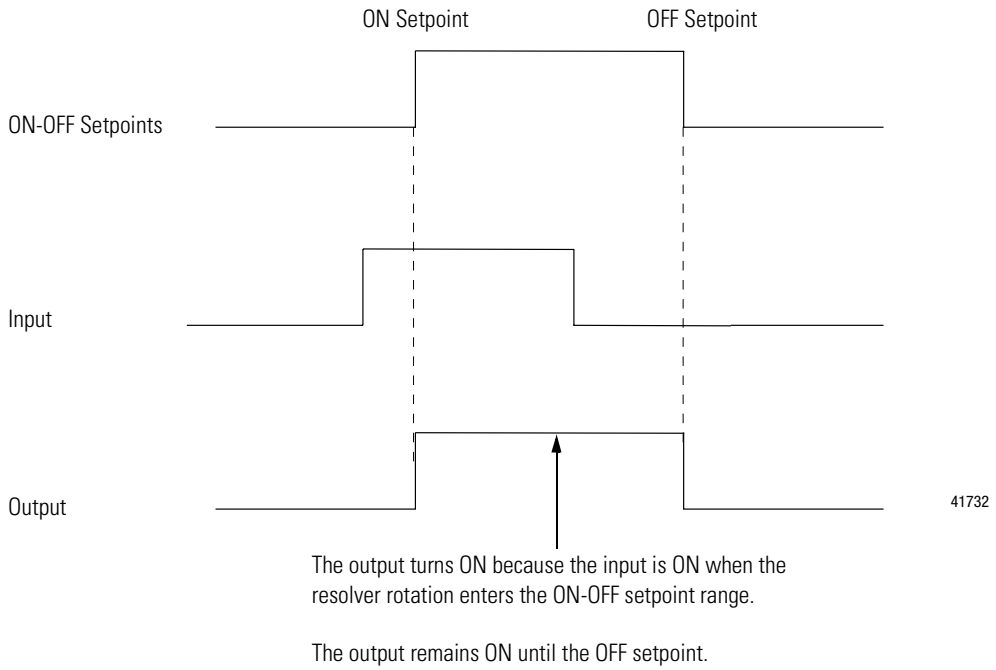
1. Set the **Low Limit** value
2. Set the **High Limit** value
3. Enable the AND Enable feature (i.e. set the **AND Enable** bit to 1).

**IMPORTANT** When using the And Enable, remember that the output only remains ON within the setpoints if the input is ON. If the input turns OFF, the output also turns OFF.

### *Pulse Enable*

The Pulse Enable turns an output ON if an input is already ON when the resolver enters the setpoint.

The graphic below shows an output, with configured Low and High Limit setpoints, turn ON because an input is already ON when the resolver rotation reaches the ON setpoint.




You must perform the following steps to use the Pulse Enable:

1. Set the **ON Setpoint** value
2. Set the **OFF Setpoint** value
3. Enable the Pulse Enable feature (i.e. set the **Pulse Enable** bit to 1).

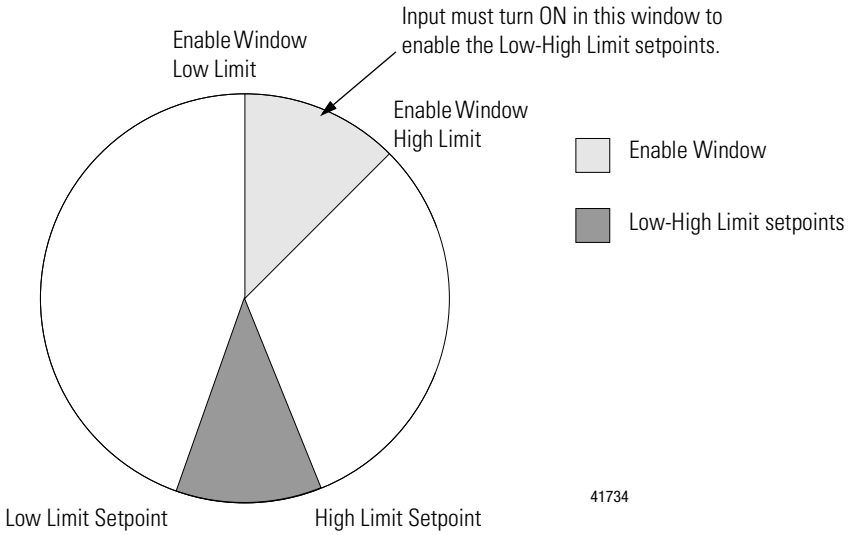
**IMPORTANT** When using the Pulse Enable, remember that the output remains ON within the setpoints even if the input turns OFF after the output has been energized.

### *Windowed Enable*

Windowed Enable allows you to turn an output ON within its high and low setpoints if an input turns on within a separately configured window. Typically, the output setpoints follow the input window in the resolver's rotation.

For example, an input may turn ON when a part reaches a specific point in an assembly line and, because the application is using the windowed enable feature, an output will turn ON at a later point in the same line causing rivets to be drilled into the part.

The graphic below shows a resolver rotation with an enable window and ON-OFF setpoints.




You must perform the following steps to use the Pulse Enable:

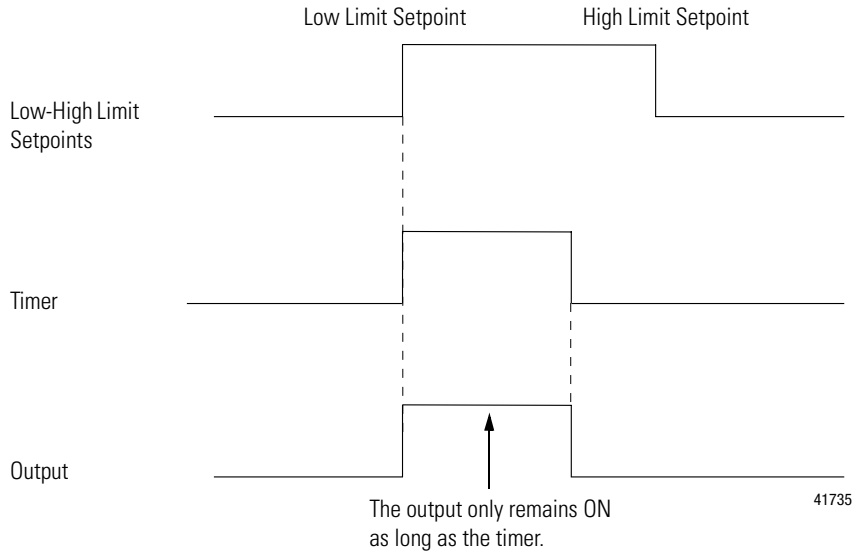
1. Set the **Windowed Enable High Limit** value
2. Set the **Windowed Enable Low Limit** value
3. Set the **Windowed Enable Input** value
4. Enable the **Windowed Enable** feature (i.e. set the **Windowed Enable** bit to 1).

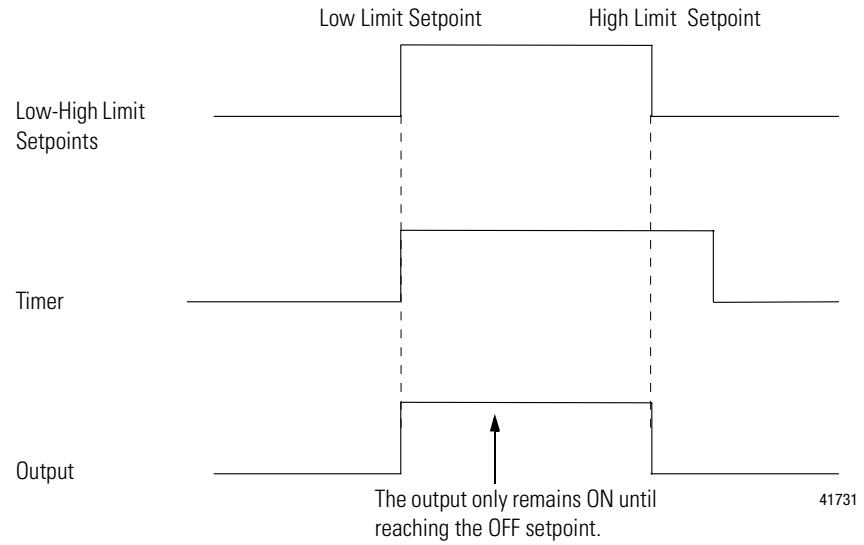
### *Timer Enable*

The Timer Enable limits the time an output can be ON. You set the **On Time Maximum (ms)** field to set how long the output can be ON.

**IMPORTANT** If you set the On Time Maximum to 0, the output will always be ON while in the Low and High Limits (i.e. a value of 0 means you are not using the Timer Enable).

After the time has expired, the output turns OFF even if the resolver rotation is still within the setpoints. Also, the output turns OFF at the OFF setpoint, regardless of whether the timer has expired, as shown in the second graphic below.



You must perform the following steps to use the Timer Enable:

1. Set the **Low Setpoint** value
2. Set the **High Setpoint** value
3. Enable the **Timer Enable** feature (i.e. set the **Timer Enable** bit to 1).

**IMPORTANT** The Timer Enable turns the output ON at the beginning of the setpoints.

### *RPM Enable*

RPM Enable causes the PLS module to turn outputs ON at speeds above the RPM Enable Low Limit and below the RPM Enable High Limit values.

Typically, you configure the RPM low-high limits to prevent the outputs from turning ON during start-up. The following table lists common RPM Low Limit and High Limit values used in PLS module applications.

**Table 3.B**
**Common RPM Minimum and Maximum Values**

| RPM Enable Low Limit: | RPM Enable High Limit | Resulting Operation: |
|---|---|---|
| 0 | 1,000 | Output enabled from 0 to 1,000 RPM |
| -1,000 | 1,000 | Output enabled from -1,000 to 1,000 RPM |
| -32,768 | 32,767 | Output is enabled at any RPM |
| 1,000 | 0 | Output is disabled from 0 to 1,000 RPM |

The RPM Enable can also be used for applications using a resolver that rotates backwards. Use negative values for the RPM Enable Low Limit and High Limit to establish an RPM range.

You must perform the following steps to use the Timer Enable:

1. Set the **RPM Low Limit** value
2. Set the **RPM High Limit** value
3. Enable the **RPM Enable** feature (i.e. set the **RPM Enable** bit to 1).

### *RPM Low Limit*

RPM Low Limit determines the minimum resolver RPM needed to enable an output. For example, if you set the RPM minimum to 10, any resolver speed under 10 RPM will disable the output.

### *RPM High Limit*

RPM Maximum determines the maximum resolver RPM which enables an output. For example, if you set the RPM maximum to 1000, any resolver speed above 1,000 RPM will disable the output.

### *Input Invert*

The Input Invert function, used with Enable Logic, causes the PLS module to invert the action of an input channel.

For example, you can configure an input channel on the PLS module to AND Enable a limit switch when the sensor connected to the AND Enable input is OFF.

**IMPORTANT** Set the **Input Invert** bit to 1 to enable this feature.

### *Output Override*

The Output Override feature allows you to directly control the state of PLS module outputs. For example, an output may be configured to turn ON at 45° in the resolver rotation and stay ON until 270°. Using Output Override, you can turn the output OFF or ON at any time. This features allows you to use surplus outputs as normal digital outputs or for wire testing.

**IMPORTANT** The Output Override feature is secondary to PLS Enable Outputs. You can only use the Output Override if the PLS module outputs are enabled and message communication is active.

You must perform the following steps to use Output Override:

1. Set the **Output Override** value.
2. Enable the Output Override feature (i.e. set the **Output Override** bit to 1).

### *Input Override*

The Input Override feature allows you to directly control the state of PLS module inputs. You can use this feature to test PLS module behavior or to override malfunctioning sensors.

You must perform the following steps to use Output Override:

1. Set the **Input Override** value.
2. Enable the Input Override feature (i.e. set the **Input Override** bit to 1).

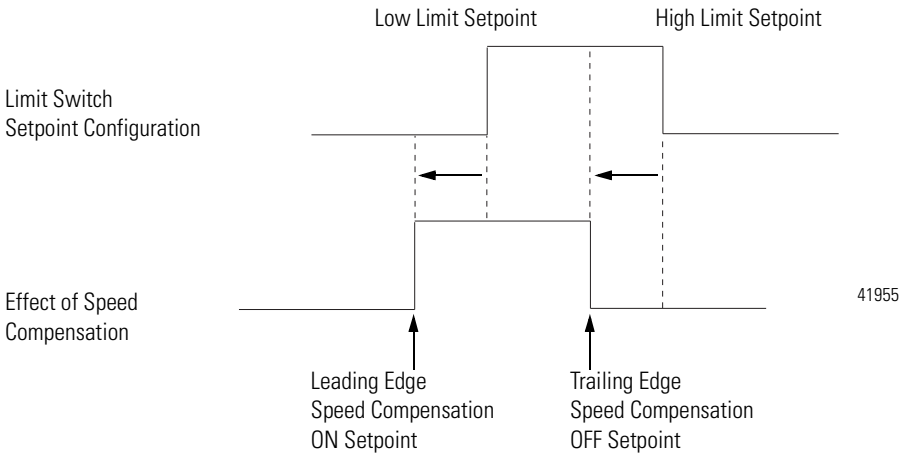
## Speed Compensation

The PLS module can compensate ON and OFF positions for the speed at which a resolver is turning. Speed Compensation compensates for device turn ON/OFF latency. This feature allows the module to turn outputs ON and OFF at correct times so that the equipment connected to the PLS accurately works with the resolver.

The following factors affect how the module compensates for the speed of the resolver:

- Lead Speed - Distance and RPM - affects ON position
- Trail Speed - Distance and RPM - affects OFF position

The graphic below shows limit switch Low-High Limit setpoints and, below it, Low-High Limit setpoints after they have been adjusted by Speed Compensation on the PLS module. Notice that the outputs actually turn ON and OFF before the actual setpoints are reached because of the device latency.



### *Lead Speed Distance and RPM*

**Lead Speed Distance** sets the distance (in engineering units) to advance the beginning of an output setpoint window for speed compensation.

This parameter must be configured in conjunction with the Lead Speed RPM; **Lead Speed RPM** sets the speed at which a resolver must be turning to compensate by Lead Speed Distance. Lead Speed Distance is applied proportionally to the actual RPM and Lead Speed RPM.

For example, you may configure the PLS module with a Lead Speed Distance of 10 and a Lead Speed RPM of 5. For every 5 RPM, the setpoint window will advance 10. An RPM of 20 will cause the module to advance the beginning of the setpoint window 40. An RPM of 13 will cause the module to advance the beginning of the setpoint window 26.

**IMPORTANT** If you use a distance value of 0, the module will not use speed compensation.

### *Trail Speed Distance and RPM*

**Trail Speed Distance** sets the distance (in engineering units) to advance the trailing (ON to OFF edge) of an output setpoint window for speed compensation.

This parameter must be configured in conjunction with the Trail Speed RPM; **Trail Speed RPM** sets the speed at which a resolver must be turning to compensate by the Trail Speed Distance. Trail Speed Distance is applied proportionally to the actual RPM and Trail Speed RPM.

For example, you may configure the PLS module with a Trail Speed Distance of 10 and Trail Speed RPM of 5. For every 5 RPM, the setpoint window will be advance a value of 10. An RPM of 20 will cause the module to advance the ending of the setpoint window 40. An RPM of 13 will cause the module to advance the ending of the setpoint window 26.

### *Configuring Speed Compensation for a Specific Lead Time*

If the PLS module controls a device with a known lag time, it may be more convenient to configure the speed compensation using time units. Use the following formula to convert the lag time to equivalent distance and RPM:

$$\text{mS of Advance} = \frac{\text{LeadSpeedDistance}}{\text{LeadSpeedRPM}} \text{ X } \frac{60000}{\text{Rollover Position} + 1}$$

Using this equation, if:

- Rollover Counts = 4095
- Rollover Position = 4095
- LeadSpeedDistance = 1
- LeadSpeedRPM =1

These parameters cause the Limit Switch to turn ON 14.6mS before reaching the programmed setpoint angle.

