Allen-Bradley

GuardPLC™ Controller Systems

Bulletin 1753, 1754, 1755

User Manual

Rockwell Automation

Important User Information

Solid state equipment has operational characteristics differing from those of electromechanical equipment. *Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls* (Publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://www.ab.com/manuals/gi) describes some important differences between solid state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

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

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

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

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

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

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

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

Preface

Chapter 1
Overview of Safety Controllers

Chapter 2
Installation

Chapter 3

Wiring

Chapter 4

Connecting to the GuardPLC Controller

Chapter 12

Peer-to-Peer Communication Overview

Chapter 13

Configuring Peer-to-Peer Communication

Appendix B
System Variables

Appendix C
Replacing the Backup Battery

Index

Who Should Use This Manual

Use this manual if you are responsible for designing, installing, programming, or troubleshooting control systems that use GuardPLC controllers.

Personnel responsible for installation, programming, operation, and troubleshooting of safety-related controllers must be familiar with relevant safety standards for Programmable Electronic Systems (PES).

Purpose of this Manual

The manual only briefly describes the safety concept of the GuardPLC family of controllers. Its purpose is to provide the information on installing and operating your controller system.

For detailed information on the safety policy regarding GuardPLC controllers, including information on the controller's central functions, input and output channels, operating system, application program safety and regulations for use, refer to the *GuardPLC Controller Systems Safety Reference Manual*, publication number 1755-RM001.

Related Documentation

The table on the following page lists documents that contain additional information concerning Rockwell Automation GuardPLC products. If you would like a copy, you can:

- download a free electronic version from the internet: www.theautomationbookstore.com
- purchase a printed manual by:
  - contacting your local distributor or Rockwell Automation representative
  - visiting www.theautomationbookstore.com and placing your order
  - calling 1.800.963.9548 (USA/Canada) or 001.330.725.1574 (Outside USA/Canada)

For	Read this Document	Document Number
In-depth information on the safety concept of GuardPLC controller systems	GuardPLC Controller Systems Safety Reference Manual	1755-RM001
Information on installing 1754-L28BBB GuardPLC 1200 Controllers.	1754-L28BBB GuardPLC 1200 Installation Instructions	1754-IN001
Information on installing 1755-A6 GuardPLC 2000 I/O Chassis.	1755-A6 GuardPLC 2000 I/O Chassis Installation Instructions	1755-IN001
Information on installing 1755-L1 GuardPLC 2000 Controllers.	1755-L1 GuardPLC 2000 Controller Installation Instructions	1755-IN002
Information on installing 1755-IB24XOB16 GuardPLC 2000 Digital Input Modules.	1755-IB24XOB16 GuardPLC 2000 Digital Input Module Installation Instructions	1755-IN003
Information on installing 1755-IF8 GuardPLC 2000 Analog Input Modules.	1755-IF8 GuardPLC 2000 Analog Input Module Installation Instructions	1755-IN004
Information on installing 1755-OF8 GuardPLC 2000 Analog Output Modules.	1755-OF8 GuardPLC 2000 Analog Output Module Installation Instructions	1755-IN005
Information on installing 1755-HSC GuardPLC 2000 High Speed Counter Modules.	1755-HSC GuardPLC 2000 High Speed Counter Module Installation Instructions	1755-IN006
Information on installing 1755-PB720 GuardPLC 2000 Power Supplies.	1755-PB720 GuardPLC 2000 Power Supply Installation Instructions	1755-IN007
Information on installing GuardPLC 1600 controllers	GuardPLC 1600 Controllers Installation Instructions	1753-IN001
Information on installing GuardPLC 1800 controllers	GuardPLC 1800 Controllers Installation Instructions	1753-IN002
Information on installing GuardPLC 1753-IB20XOB8 I/O module	GuardPLC 1753-IB20XOB8 I/O Module Installation Instructions	1753-IN003
Information on installing GuardPLC 1753-IB16 Input Module	GuardPLC 1753-IB16 Input Module Installation Instructions	1753-IN004
Information on installing GuardPLC 1753-OB16 Output Module	GuardPLC 1753-OB16 Output Module Installation Instructions	1753-IN005
Information on installing and uninstalling RSLogix Guard PLUS software	RSLogix Guard PLUS Programming Software Installation Instructions	1753-IN006
Information on installing and uninstalling the GuardPLC OPC Server	GuardPLC OPC Server Installation Instructions	1753-IN007
In-depth information on grounding and wiring Allen-Bradley programmable controllers	Allen-Bradley Programmable Controller Grounding and Wiring Guidelines	1770-4.1
A description of important differences between solid-state programmable controller products and hard-wired electromechanical devices	Application Considerations for Solid-State Controls	SGI-1.1
An article on wire sizes and types for grounding electrical equipment	National Electrical Code - Published by the National Fire Protection Association of Boston, MA.	
A glossary of industrial automation terms and abbreviations	Allen-Bradley Industrial Automation Glossary	AG-7.1