The following equation determines speed compensation for **applications using degrees as the engineering units**:

$$\text{mS of Advance} = \frac{\text{shLeadSpeedDistance}}{\text{wLeadSpeedRPM}} \text{ X } \frac{60000}{359 + 1}$$

Using this equation, if:

- Rollover Counts = 4095
- Rollover Position = 359
- LeadSpeedDistance = 1
- LeadSpeedRPM =1

The limit switch turns ON 166ms before reaching the programmed setpoint angle.

### *Determining Lead Speed Distance and RPM for a Desired Lead Time*

You may want your PLS application to use a specific lead time. Use the following worksheet to calculate the appropriate Lead Speed Distance and RPM.

The worksheet below uses a Lead Speed RPM of 1000.

1. Desired Lead Time: _____ (in ms)
2. Rollover Position + 1:_____
3. Lead Speed RPM: 1000
4. Use the following equation to calculate Lead Speed Distance:

$$\frac{\text{Lead Time X Rollover Position}}{60} = \text{\_\_\_\_\_} \leftarrow \text{Lead Speed Distance}$$

For example, using the worksheet and equation to calculate a Lead Speed Distance for a desired Lead Time of 30ms and when you are measuring position in 0.1 degree units.

1. Desired Lead Time: 30ms
2. Rollover Position + 1:3599 + 1 = 3600
3. Lead Speed RPM: 1000
4. Use the following equation to calculate Lead Speed Distance:

$$\frac{30 \text{ X } 3600}{60} = 1800 \leftarrow \text{Lead Speed Distance}$$

### *RPM Filter*

The RPM filter adjusts how quickly the RPM reported by the module, and used internally for speed compensation, tracks the actual RPM. Typically, the RPM filter should be configured so that the PLS module RPM closely follows any speed changes.

You can configure the RPM filter so that minor changes in the resolver's RPM are not reported. In this case, the average RPM rate is reported.

For example, the resolver may be connected to a motor that oscillates between 100 and 102 RPM approximately every 0.5 seconds. The RPM filter, if configured to lag behind the actual RPM when reporting speeds, may report a resolver RPM of 101. In this case, continued increases in speed, such as RPM values of 103 and 104, will be reported slightly later than they actually occurred.

Use the table below to choose a value for the RPM filter.

**Table 3.C**
**RPM Filter Values and Effects**

| Filter Value: | Effect: |
|---|---|
| 16#4000 | Reported RPM matches actual RPM closely |
| 16#8000 | Reported RPM matches actual RPM somewhat closely |
| 16#C000 | Reported RPM lags actual RPM somewhat |
| 16#F000 | Reported RPM lags actual RPM significantly |

## Limit Switch-Specific Home Position

In addition to determining the relative zero point (home position) for the resolver's axis, you may also determine a relative zero point for each limit switch. Three features determine a limit switch's home position:

- Zero Offset
- Preset
- Nudge Up/Down

### *Zero Offset*

Use **Zero Offset** when you know the starting point in the resolver's axis. For example, if the limit switch's zero position is at the machine's 10° point, you can configure the PLS module with a limit of 10° to match the limit switch zero postion to the machine zero postion.

The Zero Offset value is limited to +/- the Rollover Position. For example, if you are using degrees as your engineering units, the maximum engineering units is 359°. In this case, the Zero Offset can be configured for values between -359 and +359.

**IMPORTANT** The PLS module does not save Zero Offset values after power-down. The value must be saved in the Logix5550 controller and downloaded to the PLS module with limit switch configuration data.

### *Preset*

Use the Preset to set the relative starting point for the limit switch by the energizing of an input.

For example, the PLS module can be configured to set the position to 0° when an input turns ON. The PLS module replaces the current Zero Offset value with this number.

When the Preset occurs, the PLS changes the current Position to its Preset predefined value. The Zero Offset value is recalculated by the PLS module. The new zero offset value is temporary and remains until the occurrence of the next Preset.

**Preset Input** informs the PLS module which input on the module signals the Preset event.

Perform the following steps to use the Preset:

1. Set the **Preset** value - For example, if you want a position equal to zero when the input turns ON, set **Preset** to zero.
2. Set the **Preset Input** - For example, if you want input 3 to trigger the Preset Event, set this value to 3.
3. Enable the Preset feature (i.e. set the **Arm Preset** bit to 1).

**IMPORTANT** The **Preset** value is limited to less than or equal to the Rollover Position. For example, if you are using degrees as your position units, the value of the **Preset** can range from 0° to 359°.

## Chapter Summary and What's Next

In this chapter you learned about the ControlLogix Programmable Limit Switch module features.

Move to Chapter 4 to learn about installing and wiring the PLS module.

## Notes:

# Installing and Wiring the ControlLogix Programmable Limit Switch Module

## What this Chapter Contains

This chapter describes how to install ControlLogix modules.



## Installing the ControlLogix Programmable Limit Switch Module

You can install or remove the module while chassis power is applied.

**ATTENTION**

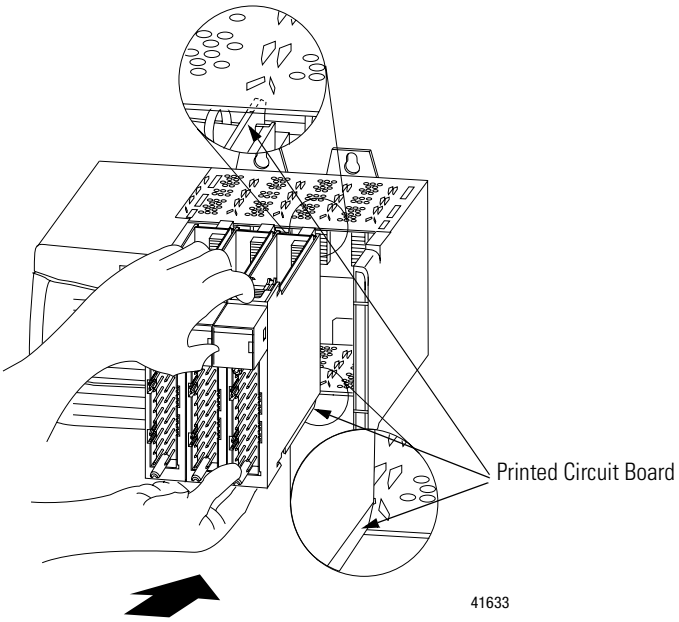
The module is designed to support Removal and Insertion Under Power (RIUP). However, when you remove or insert an RTB with field-side power applied, **unintended machine motion or loss of process control can occur**. Exercise extreme caution when using this feature.

**IMPORTANT**

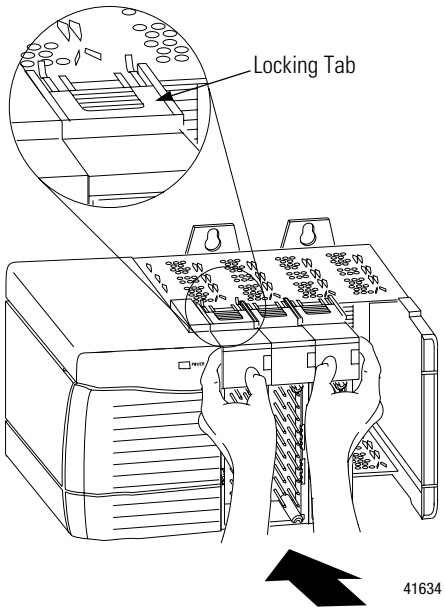
Pulling the PLS module from the chassis while power is applied causes the message instructions carrying configuration, input and output data to timeout.

If you remove the PLS module from the chassis while under power, in addition to considering the warnings mentioned previously, you must also remember to reconfigure the module upon reinsertion.

**1.** Align center circuit board with top and bottom chassis guides.




**2.** Slide the module into the chassis until all six module tabs 'click'.



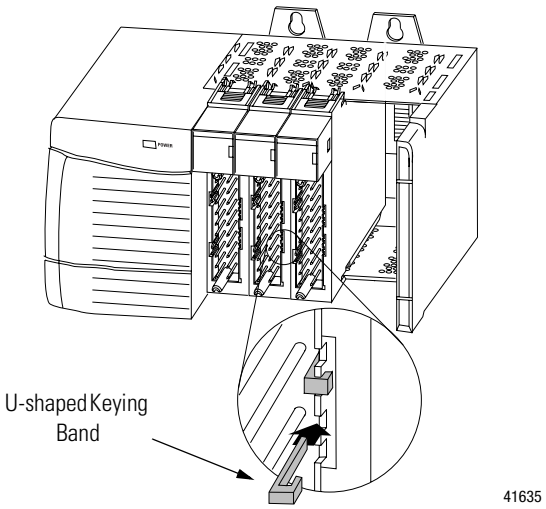
## Keying the Removable Terminal Block

Key the RTB to prevent inadvertently connecting the incorrect RTB to your module.

When the RTB mounts onto the module, keying positions will match up. For example, if you place a U-shaped keying band in position #4 on the module, you cannot place a wedge-shaped tab in #4 on the RTB or your RTB will not mount on the module.

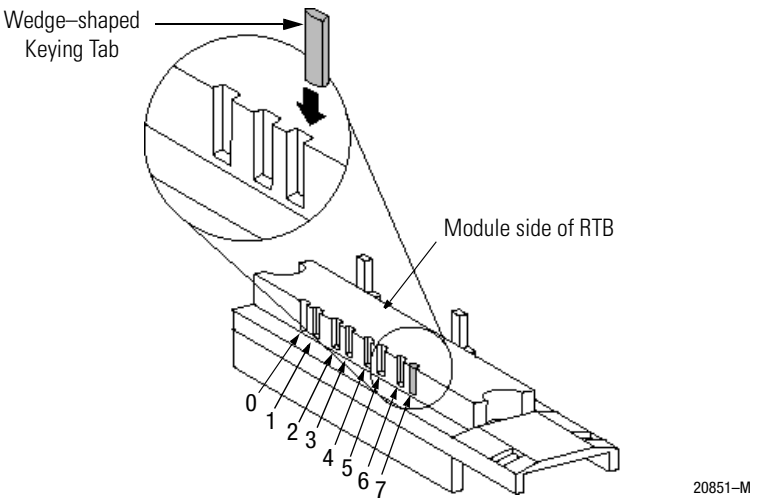
We recommend that you use a unique keying pattern for each slot in the chassis.

1. Insert the U-shaped band with the longer side near the terminals. Push the band onto the module until it snaps into place.




2. Key the RTB in positions that correspond to unkeyed module positions. Insert the wedge-shaped tab on the RTB with the rounded edge first. Push the tab onto the RTB until it stops.

When keying your RTB and module, you must begin with a wedge-shaped tab in position #6 or #7.



## Connecting Wiring

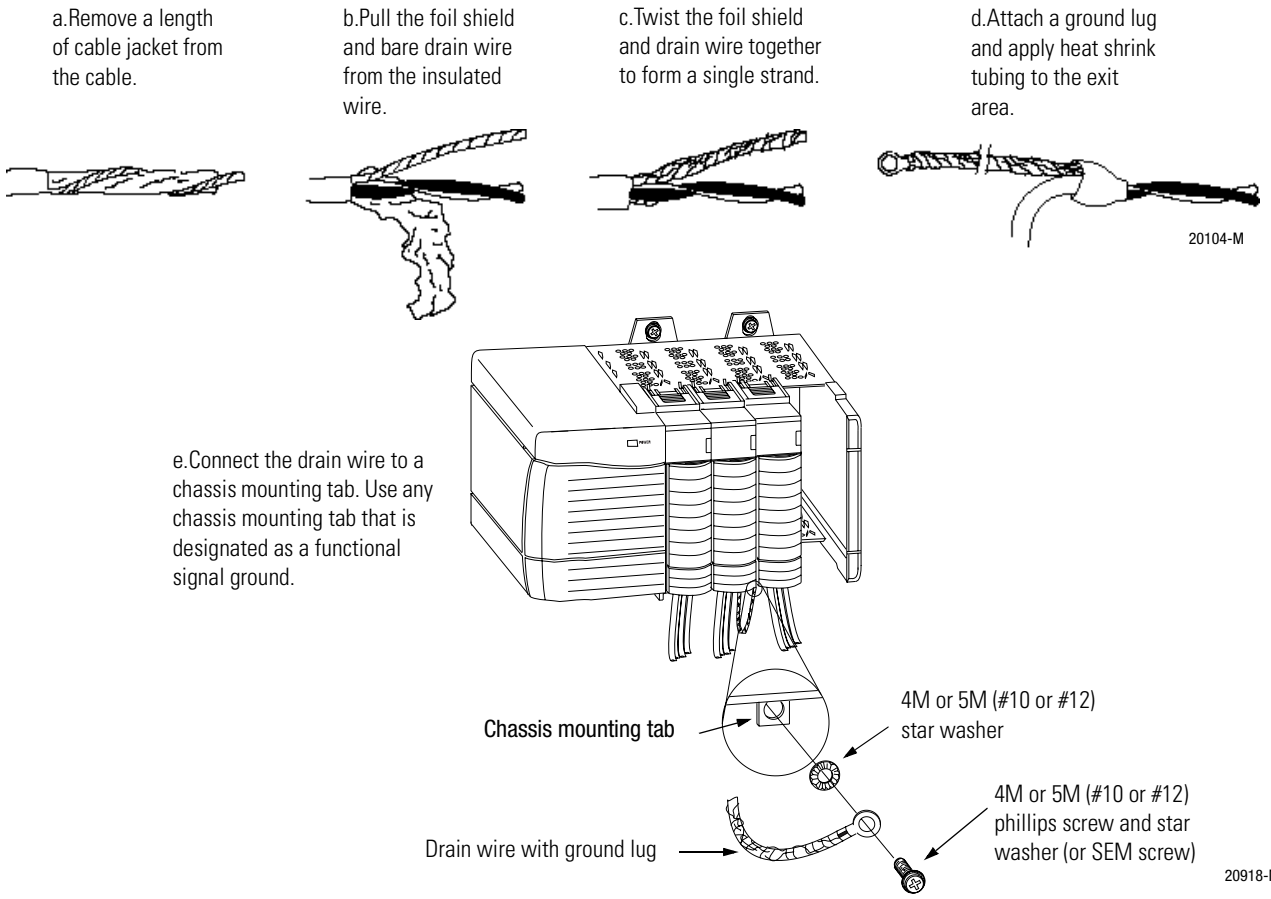
You must use an RTB to connect wiring to your module. The RTB terminations can accommodate 14-22 gauge shielded wire. Before wiring the RTB, you must connect ground wiring to the resolver.

**IMPORTANT** The I/O sections of the PLS module are grounded when wiring is connected and the module is inserted into the ControlLogix chassis. You must ground the resolver separately though.

### Connect Grounded End of the Cable

1. Ground the drain wire at the resolver-side.

We recommend you ground the drain wire at the resolver-side. If you cannot ground at the resolver-side, ground at an earth ground on the chassis as shown below.




2. Connect the insulated wires to the resolver-side.

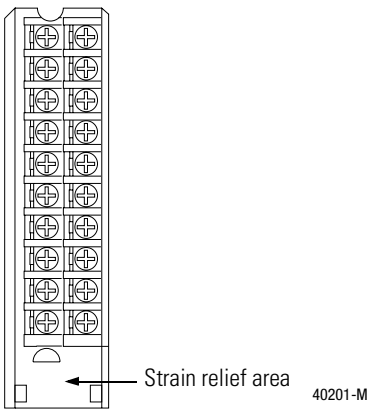
## Connect Ungrounded End of the Cable

1. Cut the foil shield and drain wire back to the cable casing and apply shrink wrap.
2. Connect the insulated wires to the RTB, as shown on page 4-5.

*Two Types of RTBs (each RTB comes with housing)*

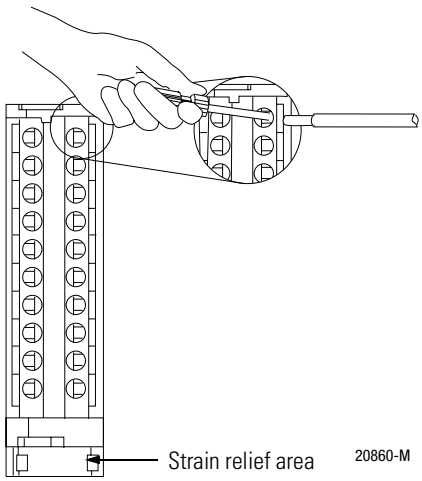
- **NEMA clamp** - Catalog number 1756-TBNH

Terminate wires at the screw terminals.




- **Spring clamp** - Catalog number 1756-TBSH

1. Insert the screwdriver into the outer hole of the RTB.
2. Insert the wire into the open terminal and remove the screwdriver.




*Recommendations for Wiring Your RTB*

We recommend you follow these guidelines when wiring your RTB:

1. Begin wiring the RTB at the bottom terminals and move up.
2. Use a tie to secure the wires in the strain relief area of the RTB.
3. Order and use an extended-depth housing (Cat. No.1756-TBE) for applications that require heavy gauge wiring.

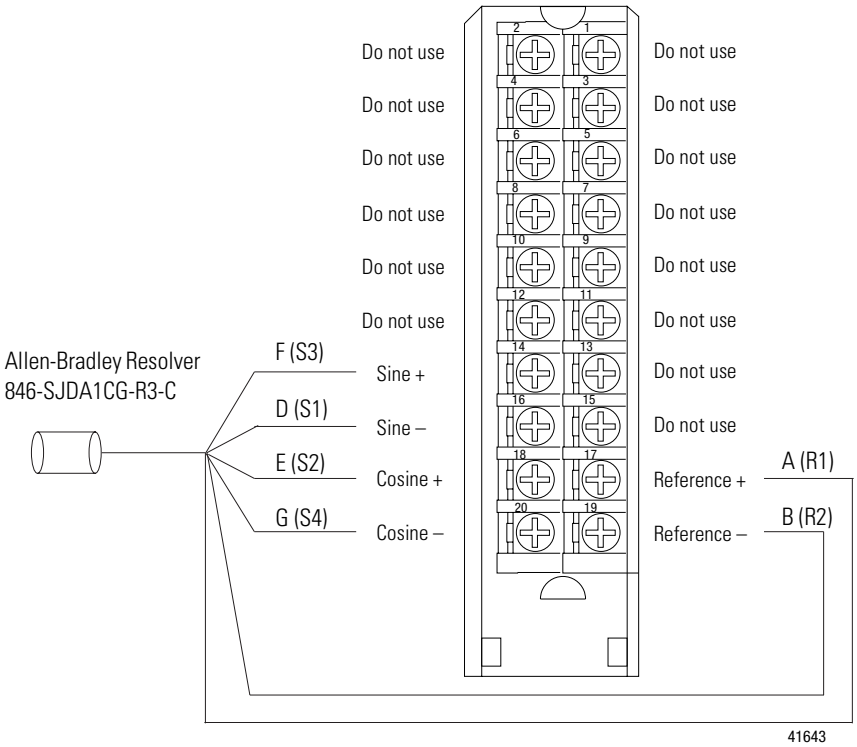
## Wiring the Resolver Module

Use the table below to connect the resolver and resolver connector to the PLS resolver module.

**IMPORTANT** This wiring yields increasing position with counterclockwise resolver shaft rotation (viewed from shaft). Switch the wiring to RTB terminals 14 and 16 to reverse direction.

**Table 4.A**
**Connections from Allen-Bradley Resolvers to the PLS Resolver Module**

| RTB Terminal Number | 846-SJxNxCG-R3 846-SJxAxCG-R3 Allen-Bradley Resolvers Designation | 845-CA-E-25 845-CA-F-25 Connector Cables Color Pairs | Resolver Signal Name |
|---|---|---|---|
| 17 (Reference+) | A | White | R1 |
| 19 (Reference-) | B | Black of white | R2 |
| 14 (Sine+) | F | Black of red | S3 |
| 16 (Sine-) | D | Red | S1 |
| 18 (Cosine+) | E | Green | S2 |
| 20 (Cosine-) | G | Black of green | S4 |

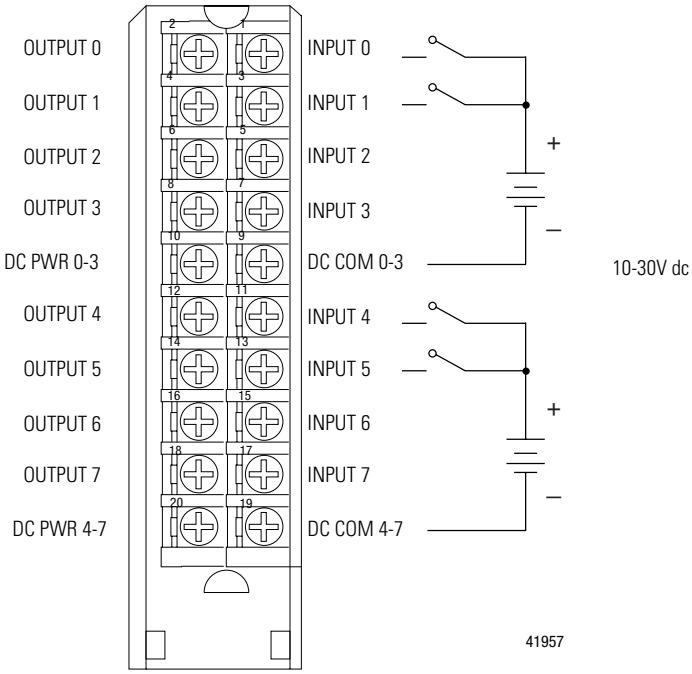


# Wiring the I/O Modules

## Wiring Inputs

The example diagram below shows switches connected to the inputs on the left section I/O module.

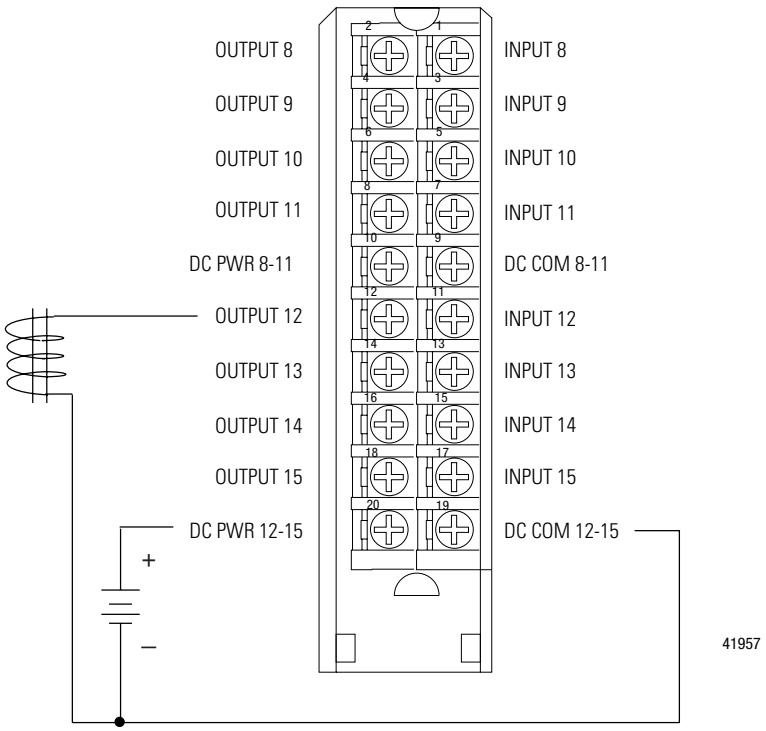
**Left Section I/O Module**



## Wiring Outputs

The example diagram below shows a solenoid connected to an output on the right section I/O module.

**Right Section I/O Module**

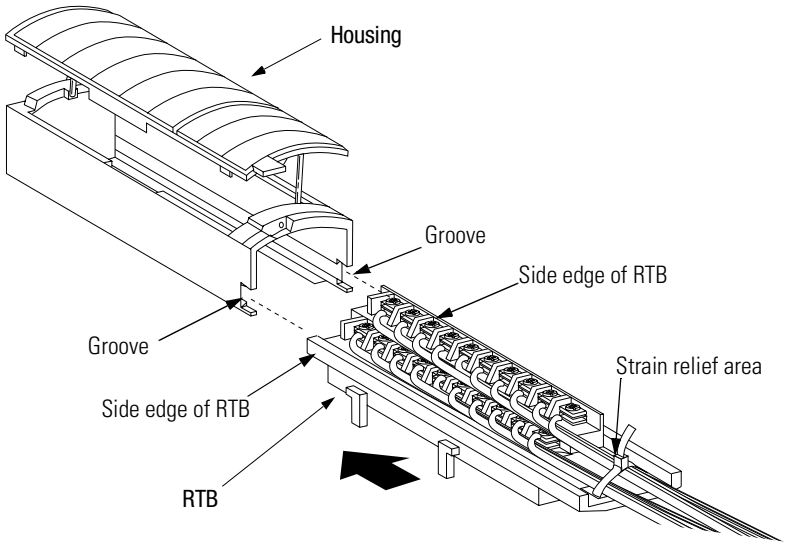


## Assembling the Removable Terminal Block and the Housing

Removable housing covers the wired RTB to protect wiring connections when the RTB is seated on the module.

1. Align the grooves at the bottom of each side of the housing with the side edges of the RTB.

2. Slide the RTB into the housing until it snaps into place.




**IMPORTANT** If additional wire routing space is required for your application, use extended-depth housing 1756-TBE.

## Installing the Removable Terminal Block onto the Module

Install the RTB onto the module to connect wiring.

**ATTENTION**

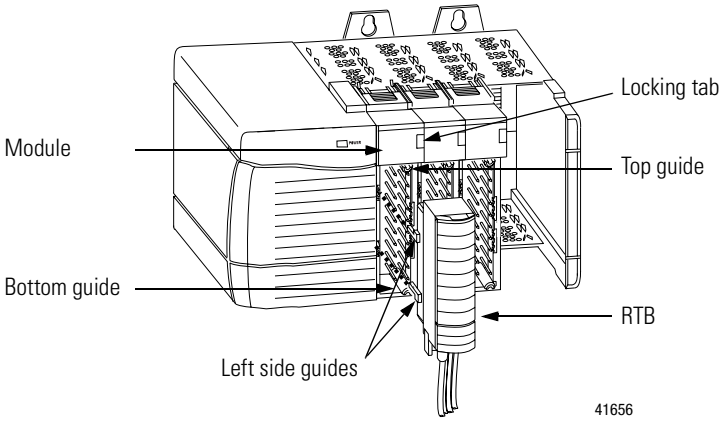
**Shock hazard exists.** If the RTB is installed onto the module while the field-side power is applied, the RTB will be electrically live. Do not touch the RTB's terminals. Failure to observe this caution may cause personal injury.

The RTB is designed to support Removal and Insertion Under Power (RIUP). However, when you remove or insert an RTB with field-side power applied, **unintended machine motion or loss of process control can occur**. Exercise extreme caution when using this feature. It is recommended that field-side power be removed before installing the RTB onto the module.

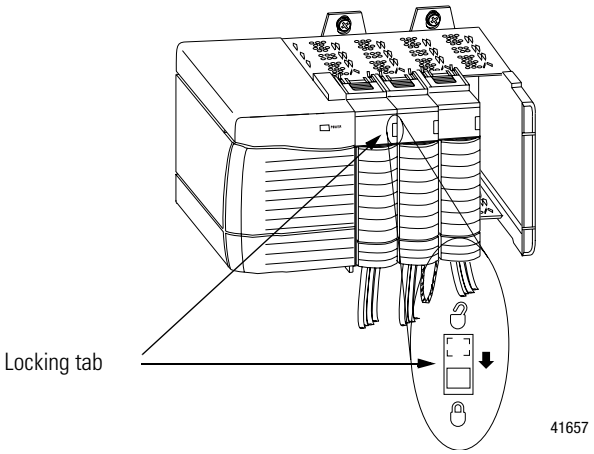
Before installing the RTB, make certain:

- field-side wiring of the RTB has been completed.
- the RTB housing is snapped into place on the RTB.
- the RTB housing door is closed.
- the locking tab on the front of the module is unlocked.

1. Align the top, bottom and left side guides of the RTB with matching guides on the module.




2. Press quickly and evenly to seat the RTB on the module until the latches snap into place.




3. Slide the locking tab down to lock the RTB onto the module.

## Removing the Removable Terminal Block from the Module

If you need to remove the module from the chassis, you must first remove the RTB from the module.

**ATTENTION**

**Shock hazard exists.** If the RTB is removed from the module while the field-side power is applied, the module will be electrically live. Do not touch the RTB's terminals. Failure to observe this caution may cause personal injury.
The RTB is designed to support Removal and Insertion Under Power (RIUP). However, when you remove or insert an RTB with field-side power applied, **unintended machine motion or loss of process control can occur**. Exercise extreme caution when using this feature. It is recommended that field-side power be removed before removing the module.

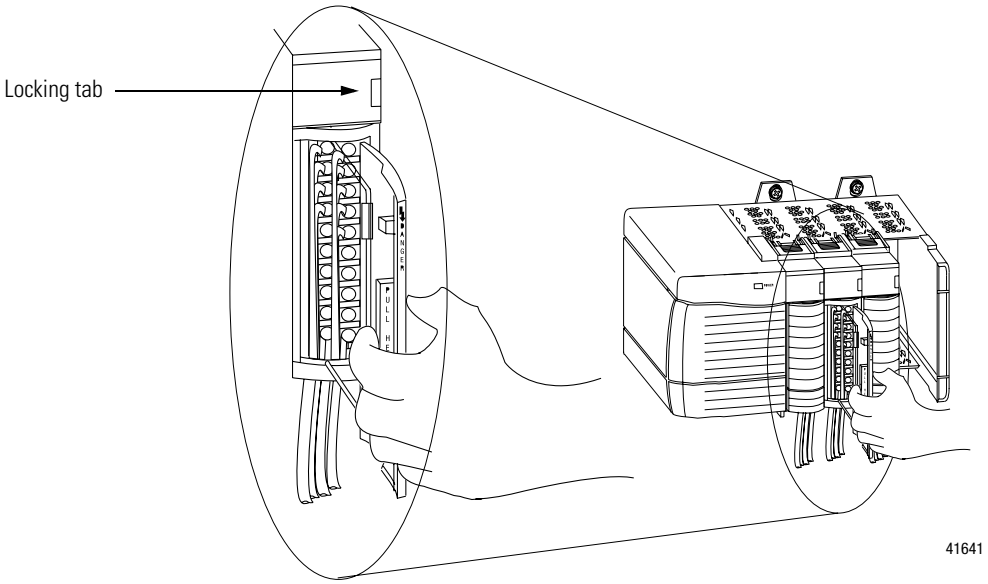
**IMPORTANT**

Pulling the PLS module from the chassis while power is applied causes the message instructions carrying configuration, input and output data to timeout.

If you remove the PLS module from the chassis while under power, in addition to considering the warnings above, you must also remember to reconfigure the module upon reinsertion.

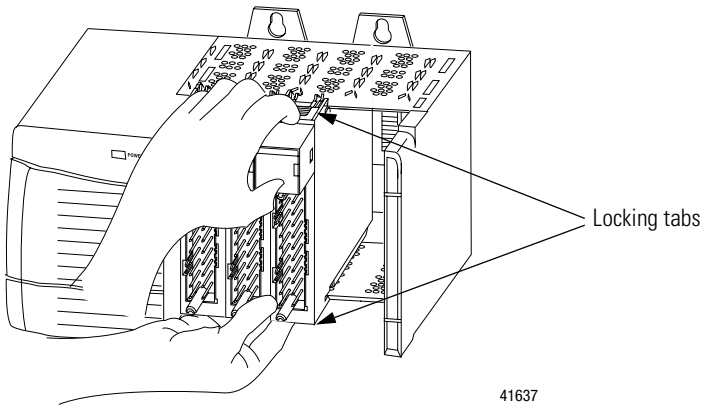
1. Unlock the locking tab at the top of the module.
2. Open the RTB door using the bottom tab.
3. Hold the spot marked PULL HERE and pull the RTB off the module.

Do not wrap your fingers around the entire door. A shock hazard exists.