Overview of Safety Controllers

Using This Chapter

Safety Concept

GuardPLC controllers feature a fail-safe CPU according to IEC 61508 (SIL3) and EN954-1 (Cat.4). Faults that cause loss of safety function are detected within the safety time you specify. Faults that cause loss of safety function only in combination with another fault, are detected at least within the multiple error occurrence time (24 hours).

This results in the following requirements for the safety concept:

- You specify the safety time and the watchdog time. The multiple error occurrence time is preset to 24 hours.
- Even upon the detection of an error, the controller continues to react in a safety-related way.
- Faulty input signals (e.g. incorrectly transmitted input values) do not affect the safe function of the controller. Faulted input signals have a “0” value.
- An error in a non-safety-related module does not affect the safety of the controller.
- The failure of the controller has no effect on the safety of other safety-related modules.

For more information on the safety concept, see the *GuardPLC Controllers Safety Reference Manual*, publication 1755-RM001.

Response to Faults

Type of I/O error:	Controller behavior:
permanent	If an error occurs at an I/O point, only this I/O point is considered faulty and not the entire module. In case of faulty input points, "0" is assumed to be the safe value. Faulty output channels are de-energized. If it is not possible to de-energize a single point, the entire module is considered to be faulty, the entire module is de-energized, and the corresponding error status is set. The controller reports the error to the user program. If the entire module cannot be de-energized, the controller goes to FAILURE_STOP.
transient	A transient error is an error that occurs in an I/O module and then disappears by itself. If a transient error occurs, the module performs a self test. If the test is successful, the status of the I/O module is set to "good" and the module's normal function continues. In the process, the GuardPLC controller performs a statistical evaluation of the frequency of errors. The I/O module is permanently set to "faulty" if the pre-set error frequency is exceeded. In this case, the module does not resume its normal function after the error has disappeared. To resume normal function, you must cycle power or change the controller to STOP and then RUN. If an error persists for a period of time exceeding that of the multiple error occurrence time (24 hours), the I/O module is permanently set to "faulty" and does not continue normal function after the disappearance of the error. The I/O module can only resume normal function after you cycle power or STOP/START the controller. For faulty modules, the controller uses safe values ("0", LOW).
controller	Upon the detection of an error, the controller goes to FAILURE_STOP and all output channels are set to the safe state (value = "0"). In some cases in which a FAILURE_STOP occurs, a power cycle will not enable normal operation. A manual reset from STOP to RUN, using RSLogix Guard PLUS, is required. CAT 4 faults typically require manual resets. An error in the user program is not considered an error of the controller.

The controller also monitors the timing and consistency of the:

- hardware self-tests and software self-tests of the controller
- cycle of the user program
- processing of the I/O signals including I/O tests
- RUN cycle of the controller
- transition from RUN to STOP

Safe States

Inputs

The safe state of an input is indicated by a 0 signal being passed to the user program logic. When a fault occurs, the inputs are switched off (0).