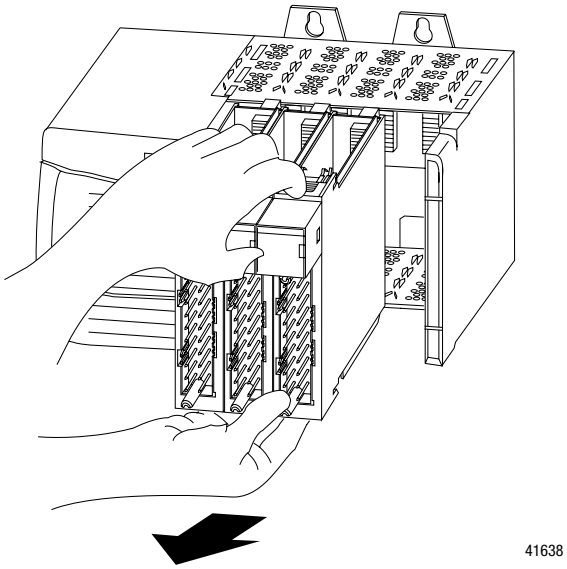


## Removing the Module from the Chassis

1. Push in all the top and bottom locking tabs.




2. Pull module out of the chassis.



## Chapter Summary and What's Next

In this chapter you learned about:

- installing the module
- keying the removable terminal block
- connecting wiring
- assembling the removable terminal block and the housing and installing them onto the module
- removing the removable terminal block from the module
- removing the module from the chassis

Move on to chapter 5 to learn how to configure your module.

## Notes:

# Configuring the ControlLogix Programmable Limit Switch Module

## What This Chapter Contains

This chapter describes how to configure ControlLogix Programmable Limit Switch modules.



## Configuring Your Programmable Limit Switch Module

You must configure your module upon installation. The module will not work until it has been configured.

**IMPORTANT**

The 1756-PLS module uses the following data types for module configuration:

- Module-defined data types
- User-defined data types

Full configuration of the PLS module requires use of all data types.

User-defined types are explained in this chapter, in the RSLogix 5000 online help and in the application itself.

## RSLogix 5000 Programming Software

You must use RSLogix 5000 to configure the PLS module. Configuration changes to the module-defined data types are made in the Data Monitor of RSLogix 5000 and downloaded to the module.

- Changes made before module operation begins are sent to the PLS module during the initial configuration.
- Changes made after module operation has begun must be sent to the PLS module by a message instruction.

### *Module-Defined Data Types*

When you create a PLS module in RSLogix 5000 (i.e. add the module to the Controller Organizer, see page 5-5), module-defined data types and tags are created. There are three module-defined data types:

- Input data type - type reports position and state of digital inputs and outputs on the module
- Output data type - type allows controller to manually override input and output values
- Configuration data type - type configures scaling, zero offset and other module-wide features

### *User-Defined Data Types*

Before you can use your PLS module, you must also generate the following user-defined data types:

- Limit switch data type - type allows control of each module output and, consequently, any limit switch connected to that output
- Offset data type - type reports the active zero offset and preset values for each axis and limit switch
- Registration data type - type reports position at which each module input last went ON or OFF

You cannot complete PLS module configuration without user-defined data types.

**IMPORTANT**

The sample project (Messaging_Configuration_1756_PLS.acd) provided with RSLogix 5000, Version 2.50 offers example user-defined data types and a sample ladder logic application.

It is recommended that you copy the user-defined data types from the sample project to your specific application to configure your PLS module.

If you choose to write user-defined data types instead of using the example types provided, you must write your user-defined data types in the same order as those in the sample file.

### *Using Tags for PLS Configuration*

When you use user-defined data types to write configuration for a module, you create uniquely-named tags in the Tag Editor of RSLogix 5000.

Each configurable feature on a PLS module must have a distinct tag in the controller's ladder logic.

- Tags for module-defined data types are created at module creation.
- Tags for user-defined data types are created when you download the example types from the sample application file (plscfg.ACD).

If you write separate user-defined data types, you must also make sure you create uniquely-named tags for each type.

For more information on creating uniquely-named tags for user-defined data types, see page 5-10.

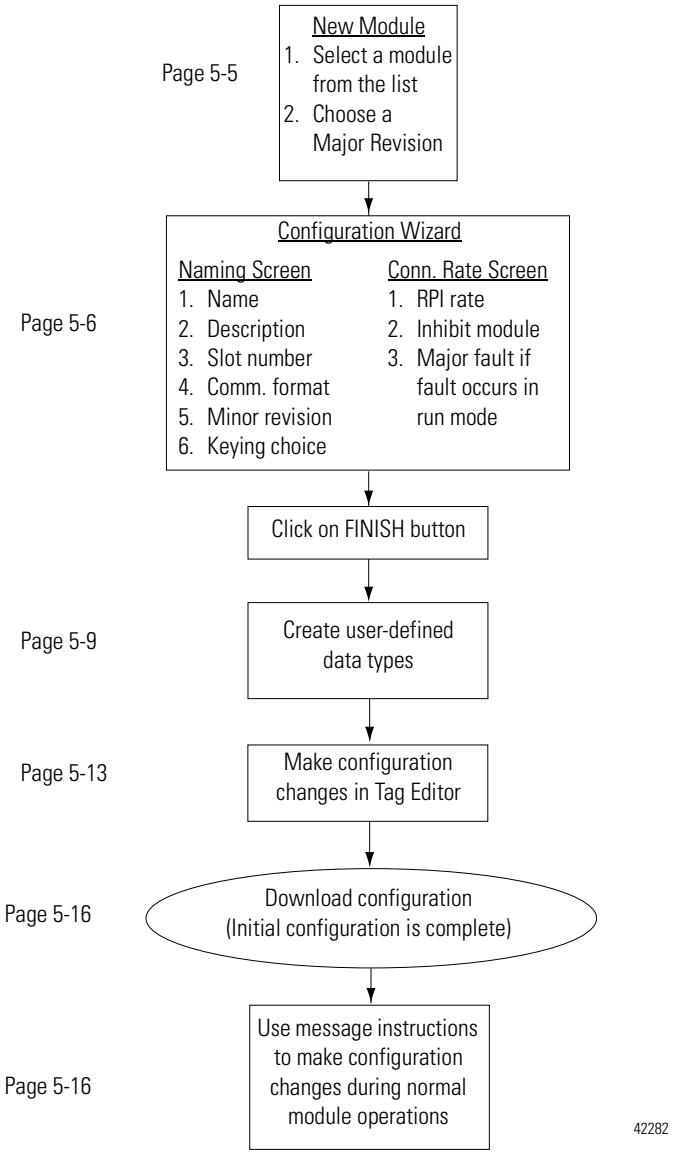
**IMPORTANT**

If your application requires more than one PLS module, you must create unique tags for each PLS module.

For more information on how to create the tags needed for a PLS module, see page 5-10.

## Overview of the Configuration Process

The following diagram shows an overview of the configuration process.

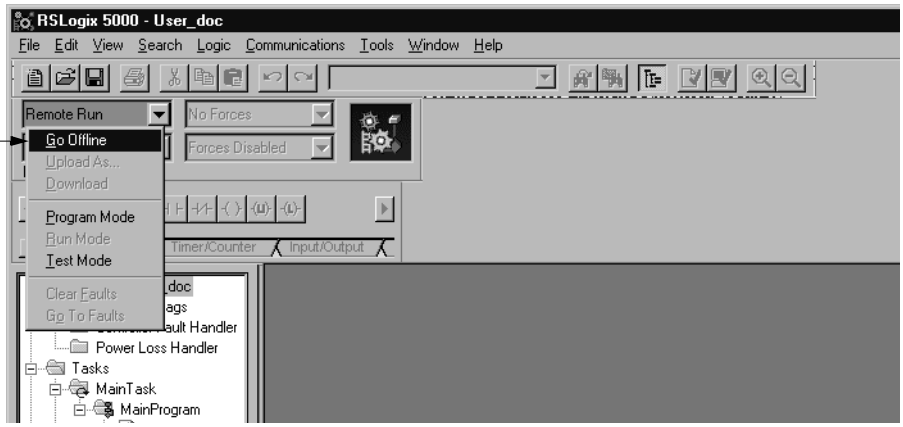


## Creating a New Module

After you have started RSLogix 5000 and created a controller, you must create a new PLS module. The wizard allows you to create a new module and set comunications options.

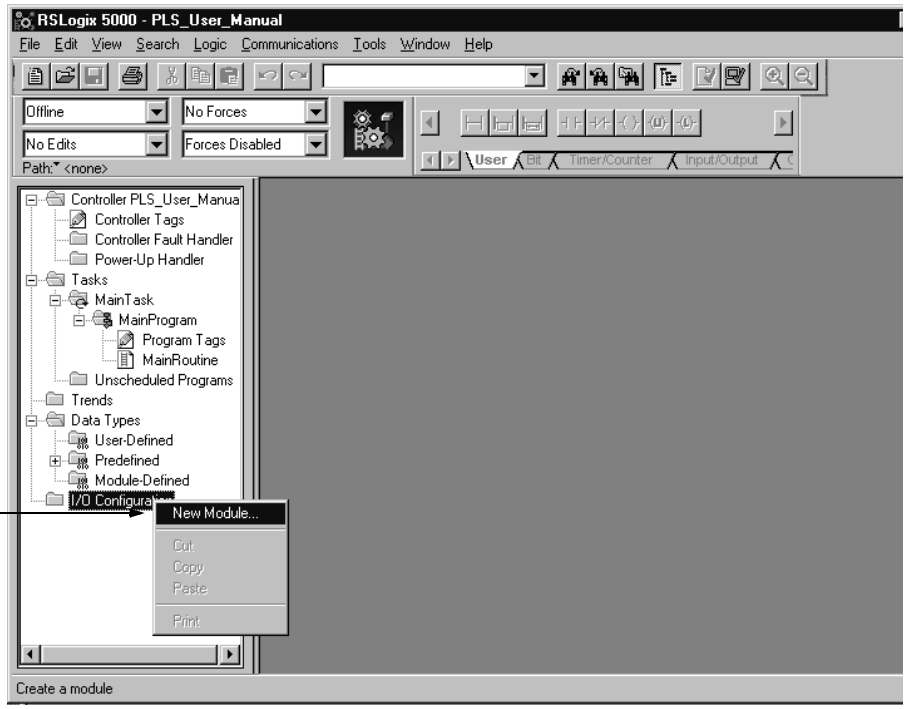
**IMPORTANT** You must be offline when you create a new module.

If you are not offline, use this pull-down menu to go offline

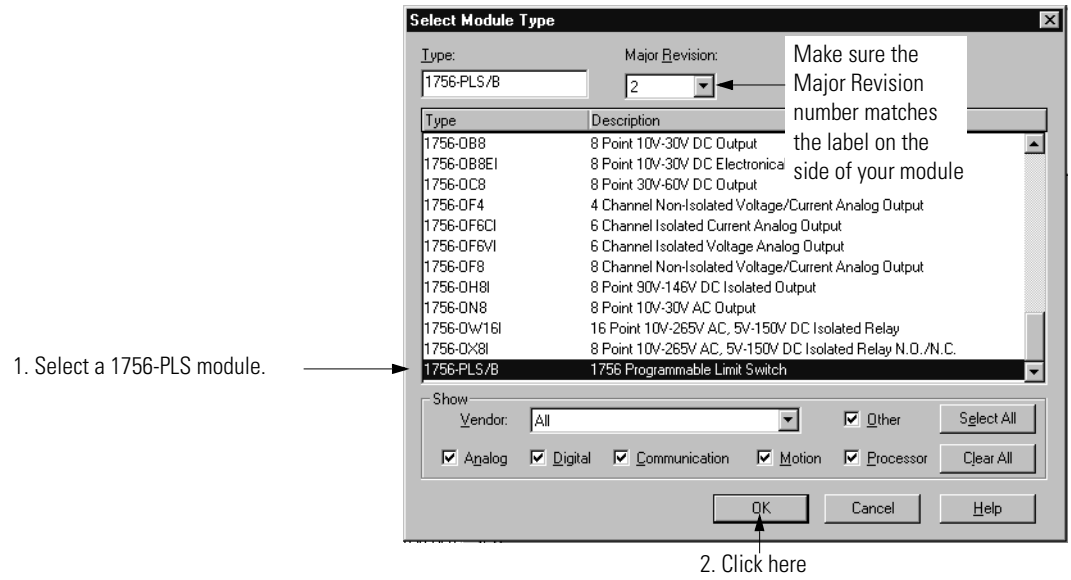



When you are offline, you must select a PLS module.

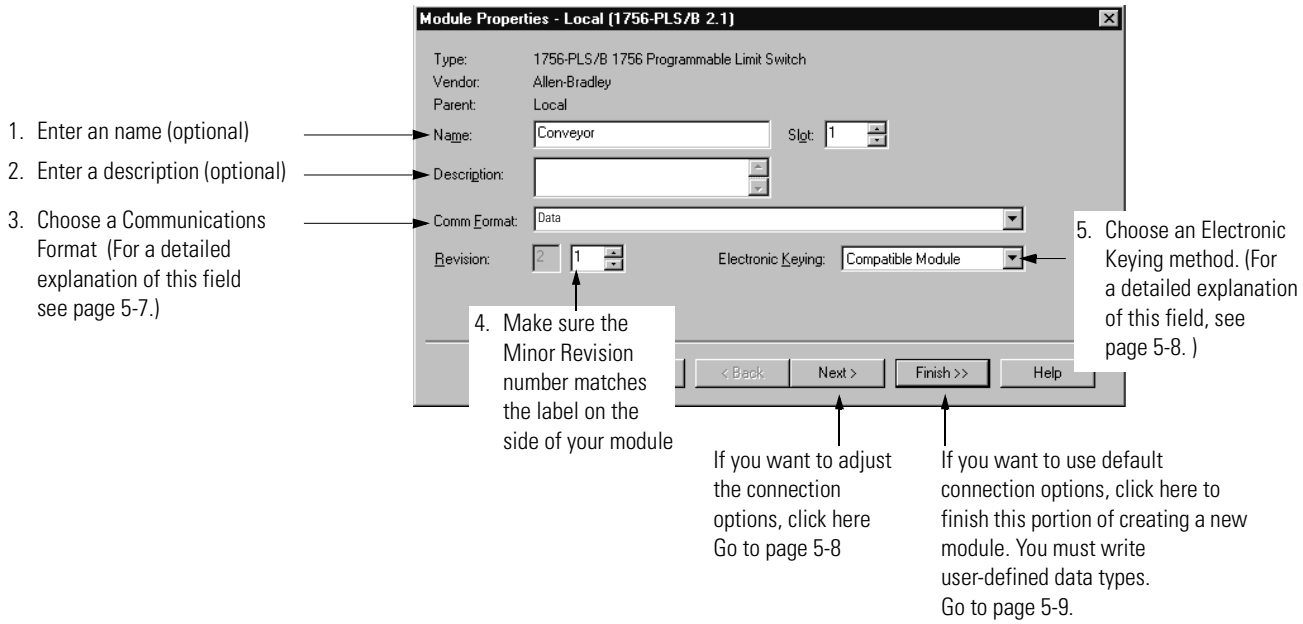
1. Select I/O Configuration.
2. Click on the right mouse button to display the menu.
3. Select New Module



A screen appears with a list of possible new modules for your application.




You enter the new module creation wizard.



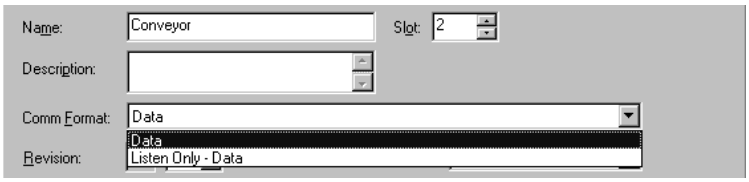
## Communications Format

The communications format determines what type of configuration options are made available, what type of data is transferred between the module and its owner controller. This feature also defines the connection between the controller writing the configuration and the module itself.

The following communications formats are available for your PLS module:

- **Data** - format used by a controller that wants to own the PLS module and control its configuration data
- **Listen-only Data** - format used by a controller that wants to listen to the PLS module but not own it

The screen below shows the choices available.