Outputs

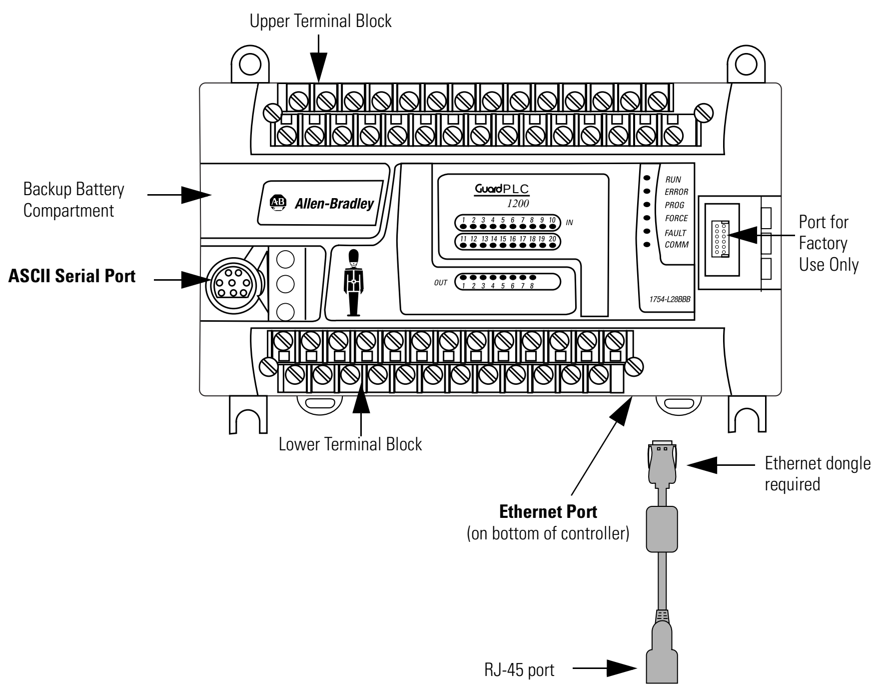
An output is in the safe state when it is de-energized. In the event of a fault, all outputs are switched off. This includes faults in Ethernet communication.

Hardware Overview

GuardPLC 1200 System

The GuardPLC 1200 is a compact system consisting of a CPU, watchdog, and on-board digital I/O. The GuardPLC 1200 features 20 digital inputs, 8 digital outputs, and 2 high-speed counters. An RS-232 serial port supports ASCII communications and an Ethernet port provides safety-related communication. A user-supplied 24V dc power supply is required. See page 2-2 for power supply requirements and page 3-15 for power supply connections.

Figure 1.1 GuardPLC 1200 Controller



GuardPLC 1600 and GuardPLC 1800 System

Figure 1.2 GuardPLC 1600 Controller

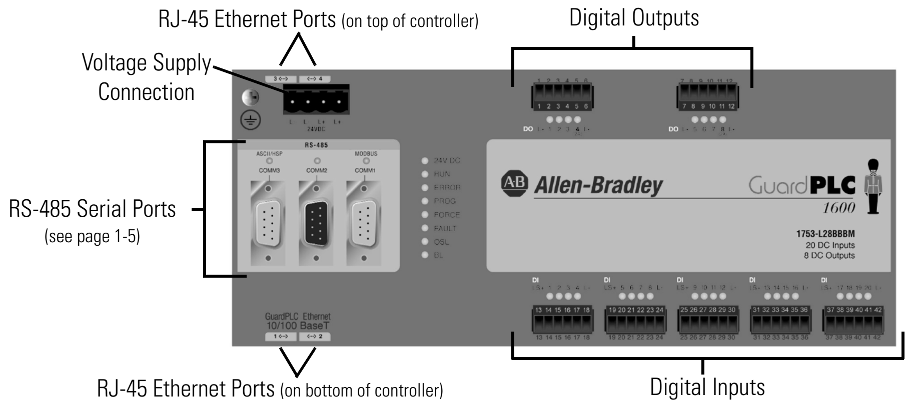
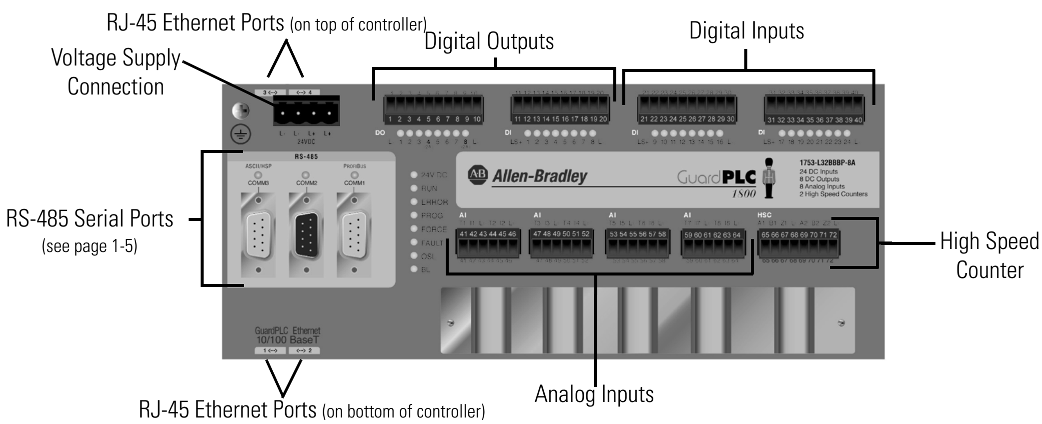


Figure 1.3 GuardPLC 1800 Controller



The GuardPLC 1600 system features 20 digital inputs, 8 digital outputs with the addition of optional distributed Safety I/O. The GuardPLC 1800 system features 24 digital inputs, 8 digital outputs, 8 safety-related analog outputs, and 2 high speed counters, as well as optional distributed Safety I/O. The status of inputs and outputs is indicated via LEDs. A user-supplied 24Vdc power supply is required. See page 3-15 for information on power supply requirements.

Each controller features four 10/100BaseT, RJ-45 connectors to provide safety-related communications via GuardPLC Ethernet to distributed I/O and other GuardPLC controllers, OLE for Process Control (OPC) servers[1], and with RSLogix Guard PLUS programming software. The four connectors and the controller are connected via an internal Ethernet switch.

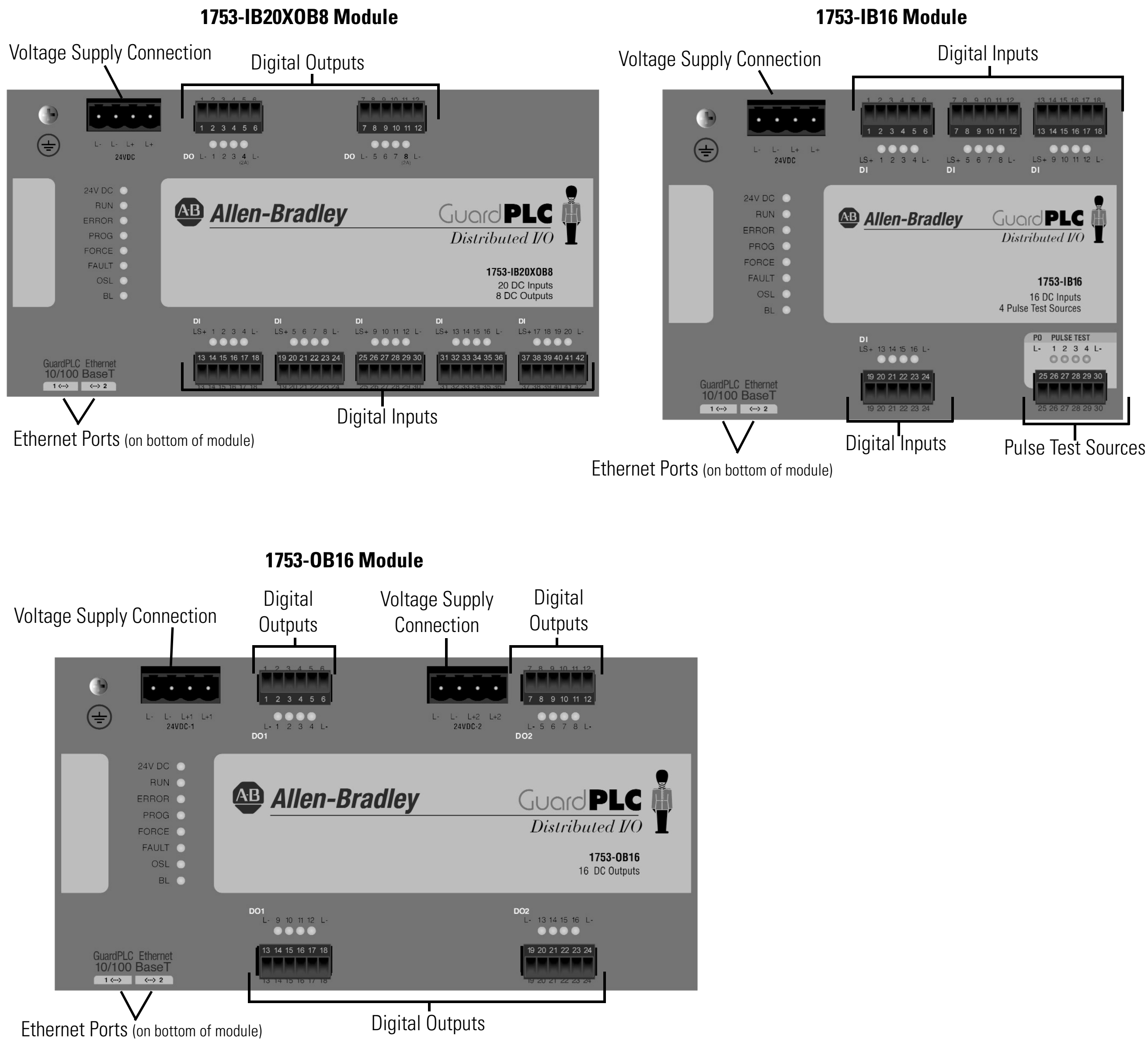
(1) The OPC server is not suitable for safety-related communications.