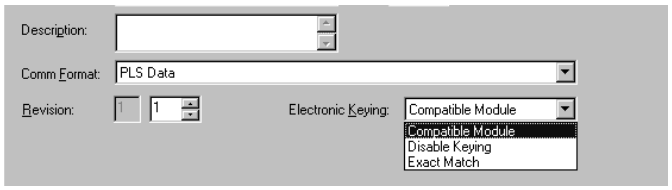



**IMPORTANT** Once the module is created, the communications format cannot be changed. The module must be deleted and recreated.

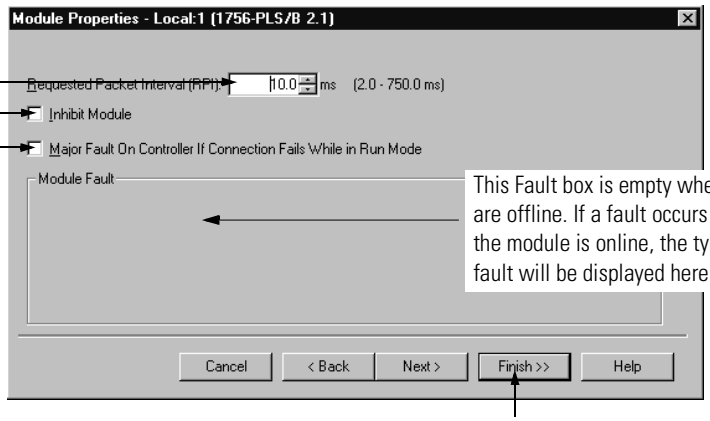
## Electronic Keying

When you create a new PLS module, you can choose how specific the keying must be when a module is inserted into the PLS module's slot in the chassis.

The screen below shows the choices available.




After the naming page, this screen appears.




Adjust the Requested Packet Interval here

Inhibit the connection to the module here

If you want a Major Fault on the Controller to occur if there is connection failure with the I/O module, click here

This Fault box is empty when you are offline. If a fault occurs while the module is online, the type of fault will be displayed here

Click here to finish this portion of creating a new module. You must write user-defined data types. Go to page 5-9.

Two other screens appear in the software but are only used during online operations and are not shown here.

## Creating User-Defined Data Types

Before you can use your PLS module, you must also generate the following user-defined data types:

- Limit switch data type
- Offset data type
- Registration data type

You cannot complete PLS module configuration without user-defined data types.

**IMPORTANT**

The sample project (Messaging_Configuration_1756_PLS.acd) provided with RSLogix 5000, Version 2.5 offers example user-defined data types and a sample ladder logic application.

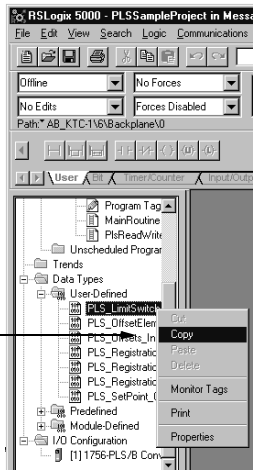
It is recommended you copy the user-defined data types from the sample project to your project to configure your PLS module.

If you choose to write user-defined data types instead of using the example types provided, you must write your user-defined data types in the same order as those in the sample file.

Follow these steps to copy the user-defined data types from the sample project to your project.

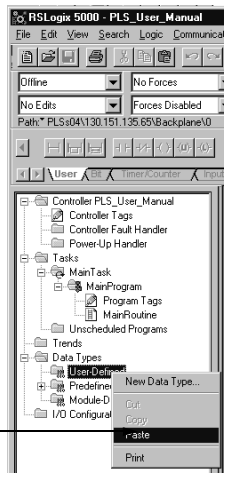
1. Open the sample project (Messaging_Configuration_1756_PLS.acd) in RSLogix 5000.
2. Open your project in a second window of RSLogix 5000.
3. Copy the user-defined data types in the sample project.

1. Select each user-defined tag
2. Right-click to display the menu
3. Select Copy



4. Paste the user-defined data type in your project.

1. Select User-Defined data type
2. Right-click to display the menu
3. Select Paste




## Creating Unique Tags

After you have copied the user-defined data types to your project, you must create unique tags for each in the Tag Editor of RSLogix 5000.

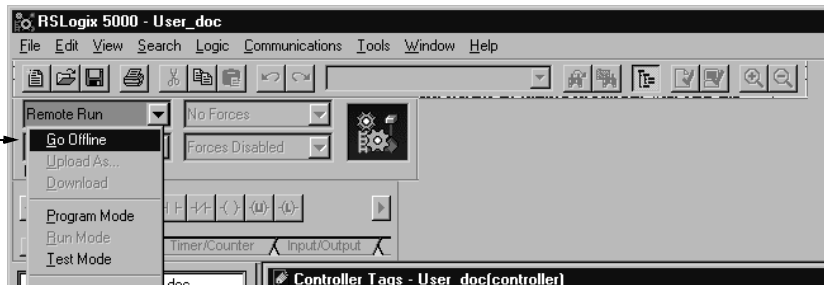
**IMPORTANT** You must create unique tags for data types in the order used in the sample project.

Module-defined and user-defined data types are listed in order later in this chapter and in the RSLogix 5000 online help files.

Follow these steps to create additional data types for use with multiple PLS modules:

1. Before you define the new tags, you must first go offline.

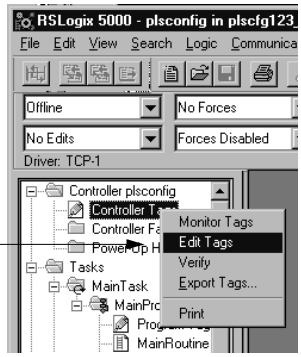
Pull down this menu and click here to go offline.



**2.** Access the Controller Tags.

1. Select Controller Tags
2. Right-click to display the menu
3. Select Edit Tags

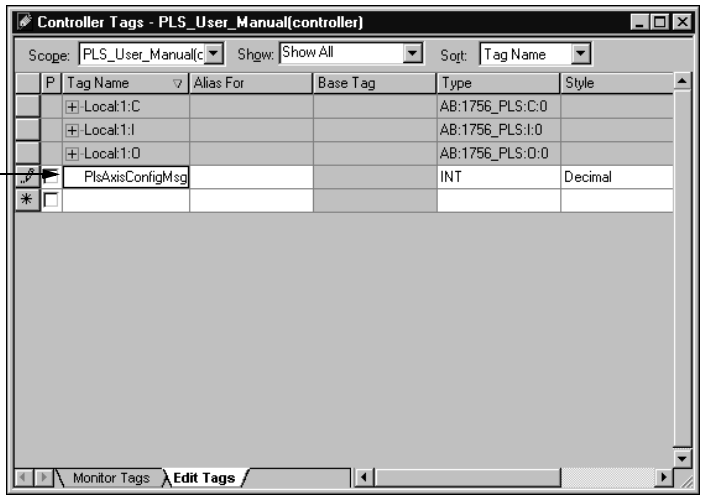
**IMPORTANT**: You must access the tags through the Edit Tags option to create new user-defined tags




**3.** Enter the tag name. The bottom line of the Edit Tags screen allows new tags.

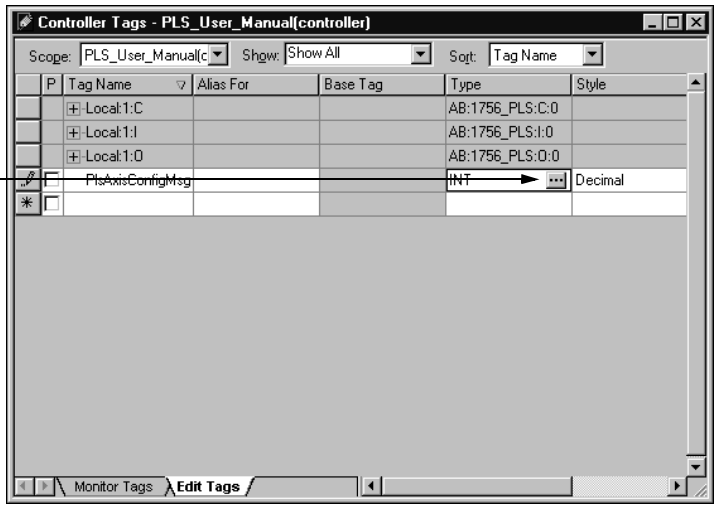
Move the cursor in the open line and type in the name of the first tag listed in the sample project's example user-defined data types.

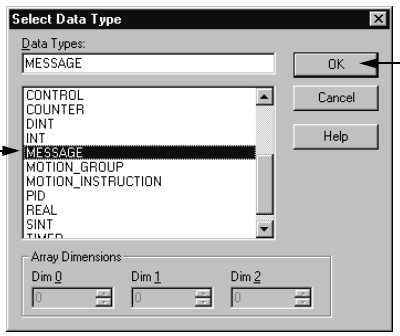
*PlsAxisConfigMsg*



**4.** Choose the correct data type for the type.

1. Click on the Type field and a pulldown appears.
2. Click on the pulldown to see the following menu.




3. Choose the data type that matches the data type used by the first data type, *PlsAxisCfg*.

4. Click here.

**5.** Click outside the newly created tag.

**6.** Repeat steps 1-5 for all tags listed in the user-defined data types.

**IMPORTANT** You must repeat these steps for each data type in the order they appear in the sample application.

## Changing Configuration Through the Tags

Some configurable features are changed on a module-wide basis and some on a point-by-point basis.

**IMPORTANT** Although you can change the value for any point in the tags, the module's configuration is not updated until you send another configuration message instruction, see page 5-23.

### Accessing the Tags

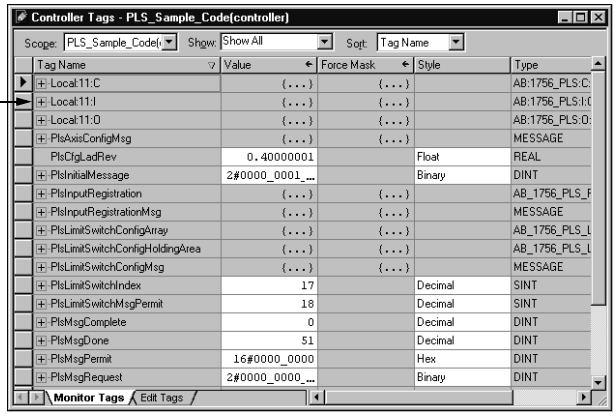
When you access tags, you have two options. You can:

- **monitor tags** - this option allows you to view tags and change their values
- **edit tags** - this option allows you to add or delete tags but not to change their values

1. Select Controller Tags
2. Right-click to display the menu
3. Select Monitor Tags

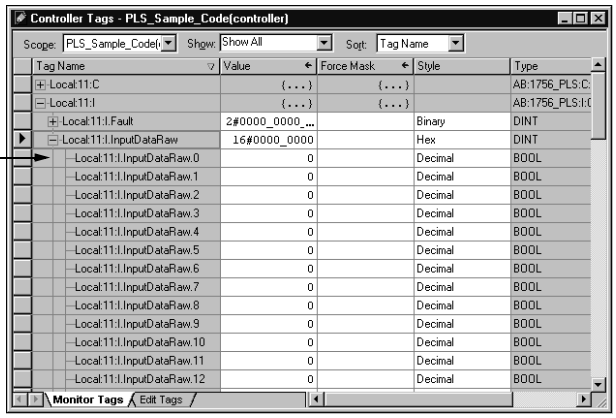
You can view tags here.

Click on the tag name you want to see




Configuration information is listed for each channel on the module.

For example, this screen shows Input Data Fault information.




## Configurable Features

There are two ways to change the configuration. You can either:

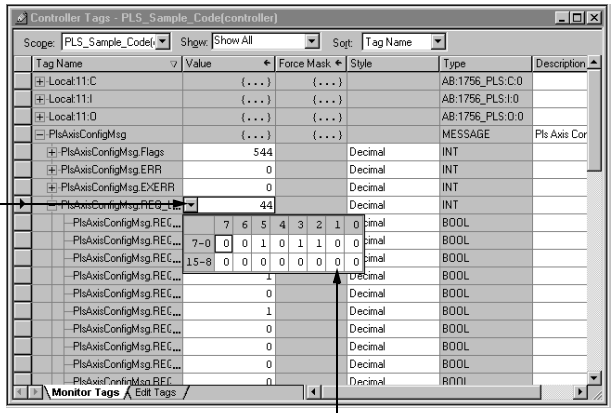
- use a pulldown menu

or

- highlight the value of a particular feature for a particular point and type a new value

*Pulldown menu*

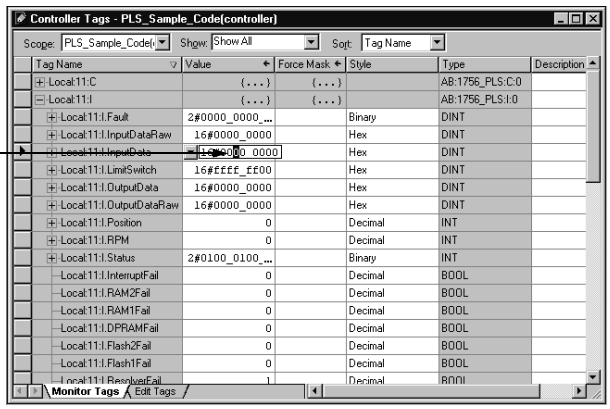
1. Click on the far left side of the Value column and a pulldown menu appears




2. Highlight the point that needs to be changed and type a valid new value

*Highlight value*

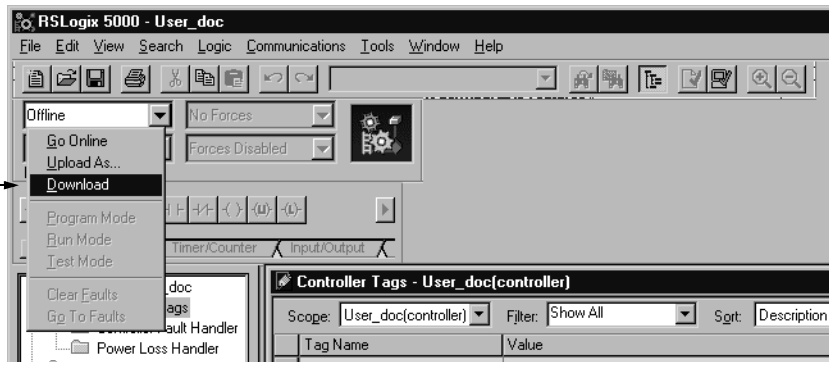
1. Highlight the value of the feature you want to change. Note that this series of values is listed in descending order of point number (i.e. the bit value for point 15 is listed first). Make sure you have highlighted the point you want to change.
2. Type in the valid new value.



## Downloading Initial Configuration

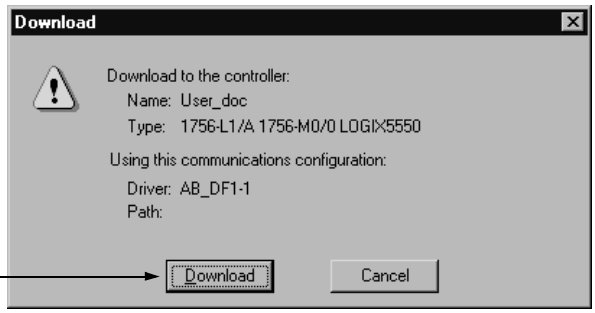
After you have changed the configuration data for a PLS module, the change does not actually take affect until you download the new program which contains that information. This downloads the entire program to the controller overwriting any existing programs.

Pull down this menu and click here to download the new data




RSLogix 5000 verifies the download process with this pop-up screen.

Click here to download new data




This completes the download process.

## Changing Configuration During PLS Module Operation

After the PLS module has begun operation, you can only change configuration by using ladder logic and message instructions.

Follow these steps to change PLS module configuration during operation:

1. Access the PLS data types through the tag monitor to make specific configuration changes, see page 5-13.

2. Use ladder logic and a configuration message instruction to send the configuration changes to the PLS module, see Using Ladder Logic on page 5-17.

## Using Ladder Logic

You use ladder logic to perform the following operations on your PLS module:

- change configuration during module operation
- check status
- perform run time services

Ladder logic uses message (MSG) instructions to exchange data between the controller and PLS module.

The sample project (Messaging_Configuration_1756_PLS.acd) described on page 5-2, provides examples of how to write configuration for outputs, read offset values and update registration information.

You can access the ladder logic by double-clicking on the MainRoutine portion of the MainProgram.


Double-click here to enter the Main Routine

**Rung 0** is an unconditional "jump to subroutine". It directs the program scan through the "Pls Read/Write Request" program. This subroutine checks the "request read/write word" and also sets the "permit" bits for the corresponding message to be sent.

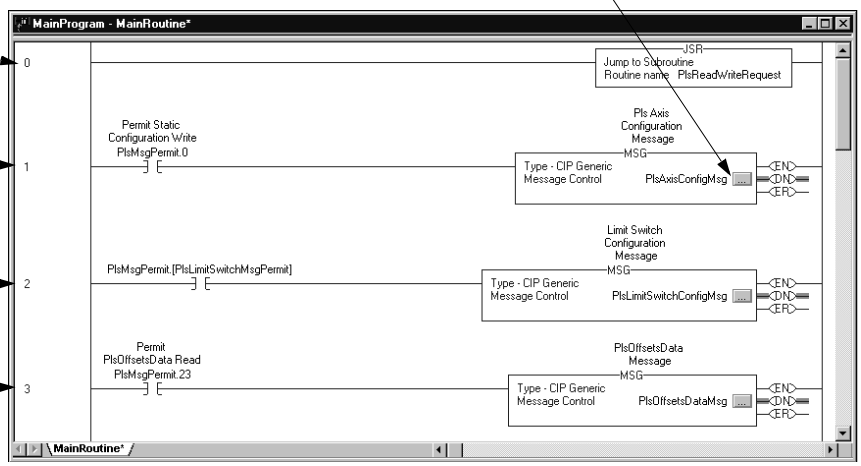
Click here to access the configuration message instruction for this rung.

**IMPORTANT**: Message instructions are explained on page 5-18.

**Rung 1** uses the PLS Axis Configuration Message to write configuration for each of the PLS module's 20 outputs. You must set the "Request Read/Write", and when the ladder code determines it is time to send the message, it will set the appropriate "Permit Read/Write" bit.

**Rung 2** uses the PLS Limit Switch message to write configuration for each of the PLS module's 20 outputs. You must set the "Request Read/Write", and when the ladder code determines it is time to send the message, it sets the appropriate "Permit Read/Write" bit.

**Rung 3** uses the Offset Data message to read the PLS module offset values. You must set the "Request Read/Write", and when the ladder code determines it is time to send the message, it sets the appropriate "Permit Read/Write bit.



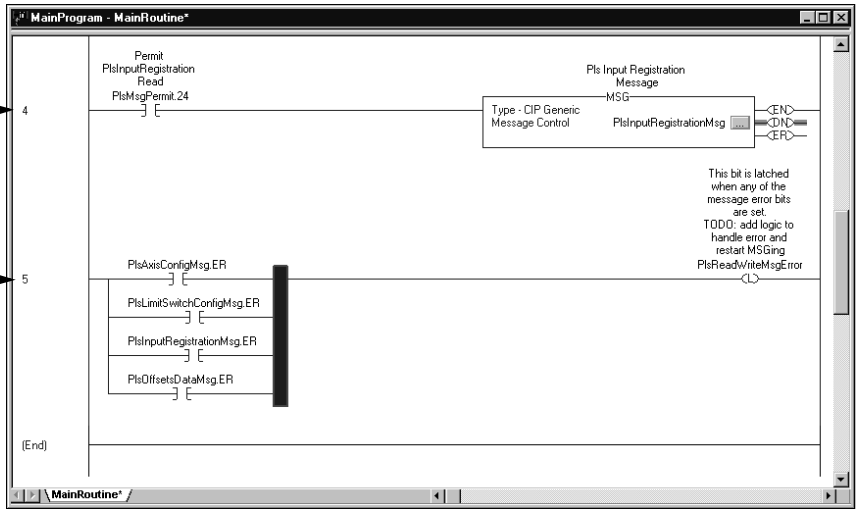

Rungs 4 and 5 are shown on the next page.

Rungs 4 and 5 are show below.

**Rung 4** uses the PLS Input Registration message to update PLS module registration information. You must set the "Request Read/Write", and when the ladder code determines it is time to send the message, it sets the appropriate "Permit Read/Write bit.

**Rung 4** writes a bit that is set when any of the CIP Generic messages have an error bit set.

**IMPORTANT:** You must use the error bits to perform an orderly shutdown or any other necessary procedure. You must modify the logic to support any error sequences that are required. When a message contains an error, the message is not reattempted. In this case, all messaging to the PLS module will stop. You must handle the error and then clear the error bit.




## Using Message Instructions

In ladder logic, you use message instructions to send services to the PLS module. MSG instructions send an explicit service to the module, causing specific behavior to occur, for example, changing Preset values.

Message instructions have the following characteristics:

- messages use unscheduled portions of system communications bandwidth
- one service is performed per instruction
- performing module services does not impede module functionality, such as measuring position

## Processing Real-Time Control and Module Services

Because MSG instructions use unscheduled portions of systems communications bandwidth, the services requested of an PLS module are not guaranteed to occur within a specific time period, although the module response typically occurs in much less than a second.

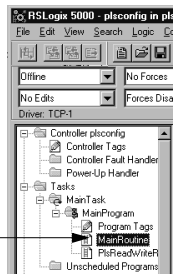
## One Service Performed Per Instruction

Message instructions only cause a module service to be performed once per execution. For example, if a message instruction sends a service to the module requesting new input data, the current input data is returned to the controller. The message instruction must then be reexecuted to update the data in the future.

# Creating a New Tag

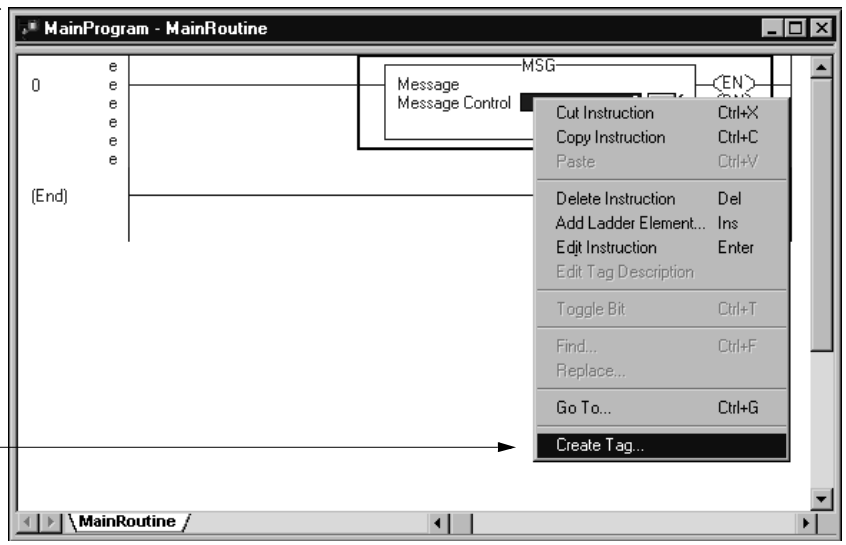
This ladder logic is written in the Main Routine section of RSLogix 5000.

Double-click here to enter the Main Routine




After adding a message instruction to a rung, you must create a tag for the message instruction

1. Right-click on the question mark (?) to see this pull-down menu.
2. Click here to Create a Tag.



You must fill in the following information when the New Tag pop-up screen appears:

**IMPORTANT** We suggest you name the tag to indicate what module service is sent by the MSG instruction. For example, the MSG instruction below is used to send PLS Axis Configuration, and the tag is named to reflect this.

.

Name the tag here.

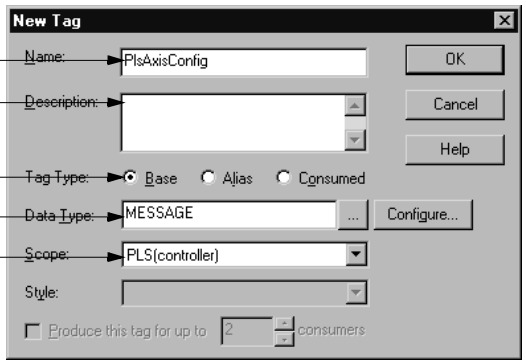
Enter an optional description here.

Choose the **Base** Tag Type here.

Choose the **Message** Data Type here.

Choose the **Controller** Scope here.

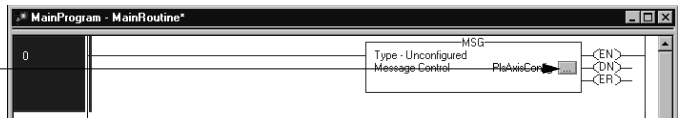
**IMPORTANT**: Message tags can only be created with the Controller Scope.




## Enter Message Configuration

After creating a new tag, you enter message configuration.

Click here to see the message configuration pop-up screens




Enter message configuration on the following screens:

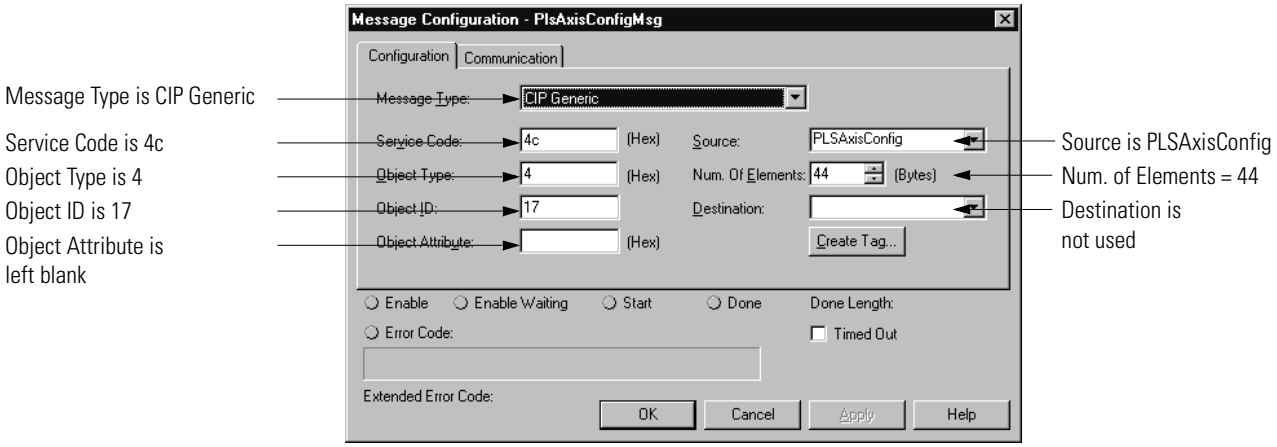
- Configuration pop-up screen
- Communications pop-up screen

A description of the purpose and set-up of each screen follows.

### *Configuration Pop-Up Screen*

This pop-up screen provides information about module location and service.

For example, the screen below shows the information needed to send an axis configuration message (module service) to a 1756-PLS module (where to perform service).




The following table contains information that must be entered on the configuration pop-up screen to perform PLS module services:

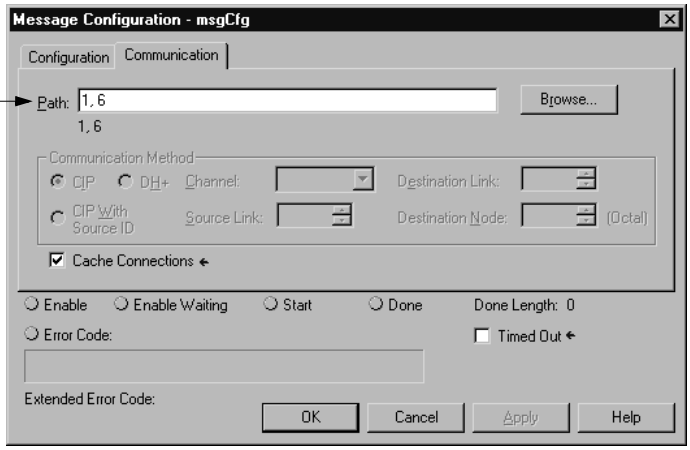
**Table 5.A**
**Service Codes for the 1756-PLS Module**

| Enter the following: | To send an Axis Configuration Message: | To send a Limit Switch Configuration Message: | To send a PLS Offsets Message: | To send a PLS Input Registration Message: |
|---|---|---|---|---|
| Service Code | 4c | 4c | 4b | 4b |
| Object Type | 4 | 4 | 4 | 4 |
| Object ID (Channel Number) | 17 | 33 | 37 | 36 |
| Source | PLSAxisConfig | PLS2LimitSwitchWork Var | N/A | N/A |
| Number of Elements | 44 | 124 | 0 | 0 |
| Destination | N/A | N/A | PLSOffsetsData | PLSInput Registration |

### *Communications Pop-Up Screen*

This pop-up screen provides information about the path of the MSG instruction. For example, the slot number of a 1756-PLS module distinguishes exactly which module a message is intended for.

When setting the slot number in the path, remember the PLS module is a three-slot module. See IMPORTANT on page 5-23.




The Communications pop-up screen is exactly the same for each MSG instruction of the plscfg.ACD file. Each MSG instruction uses a path of 1,6. The example program has the PLS module in slots 5, 6, and 7.

**IMPORTANT** The path will change, according to the ControlLogix chassis and slot number in which your PLS module resides. Make sure you account for each hop in the message's path.

### *Setting the Path*

If the PLS module resides in the same chassis as the controller, the path contains two numbers, accounting for the backplane and slot number of the PLS module. In the example above, a path of 1,6 is used. 1 = backplane between the controller and PLS module and 6 = slot number of the PLS.

If the PLS module resides in a chassis other than that of the controller, the path must specify each hop. For example, if the PLS module is remotely connected to controller via ControlNet, a longer path of 1,7,2,25,1,4 may be used. These digits specify for the following hops:

- 1 = backplane of first chassis
- 7 = slot number of 1756-CNB module providing ControlNet connection
- 2 = ControlNet connection
- 25 = ControlNet node address of second chassis
- 1 = backplane of second chassis
- 4 = slot number of 1756-PLS module in second chassis

The path described above uses values strictly for example purposes. You must specify a path to the location of your PLS module.

After you have created tags in the controller, you can use them to change configuration.

**IMPORTANT** The slot of a PLS module is the slot of its center section. For example, if the PLS module is installed in slots 0, 1, and 2, use slot 1 in the path.

## Sending New Configuration

After you make the configuration change, you must send a configuration message instruction to the PLS module. The configuration changes will not take affect until another configuration message instruction is sent.

# Module-Defined Configuration Data Types

This section lists the module-defined data types (input, output and configuration) as they appear when you create a new module.

**IMPORTANT** Module-defined and user-defined data types are also listed in the RSLogix 5000 online help files and the sample project provided with the software.

## Configuration Data Type

The Configuration data type configures scaling, zero offset and other features which apply to the entire module.

This data type is a one time only configuration. Values are sent from the controller to the PLS module once and those values are maintained until another PLS Axis configuration message is sent to the module, unlike input and output messages which are constantly updated between the processor and module.

If changes are made to rollover values, you must resend all Limit Switch configuration tags also.

**IMPORTANT** The Configuration data type affects the entire module, not just a particular output.

Also, remember if you are using multiple PLS modules in the same chassis, you must create unique configuration tags for each module.

The following table lists and defines Configuration data type members.