Three ports are located on the front of the controller, providing the following non-safety-related communication options:

Serial Port Designation	Function
COMM1 (RS-485)	Modbus RTU Slave (1753-L28BBBM or 1753-L32BBBM-8A) Profibus-DP-Slave (1753-L28BBBP or 1753-L32BBBP-8A) Read/Write
COMM2	not used
COMM3 (RS-485)	GuardPLC ASCII Protocol (Read-only)

Distributed I/O for GuardPLC

Figure 1.4 GuardPLC Bulletin 1753 I/O



Three modules are available for use with the GuardPLC 1600, GuardPLC 1800, series C GuardPLC 1200 controllers and series C GuardPLC 2000 CPUs. Module status is indicated via LEDs.

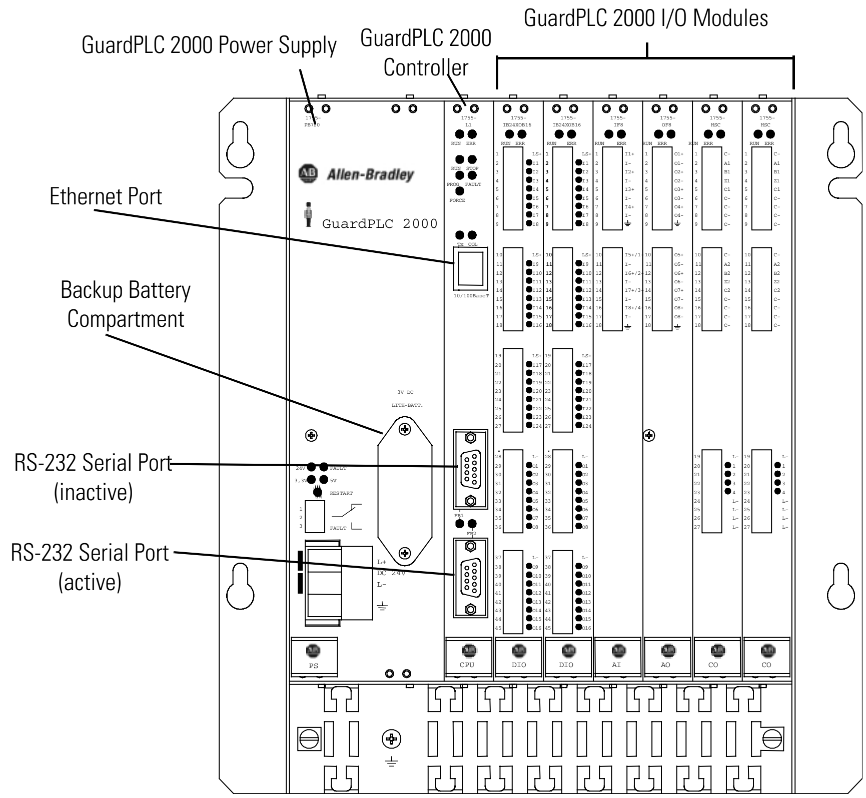
Catalog Number	Description	Inputs	Outputs
1753-IB16	Input Module	16 digital (not isolated) 4 pulse test sources	NA
1753-OB16	Output Module	NA	16 digital (not isolated)
1753-IB20XOB8	Input/Output Module	20 digital (not isolated)	8 digital (not isolated)