**Table 5.B**
**1756-PLS Configuration Data Type**

| Member Name | Type | Default Display | Description |
|---|---|---|---|
| OutputInvert | DINT | Hex | There is 1 bit per input/output. Setting each of these to 1 to inverts inputs/outputs. |
| InputInvert | DINT | Hex | There is 1 bit per input/output. Setting each of these to 1 to inverts inputs/outputs. |
| RPMFilter | INT | Hex | RPM Filter adjustment.<br>16#8000 = medium adjustment<br>16#F000 = slow adjustment |
| RolloverPosition | INT | Decimal | Maximum scaled position. For example, 359 degrees. |

**Table 5.B**
**1756-PLS Configuration Data Type**

| Member Name | Type | Default Display | Description |
|---|---|---|---|
| RolloverCounts | INT | Decimal | Maximum raw position. Set to 4095 unless you are using multi- or fractional turns. |
| ApplyFlags | SINT | Binary | Master Apply Flags register for MultidropSlaveResolver, ApplyZeroOffset, and ApplyPreset. |
| MultidropSlaveResolver | BOOL | Decimal | When there are multiple 1756-PLS modules in a chassis, it is possible to daisy chain them together so that they are all using the same resolver. In order to achieve this, it is necessary to set this bit for all 1756-PLS modules, except the first bit. |
| ApplyZeroOffset | BOOL | Decimal | Set this member to ignore or apply the Zero Offset value.<br>0 = Ignore the Zero Offset value<br>1 = Use the Zero Offset in this message |
| ApplyPreset | BOOL | Decimal | Set this member to ignore or apply the Preset value.<br>0 = Ignore the Preset value<br>1 = Use the Preset in this message |
| ArmFlags | SINT | Binary | Master ArmFlags register for ArmPresetInput, ArmNudgeUpInput, and ArmNudgeDown. |
| ArmPreset | BOOL | Decimal | Set this member to use PresetInput as a signal to set the position to the Preset value.<br>0 = Do not use PresetInput<br>1 = Use PresetInput |
| ArmNudgeUp | BOOL | Decimal | Set this member to use NudgeUpInputId to adjust the Zero Offset (and preset) by the NudgeUpOffset.<br>0 = Do not use NudgeUpInput<br>1 = Use NudgeUpInput |
| ArmNudgeDown | BOOL | Decimal | Set this member to use NudgeUpInputId to adjust the Zero Offset (and preset) by the NudgeDownOffset.<br>0 = Do not use NudgeDownInput<br>1 = Use NudgeDownInput |
| PresetInput | SINT | Decimal | Input point to use for Preset function. |
| NudgeUpInput | SINT | Decimal | Input point to use for Nudge Up function. |
| NudgeDownInput | SINT | Decimal | Input point to use for Nudge Down function. |
| ZeroOffset | INT | Decimal | Move the Position by this value to align Home. |
| Preset | INT | Decimal | Set the Position to this value when the PresetInput goes on. |
| NudgeUpOffset | INT | Decimal | Add this value to Zero Offset when the NudgeUpInput goes on. |
| NudgeDownOffset | INT | Decimal | Add this value to Zero Offset when the NudgeDownInput goes on. |

## PLS Input Data Type

The Input data type constantly reports the position and current state of digital inputs and digital outputs. This type also notifies the controller of memory, I/O module, status, and module failures.

**IMPORTANT** If you are using multiple PLS modules in the same chassis, create unique tags for each PLS module using the PLS Input Data user-defined data type.

The following table lists Input data type members.

**Table 5.C**
**1756-PLS Input Data Type**

| Member Name | Type | Default Style | Description |
|---|---|---|---|
| Fault | DINT | Binary | An abnormal condition was detected by the 1756-PLS hardware or software. A fault is identified by an error code in RSLogix 5000. |
| InputDataRaw | DINT | Hex | Input state at the Remote Terminal Block. |
| InputData | DINT | Hex | Input state after Override and/or Inversion. |
| LimitSwitch | DINT | Hex | Position within Limit Switch Setpoint. |
| OutputData | DINT | Hex | Limit switch including Enable Logic, but not Override or Inversion. |
| OutputDataRaw | DINT | Hex | Output state at the Remote Terminal Block. |
| Position | INT | Decimal | Scaled position. |
| RPM | INT | Decimal | Resolver revolutions per minute. |
| Status | INT | Binary | Reports the status of various events that occur within the 1756-PLS module. For example, module failures and RAM failures. |
| InterruptFail | BOOL | Decimal | An unexpected interrupt has occurred. |
| RAM2Fail | BOOL | Decimal | External RAM failure detected. |
| RAM1Fail | BOOL | Decimal | Internal RAM failure detected. |
| DPRAMFail | BOOL | Decimal | Dual-port RAM failure detected. |
| Flash2Fail | BOOL | Decimal | External flash memory failure detected. |
| Flash1Fail | BOOL | Decimal | Internal flash memory failure detected. |
| ResolverFail | BOOL | Decimal | If equal to 1, resolver is disconnected or its RPM is too fast. |
| RightIOFail | BOOL | Decimal | If equal to 1, I/O on right section has a failure. |
| LeftIOFail | BOOL | Decimal | If equal to 1, I/O on left section has a failure. |

**Table 5.C**
**1756-PLS Input Data Type**

| Member Name | Type | Default Style | Description |
|---|---|---|---|
| DcDcFail | BOOL | Decimal | A DC-DC converter failure has been detected. The DC-DC converter supplies isolated power to critical sections such as the resolver. |
| SpeedCompLimit | BOOL | Decimal | If equal to 1, the Speed compensation is not being performed as requested because resolver RPM is too fast. (Edges advance over previous/ next edge.) This clears once the RPM returns to within a valid range. |
| OutputsEnabled | BOOL | Decimal | If equal to 1, indicates that outputs are enabled. |
| ResolverRaw | INT | Hex | Bits 0-11 = Unscaled resolver value<br>Bits 12-15 = Turn count |
| CSTTimestamp[2] | DINT[2] | Decimal | Coordinated System Time Timestamp - Timestamp can be configured to indicate time that data changed and/or time that a diagnostic fault occurred. |

## PLS Output Data Type

The Output data type allows the processor to manually override input and output values. This type also verifies that communications are continuing between the processor and module. Finally, the Output data type enables PLS outputs and clears latched errors.

**IMPORTANT** If you are using multiple PLS modules in the same chassis, tags for the additional PLS modules must be created using data types as listed on page .

The following table lists Output data type members.

**Table 5.D**
**1756-PLS Output Data Type**

| Member Name | Type | Default Style | Description |
|---|---|---|---|
| OutputOverrideEnable | DINT | Hex | Setting any of the bits in this member causes the corresponding bits in OutputDataRaw to be assigned the state specified by the corresponding bit specified in OutputOverrideValue. |
| OutputOverrideValue | DINT | Hex | The data source for the bits of OutputDataRaw when the corresponding bits of OutputOverrideEnable are set. |

**Table 5.D**
**1756-PLS Output Data Type**

| Member Name | Type | Default Style | Description |
|---|---|---|---|
| InputOverrideEnable | DINT | Hex | Setting any of the bits in this member causes the corresponding bits in InputDataRaw to be assigned the state specified by the corresponding bit specified in InputOverrideValue. |
| InputOverrideValue | DINT | Hex | The data source for the bits of InputDataRaw when the corresponding bits of InputOverrideEnable are set. |
| Control | SINT | Binary | Instructs the 1756-PLS to perform certain control operations (ie: clear errors or enable outputs). |
| EnableOutputs | BOOL | Decimal | 0 = Turn outputs OFF<br>1 = Turn outputs ON |
| ClearErrors | BOOL | Decimal | A rising edge transition ( 0 to 1) clears all error bits. The following 12 errors are cleared:<br>spurious interrupt<br>external RAM fail<br>internal RAM fail<br>dual port RAM fail<br>external flash fail<br>internal flash fail<br>resolver fail<br>right I/O fail<br>left I/O fail<br>DCDC fail<br>local I/O fail<br>speed compensation overflow |

## User-Defined Configuration Data Types

This section lists the user-defined data types as they appear in the example ladder logic application with RSLogix 5000, Version 2.5.

**IMPORTANT** Because versions of RSLogix 5000 later than V2.5 may have updated user-defined data type information, see your sample ladder logic application for the most detailed and current explanation of the example user-defined data types.

Also remember, you cannot configure your PLS module with both module-defined and user-defined data types.

You must use the following user-defined data types:

- Limit switch configuration
- Offset configuration
- Registration configuration

## Limit Switch Configuration

The Limit Switch Configuration data type has a one to one correspondence to the outputs on the module (i.e. for each output there is a specific Limit Switch Configuration tag).

**IMPORTANT** A change in one output's PLS Limit Switch Configuration will not affect the configuration of another output.

Also, if you are using multiple PLS modules in the same chassis, tags for each PLS module must be created using the data types below.

The following table lists and defines PLS Limit Switch Configuration tags.

**Table 5.E**
**1756-PLS Limit Switch Configuration Data Type**

| Member Name | Type | Default Style | Description |
|---|---|---|---|
| Output | SINT | Decimal | Output Point controlled by this limit switch. |
| EnableStitching | BOOL | Decimal | Cycle On and Off within the Setpoint. |
| ApplyZeroOffset | BOOL | Decimal | Set this member to apply or ignore the Zero offset value (Value affects only this output)<br>0 = Ignore Zero Offset value<br>1 = Apply Zero Offset value to this output<br>This member is useful if you are using Preset |
| ApplyPreset | BOOL | Decimal | Set this member to apply or ignore the Zero offset value (Value affects only this output)<br>0 = Ignore Preset value<br>1 = Apply Preset value to this output<br>This member is useful if you are using Nudge |
| ArmAndEnable | BOOL | Decimal | Set this member to use the AndEnable function<br>0 = Do not use AndEnable<br>1 = Use AndEnable |
| ArmPulseEnable | BOOL | Decimal | Set this member to use the PulseEnable function<br>0 = Do not use PulseEnable<br>1 = Use PulseEnable |
| ArmWindowedEnable | BOOL | Decimal | Set this member to use the WindowedEnable function<br>0 = Do not use WindowedEnable<br>1 = Use WindowedEnable |

**Table 5.E**
**1756-PLS Limit Switch Configuration Data Type**

| Member Name | Type | Default Style | Description |
|---|---|---|---|
| ArmPreset | BOOL | Decimal | Set this member to use the Preset function<br>0 = Do not use Preset<br>1 = Use Preset |
| ArmNudgeUp | BOOL | Decimal | Set this member to use the NudgeUp function on the NudgeUpInput<br>0 = Do not use NudgeUp<br>1 = Use NudgeUp |
| ArmNudgeDown | BOOL | Decimal | Set this member to use the NudgeDown function on the NudgeDownInput<br>0 = Do not use NudgeDown<br>1 = Use NudgeDown |
| Unused 17 | SINT | | |
| RpmEnableLowLimit | INT | Decimal | Output = 0 if the resolver RPM is below this value |
| RpmEnableHighLimit | INT | Decimal | Output = 0 if the resolver RPM is above this value |
| AndEnableInput | SINT | Decimal | Input number to use for AndEnable function. |
| PulseEnableInput | SINT | Decimal | Input number to use for PulseEnable function. |
| WindowedEnableInput | SINT | Decimal | Input number to use for WindowedEnable function. |
| PresetInput | SINT | Decimal | Input number to use for Preset function. |
| NudgeUpInput | SINT | Decimal | Input number to use for NudgeUp function. |
| NudgeDownInput | SINT | Decimal | Input number to use for NudgeDown function. |
| Unused 26 | SINT | | |
| Unused 27 | sint | | |
| WindowedEnableLowLimit | INT | Decimal | Edge of position window for input to turn ON and energize output. |
| WindowedEnableHighLimit | INT | Decimal | Other edge of position window for input to turn ON and energize output. |
| OnTimeMaxMs | INT | Decimal | Maximum time (in ms) an output remains ON. |
| ZeroOffset | INT | Decimal | Zero Offset value for Limit Switch. |
| Preset | INT | Decimal | Preset value for Limit Switch. |
| NudgeUpOffset | INT | Decimal | Nudge Up Offset value for Limit Switch. |
| NudgeDownOffset | INT | Decimal | Nudge Down Offset value for Limit Switch. |
| LeadSpeedDistance | INT | Decimal | Lead Speed Distance value for Limit Switch. |
| LeadSpeedRpm | INT | Decimal | Lead Speed RPM value for Limit Switch. |
| TrailSpeedDistance | INT | Decimal | Trail Speed Distance value for Limit Switch. |
| TrailSpeedRpm | INT | Decimal | Trail Speed RPM value for Limit Switch. |
| Unused 34 | INT | | |

**Table 5.E**
**1756-PLS Limit Switch Configuration Data Type**

| Member Name | Type | Default Style | Description |
|---|---|---|---|
| SetPointEnableMask | DINT | Hex | Any of the first 10 bits of this mask may be set to enable each of the corresponding limit switch setpoints. |
| SetPoint | AB_1756_PLS_SetPoint_C [10] | | Calls the SetPoint configuration tag and its related members. Refer to the AB_1756_PLS_SetPoint_C tag discussed below. |
| LowLimit | INT | Decimal | Edge of ON region. |
| HighLimit | INT | Decimal | Other Edge of ON region. |
| OnStitchDistance | INT | Decimal | If EnableStitching is active, the length of the ON part of Stitch. |
| OffStitchDistance | INT | Decimal | If EnableStitching is active, the length of the OFF part of Stitch.. |

## Offset Data Type

The Offset data type reports the active Zero Offset and Preset for the axis and each limit switch. Using this function, you can preserve a home value defined during machine setup.

**IMPORTANT** If you are using multiple PLS modules in the same chassis, tags for each PLS module must be created using the data types below.

The following table lists Offset data type members.

**Table 5.F**
**1756-PLS Offset Data Type**

| Member Name | Type | Default Style | Description |
|---|---|---|---|
| Axis | AB_1756_PLS_OffsetElement_I | Decimal | Calls the OffsetElement input tag and its related members. Refer to the OffsetElement:I:0 tag members discussed below that hold global Axis-wide values. |
| ZeroOffset | INT | Decimal | Current value of Axis Zero Offset. Value generated by Preset event or Zero Offset +/- any Nudge events. |
| Preset | INT | Decimal | Current value of Axis Preset. Preset value +/- any Nudge events. |

**Table 5.F**
**1756-PLS Offset Data Type**

| Member Name | Type | Default Style | Description |
|---|---|---|---|
| LimitSwitch | AB_1756_PLS_OffsetElement_I[20] | Decimal | Calls the OffsetElement input tag and its related members. Refer to the OffsetElement:I:0 tag members discussed below. |
| ZeroOffset | INT | Decimal | Current value of Axis Zero Offset. Value generated by Preset event or Zero Offset +/- any Nudge events. |
| Preset | INT | Decimal | Current value of Axis Preset. Preset value +/- any Nudge events. |

## Input Registration Data Type

The Input Registration data type reports the position at which each module input last went ON or OFF. Using this function, you can determine proper setpoint values, perform dimensional inspection, or troubleshoot failed input sensors.

**IMPORTANT** If you are using multiple PLS modules in the same chassis, tags for each PLS module must be created using the data types below.

The following table lists Input Registration data type tags.

**Table 5.G**
**1756-PLS Input Registration Data Type**

| Member Name | Type | Default Style | Description |
|---|---|---|---|
| ResolverRaw | INT | Decimal | Contains a snapshot of PlsInputData.ResolverRaw. |
| Position | INT | Decimal | Contains a scaled snapshot of PlsInputData position. |
| Input | AB_1756_PLS_RegistrationInput_I[32] | | Calls the RegistrationInput input tag and its related members. Refer to the AB_1756_PLS_RegistrationInput_I tag discussed below. |
| Rise | AB_1756_PLS_RegistrationElement_I | | A 0 to 1 transition of input (not InputRaw). Calls the RegistrationElement input tag and its related members. Refer to the AB_1756_PLS_RegistrationElement_I tag discussed below. |
| ResolverRaw | INT | Decimal | Contains a snapshot of PlsInputData.ResolverRaw. |

**Table 5.G**
**1756-PLS Input Registration Data Type**

| Member Name | Type | Default Style | Description |
|---|---|---|---|
| Position | INT | Decimal | Contains a scaled snapshot of PlsInputData position. |
| Fall | AB_1756_PLS_RegistrationElement_I | | A 0 to 1 transition of input (not InputRaw). Calls the RegistrationElement input tag and its related members. Refer to the AB_1756_PLS_RegistrationElement_I tag discussed below. |
| ResolverRaw | INT | Decimal | Contains a snapshot of PlsInputData.ResolverRaw. |
| Position | INT | Decimal | Contains a scaled snapshot of PlsInputData position. |

## Error Codes

The table below lists possible configuration errors that your PLS module may report. They are displayed on the Connection tab of the Module Properties section in RSLogix 5000 and in the .EXERR field of the message variable when a reconfiguration error occurs.

**Table 5.H**
**1756-PLS Module Error Codes**

| Code: | Explanation: |
|---|---|
| 16#0008 | RPMFilter out of range. Check that the value entered is between 16#1000 and 16#ffff. |
| 16#0010 | RolloverPosition out of range. Check that the value entered is less than 16#ffff. |
| 16#0012 | RolloverCounts out of range. Check that the value entered is less than 16#ffff. |
| 16#0018 | PresetInput is greater than 31. Enter a value less than 31. |
| 16#0019 | NudgeUpInput is greater than 31. Enter a value less than 31. |
| 16#0020 | NudgeDownInput is greater than 31. Enter a value less than 31. |
| 16#0022 | The ZeroOffset value exceeds the value of RolloverPosition. Check that the ZeroOffset entered is less than or equal to the RolloverPosition value. |
| 16#0024 | Preset exceeds RolloverPosition. Enter a value less than the RolloverPosition value. |
| 16#0026 | NudgeUpOffset exceeds the value of RolloverPosition. Check that the NudgeUpOffset entered is less than or equal to the RolloverPosition value. |
| 16#0028 | NudgeDownOffset exceeds the value of RolloverPosition. Check that the NudgeDownOffset entered is less than or equal to the RolloverPosition value. |

The following table lists the additional fault information for the 1756-PLS module Axis Configuration.

**Table 5.I**
**1756-PLS Module Axis Configuration Error Codes**

| Fault Code: | Explanation: |
|---|---|
| 8 | RPMFilter out of range. Check that the value entered is between 16#1000 and 16#ffff. |
| 10 | RolloverPosition out of range. Check that the value entered is less than 16#ffff. |
| 12 | RolloverCounts out of range. Check that the value entered is less than 16#ffff. |
| 18 | PresetInput is greater than 31. Enter a value less than 31. |
| 19 | NudgeUpInput is greater than 31. Enter a value less than 31. |
| 20 | NudgeDownInput is greater than 31. Enter a value less than 31. |
| 22 | The ZeroOffset value exceeds the value of RolloverPosition. Check that the ZeroOffset entered is less than or equal to the RolloverPosition value. |
| 24 | Preset exceeds RolloverPosition. Enter a value less than the RolloverPosition value. |
| 26 | NudgeUpOffset exceeds exceeds the value of RolloverPosition. Check that the NudgeUpOffset entered is less than or equal to the RolloverPosition value. |
| 28 | NudgeDownOffset exceeds exceeds the value of RolloverPosition. Check that the NudgeDownOffset entered is less than or equal to the RolloverPosition value. |

## Chapter Summary and What's Next

In this chapter you learned about:

- configuring ControlLogix PLS modules
- editing module configuration
- configuration tags

Move on to chapter 6 to troubleshoot your module.

# Troubleshooting Your Module

## What This Chapter Contains

This chapter describes the indicators on the ControlLogix PLS module and how to use them to troubleshoot the module. The following table describes what this chapter contains and its location.

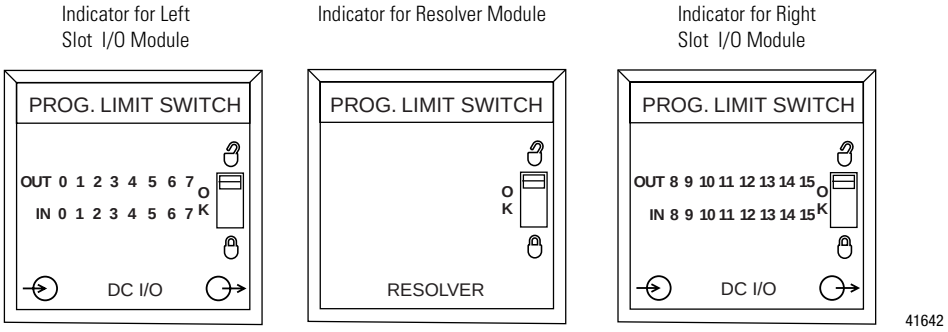


## Using Indicators to Troubleshoot Your Module

Each ControlLogix PLS module has indicators which show input and output status. LED indicators are located on the front of the module.

### LED indicators for the PLS module

The 1756-PLS module uses the following status indicators.




| LED indicator: | This display: | Means: | Take this action: |
|---|---|---|---|
| Resolver OK | Flashing green light | The resolver module is operating under normal conditions. | None |
| Resolver OK | Flashing red light | One of the following:<br>1. NVS configuration needs to be updated.<br>2. Module is currently updating NVS configuration. | 1. Update NVS configuration.<br>2. Wait for NVS configuration update to finish. |
| OK | Steady red light | The module must be replaced. | Replace the module. |
| I/O State[1] | Yellow | The point is active. | None |

[1] If the I/O State indicator is ON but the output does not drive the load, an overload or short circuit condition may exist. Remove the load and toggle the output OFF-ON-OFF (observe status indicator). Reapply a proper load and check if the output is operational. If, after performing these steps, the output is not operational, replace the module.

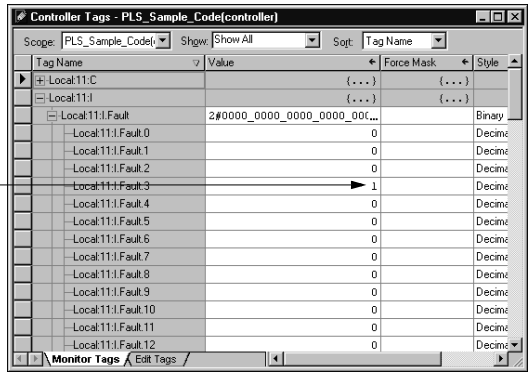
## Using RSLogix 5000 to Troubleshoot Your Module

In addition to the LED display on the module, RSLogix 5000 will display fault conditions. General module faults are also reported in the Tag Editor. You must monitor the Tag Editor to see when a fault has occurred.

The screen below displays fault notification in RSLogix 5000.

**Notification in Tag Editor**

A fault has occurred for any point that lists the number 1 in the Fault line




For a detailed listing of the possible faults, their causes and suggested solutions, see Module Faults in the online help.

## Chapter Summary and What's Next

In this chapter you learned about troubleshooting the module.

# Specifications

This appendix provides the specifications for the ControlLogix Programmable Limit Switch module. These specifications are broken into the following sections:

## 1756-PLS Module General Specifications

| **General** | |
|---|---|
| Module Location | 3 contiguous slots of a 1756 ControlLogix chassis |
| Module Configuration | PLS left section - 2 groups of 4 outputs and 4 inputs each<br>PLS center section - resolver interface and I/O control<br>PLS right section - 2 groups of 4 outputs and 4 inputs each |
| Backplane Requirements | 1A @ 5.1V dc & 125mA @ 24V dc |
| Total Power Dissipation<br>I/O @ Nominal dc Voltage<br><br>I/O @ Maximum dc Voltage | <br>22.62W @ 30°C<br>18.22W @ 60°C<br>25.7W @ 30°C<br>21.3W @ 60°C |
| Thermal Dissipation<br>I/O @ Nominal dc Voltage<br><br>I/O @ Maximum dc Voltage | <br>77.23 BTU/hr @ 30°C<br>62.2 BTU/hr @ 60°C<br>87.74 BTU/hr @ 30°C<br>72.72 BTU/hr @ 60°C |
| Field Wiring Arms and Housings | 3 - 20 Position RTBs (1756-TBNH or TBSH)[1] |
| RTB Screw Torque (NEMA) | 7-9 inch-pounds (0.8-1Nm) |
| RTB Keying | User defined |
| Screwdriver Width for RTB | 5/16 inch (8mm) maximum |
| Environmental Conditions<br>Temperature Range<br><br>Storage Humidity<br>Operating Humidity | <br>0 - 60°C (32° - 140°F) @ nominal dc I/O voltage<br>0 - 50°C (32° - 122°F) above nominal dc I/O voltage<br>-40 - 85°C (-40° - 185°F)<br>0 - 95° non-condensing |
| Left & Right Slot Conductors<br>Wire Size<br>Insulation<br>Category | <br>22-14 gauge (2mm$^2$) stranded[1]<br>3/64 inch (1.2mm) insulation maximum<br>1[2, 3] |
| Center Slot Conductors<br>Wire Size<br>Cable Distance<br>Category | <br>Alpha Cable #6054C (use 3 of 4 twisted pairs)<br>300 ft (100m) maximum<br>2[2, 3] |
| Agency Certification<br>(when product or packaging is marked) |  <br>  Class I Div 2 Hazardous[4]<br>  marked for all applicable directives<br>  marked for all applicable acts<br>N223 |

[1] Maximum wire size will require extended housing - 1756-TBE.
[2] Use this conductor category information for planning conductor routing as described in the system level installation manual.
[3] Refer to publication 1770-4.1, "Programming Controller Wiring and Grounding Guidelines."
[4] CSA certification–Class I Division 2, Group A, B, C, D or nonhazardous locations.

## 1756-PLS Resolver Interface Specifications

| **Resolver Section Specifications** | |
|---|---|
| Resolver Location | Center section |
| Compatible Resolver | Allen-Bradley resolver 846-SJxxxx-R3-x<br>(x = customer options) |
| Resolver Interface | 2Vrms, reference output (Differential pair)<br>2Vrms, sine and cosine inputs (2 Differential pairs) |
| Reference Voltage | 2Vrms ± 20% |
| Reference Frequency | 5 kHz ± 20% |
| Digital Resolution | 12 bits (4096 counts from hardware) |
| Angular Resolution | 0.088°/bit |
| Digital Count Range | 0 - 4095 (decimal) |
| Maximum Tracking Rate | ± 1800 RPM |
| Repeatability | ± 0.0488% of full scale |
| Accuracy | ± 0.0976% of full scale |
| Isolation Voltage<br>User to system | <br>100% tested at 1700V dc for 1s |

## 12 to 24 VDC Input Specifications

| **I/O Sections–Inputs** | |
|---|---|
| Number of Inputs | 16 (2 groups of 4 per I/O section) |
| Input Power Dissipation/Slot<br>Input @ Nominal dc Voltage<br>Input @ Maximum dc Voltage | <br>1.86W @ 60°C<br>2.8W @ 60°C |
| Thermal Dissipation<br>Input @ Nominal dc Voltage<br>Input @ Maximum dc Voltage | <br>6.35 BTU/hr<br>9.56 BTU/hr |
| On-State Voltage<br>Nominal Range<br>Maximum<br>Minimum | <br>10.8 - 26.4V dc<br>31.2V dc<br>10V dc |
| Off-State Voltage<br>Maximum | <br>5V dc |
| On-State Current<br>Maximum<br>Minimum | <br>10mA<br>3mA |
| Off-State Current<br>Maximum | <br>1.5mA |

| **I/O Sections–Inputs (continued)** | |
|---|---|
| Input Impedance @ 24V dc<br>Maximum | <br>3.3kΩ |
| Input Delay Time (nominal voltage)<br>Off to On(typical)<br>(maximum)<br>On to Off(typical)<br>(maximum) | <br><15µs @ 30°C<br><150µs @ 60°C<br><30µs @ 30°C<br><200µs @ 60°C |
| Reverse Polarity Protection | Yes |
| Isolation Voltage<br>Group to Group<br><br>User to System | <br>100% Tested at 2546V dc for 1s<br>(250V ac maximum continuous voltage between groups)<br>100% Tested at 2546V dc for 1s |

## 12 TO 24 VDC Output Specifications

| **I/O Sections–Outputs** | |
|---|---|
| Number of Outputs | 16 (2 groups of 4 per I/O section) |
| Output Power Dissipation/Slot<br>Output @ Nominal dc Voltage<br><br>Output @ Maximum dc Voltage | <br>5.4W @ 30°C<br>3.2W @ 60°C<br>6W @ 30°C<br>3.8W @ 60°C |
| Thermal Dissipation<br>Output @ Nominal dc Voltage<br><br>Output @ Maximum dc Voltage | <br>18.43 BTU/hr @ 30°C<br>10.93 BTU/hr @ 60°C<br>21.48 BTU/hr @ 30°C<br>11.93 BTU/hr @ 60°C |
| Output Power Dissipation/Slot<br>Thermal Dissipation | 3.2W @ 60°C<br>10.93 BTU/hr @ 60°C |
| On-State Voltage<br>Nominal Range<br>Maximum<br>Minimum | <br>10.8 - 26.4V dc<br>31.2V dc<br>10V dc |
| Voltage Drop/Output<br>Maximum | <br>0.55V dc |
| Output Switch Times | Switching 1A @ 24V dc |
| Output Delay Time<br>Off to On<br>On to Off | <br><15µs @ 60°C<br><25µs @ 60°C |

| **I/O Sections–Outputs (continued)** | |
|---|---|
| Current Rating<br>Per Point<br>(Derate 16.7mA/°C above 30°C)<br>Per Group<br>(Derate 66.8mA/°C above 30°C)<br>Per Module<br>(Derate 133.6mA/°C above 30°C) | <br>1A maximum @ 30°C<br>0.5A maximum @ 60°C<br>4A maximum @ 30°C<br>2A maximum @ 60°C<br>8A maximum @ 30°C<br>4A maximum @ 60°C |
| Surge Current/Point | 2A for 10mS every 1s @ 60°C |
| Load Current<br>Minimum | <br>40mA |
| Off-State Leakage Current/Output<br>Typical<br>Maximum | <br><10μA @ 60°C<br>300μA @ 60°C |
| Output Short Circuit Protection | Electronic (No indication of fault)<br>Remove load & toggle output On-Off to restore. |
| Current Limit | <4A (Overload) |
| Reverse Polarity Protection | Yes (Current limited)<br>If wired incorrectly, outputs may be permanently disabled. |
| Isolation Voltage<br>Group to Group<br><br>User to System | <br>100% Tested at 2546V dc for 1s<br>(250V ac maximum continuous voltage between groups)<br>100% Tested at 2546V dc for 1s |

# Notes:

## S



## T



## U

## W



## Z

# Allen-Bradley
# Publication Problem Report

If you find a problem with our documentation, please complete and return this form.

Pub. Name ControlLogix Programmable Limit Switch Module User Manual

Cat. No. 1756-PLS Pub. No. 1756-UM002A-US-P Pub. Date February Part No. 957259-80

| Check Problem(s) Type: | Describe Problem(s) | Internal Use Only |
|---|---|---|
| ☐ Technical Accuracy | ☐ text ☐ illustration | |
| ☐ Completeness<br>What information is missing? | ☐ procedure/step ☐ illustration ☐ definition<br>☐ example ☐ guideline ☐ feature<br>☐ explanation ☐ other | ☐ info in manual (accessibility)<br>☐ info not in |
| ☐ Clarity<br>What is unclear? | | |
| ☐ Sequence<br>What is not in the right order? | | |
| ☐ Other Comments<br>Use back for more comments. | | |

Your Name _____ Location/Phone _____

Return to: Marketing Communications, Allen-Bradley., 1 Allen-Bradley Drive, Mayfield Hts., OH 44124-6118Phone: (440) 646-3176
FAX: (440) 646-4320

PLEASE FASTEN HERE (DO NOT STAPLE)

Other Comments

PLEASE FOLD HERE

PLEASE REMOVE

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

**BUSINESS REPLY MAIL**

**FIRST-CLASS MAIL PERMIT NO. 18235 CLEVELAND OH**

**POSTAGE WILL BE PAID BY THE ADDRESSEE**


Rockwell Automation

Allen-Bradley

1 ALLEN BRADLEY DR
MAYFIELD HEIGHTS OH 44124-9705

**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, Vorstlaan/Boulevard du Souverain 36, 1170 Brussels, 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-UM002A-US-P - February 2000 PN 957259-80
Supersedes Publication 1756-6.5.20 – June 1999