GuardPLC 2000 System

The GuardPLC 2000 is a modular system consisting of a controller (1755-L1), which provides central CPU and communications functions, and a separate power supply and I/O residing in a GuardPLC 1755-A6 chassis. A maximum of six I/O modules may be used in a single system.

The GuardPLC 2000 controller has one active RS-232 serial port for non-safety related communications. It also features an Ethernet port for configuration and safety-related communications. The lower DB9 port supports RS-232 ASCII (read-only) communications; the upper port is inactive.

Figure 1.5 GuardPLC 2000 Controller, Power Supply, and I/O Modules



GuardPLC 2000 Power Supply

The 1755-PB720 power supply module provides two voltages (3.3V dc and 5V dc) for the GuardPLC 2000. They are electrically isolated from the supply voltage, 24V dc.

1755-IB24XOB16 I/O Module

The 1755-IB24XOB16 digital input/output module provides 24 digital inputs and 16 digital outputs. The status of each I/O signal is displayed with an LED located on the right side of the front plate connectors. Inputs and outputs are electrically isolated from the supply voltage, 24V dc.

1755-IF8 Analog Input Module

The 1755-IF8 analog input module has eight inputs. These inputs can be used as either eight single-ended inputs or four differential analog inputs which are electrically isolated from the logic side of the GuardPLC. The measured input value can be either voltage or current. If you use the input module for current, you need a shunt resistor. The measured value is digitally transferred to the processor system as a value between 0 and 2000.

1755-OF8 Analog Output Module

The 1755-OF8 analog output module provides eight outputs, galvanically isolated in groups of 2 (i.e. 2 outputs per power supply). They are electrically isolated from the processor system. Each analog output can operate as a current source or a voltage source.

1755-HSC High Speed Counter Module

The 1755-HSC counter module provides two counters and four digital outputs. They are electrically isolated from the processor system. The status of the four output signals is displayed with LEDs located at the right side of the front plate output connector.

Communication Capabilities

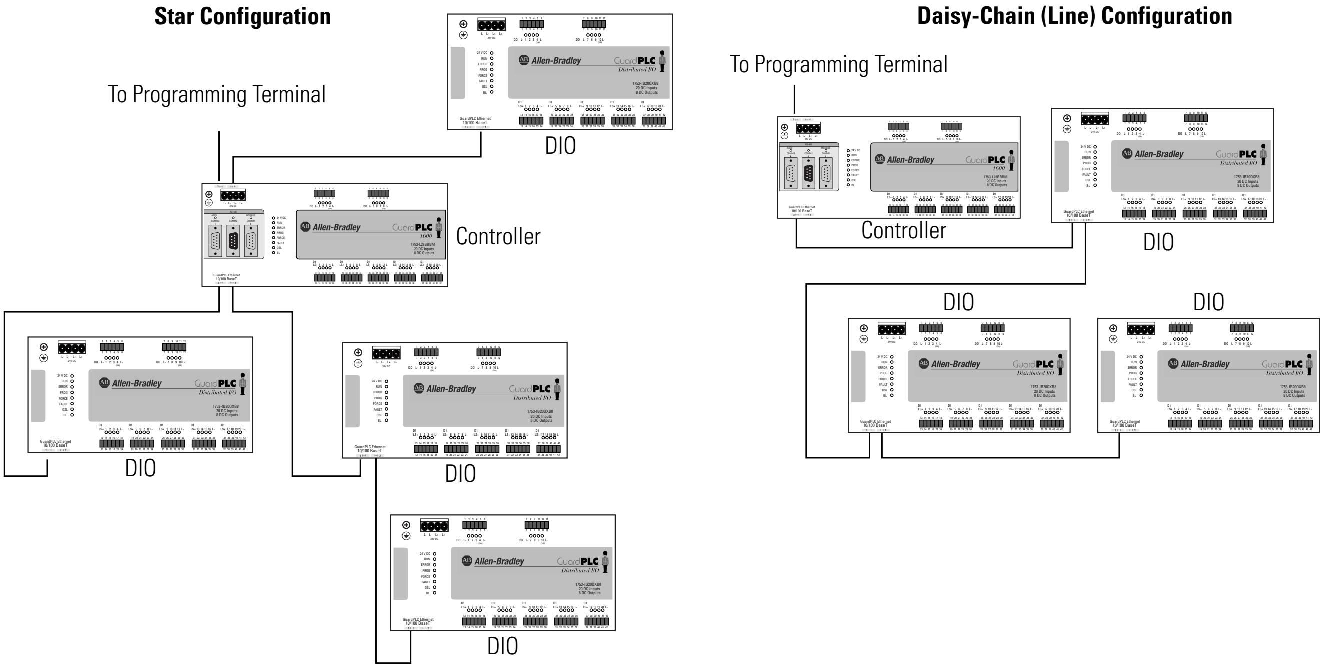
GuardPLC Ethernet

GuardPLC Ethernet provides safe communication via Ethernet for distributed I/O and peer-to-peer communications for all GuardPLC controllers. It also provides non-safety-related communication with the OPC server. Programming and configuration of controllers is accomplished via GuardPLC Ethernet.

Various GuardPLC systems can be networked together via GuardPLC Ethernet, using star or daisy-chain configurations. A programming device running RSLogix Guard PLUS can also be connected wherever required.

IMPORTANT Make sure that a network loop is not generated. Data packets must only be able to reach a node via a single path.

Figure 1.6 GuardPLC Ethernet Networking Example



ASCII

This read-only, non-safety-related protocol can be used to extract diagnostic and status information from the GuardPLC controllers. ASCII protocol is available over the RS-232 port on the GuardPLC 1200 and GuardPLC 2000 controllers and via the RS-485 Comm 3 port on GuardPLC 1600 and GuardPLC 1800 controllers.

See Chapter 14 for details on communication with ASCII devices.

Modbus RTU Slave

Modbus is a standard industrial non-safety-related serial protocol in which the Modbus master can communicate with a maximum of 255 slave devices. The Modbus master initiates and controls all communications on the network.

Modbus RTU Slave protocol is available via the RS-485 Comm 1 port on GuardPLC 1600 and GuardPLC 1800 controllers with catalog numbers ending in “M”.

Modbus RTU Slave protocol allows both the reading and writing of data.

For more information on the Modbus RTU Slave protocol, see the Modbus Protocol Specifications, available from **www.modicon.com/techpubs/**.

Profibus DP Slave

Profibus DP is a non-safety-related serial protocol, designed for high-speed data transmission between automation systems and distributed peripherals.

Profibus DP slave is available via the RS-485 Comm 1 port on GuardPLC 1600 and GuardPLC 1800 controllers with catalog numbers ending in “P”.

Profibus DP Slave protocol allows both the reading and writing of data.

OPC Server

The GuardPLC 1600, GuardPLC 1800, and series C GuardPLC 1200 and 2000 are OPC clients. An OPC server, catalog number 1753-OPC, is available from Rockwell Automation and allows PC applications to read and write data to and from the GuardPLC (non-safety-related communications only).

Installation

Using This Chapter

For information about:	See page
European Communities (EC) Directive Compliance	2-1
mounting	2-2
communication connections	2-9
reset pushbutton	2-13

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:

- EN 50081-2 EMC — Generic Emission Standard, Part 2 — Industrial Environment
- EN 50082-2 EMC — Generic Immunity Standard, Part 2 — Industrial Environment
- EN 61131-2 — Programmable Controllers, Part 2: Equipment Requirements and Tests
- EN 61000-6.2 EMC — Part 6-2: Generic Standards — Immunity for Industrial Environments

This product is intended for use in an industrial environment.

Low Voltage Directive

The power supply of the GuardPLC controllers must meet Council Directive 73/23/EEC Low Voltage, by applying the requirements of EN 61131-2 Programmable Controllers, Part 2 – Equipment Requirements and Tests, as well as one of the following:

- EN 60950 — SELV (Safety Extra Low Voltage)
- EN 60204 — PELV (Protective Extra Low Voltage)

General Safety

Open style devices must be provided with environmental and safety protection by proper mounting in enclosures designed for specific application conditions. See NEMA Standards publication 250 and IEC publication 60529, as applicable, for explanations of the degrees of protection provided by different types of enclosure.

ATTENTION

Consider the following before installing your GuardPLC 1200/1600/1800 controller or distributed I/O:

These products are grounded through the DIN rail. Use zinc-plated yellow-chromate steel DIN rails to assure proper grounding. The use of other DIN rail materials (e.g. aluminum, plastic, etc.) that can corrode, oxidize, or are poor conductors, can result in improper or intermittent grounding.

Mounting

GuardPLC 1200

The GuardPLC 1200 can be either snapped onto a DIN rail or mounted to a back panel using bolts. DIN rail mounting is the easiest way to attach the controller and should be used wherever possible.

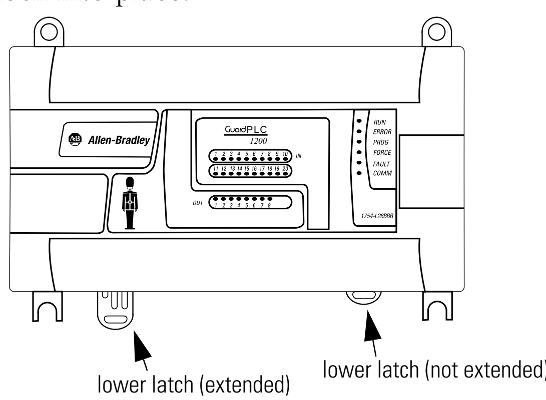
IMPORTANT For cooling reasons:

- The GuardPLC 1200 must be mounted horizontally with the Ethernet socket facing down.
- Select a location where air flows freely or use an additional cooling fan.
- The minimum clearance around the GuardPLC 1200 must be at least 100 mm.
- Do not mount the GuardPLC 1200 over a heating device.

DIN Rail Mounting

1. Hook the two top latches, on the back of the GuardPLC 1200, over the top of the DIN rail.

2. If the lower latches are extended (see figure below), push them up until they lock into place. If the lower latches are not extended, press the GuardPLC 1200 into the DIN rail until they lock into place.



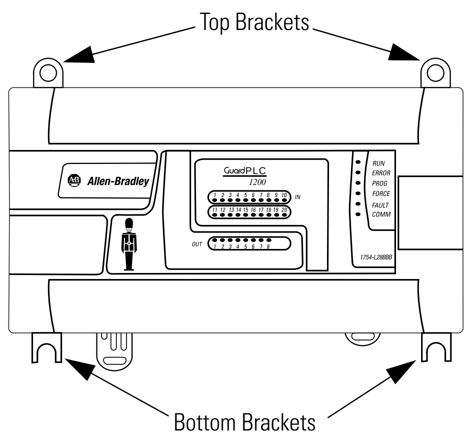
TIP If you need to remove the controller from the DIN rail, use a screwdriver to pull down the lower latches, then lift the controller toward you.

Back Panel Mounting

ATTENTION

Do not bend the controller. Bending the controller will damage it.

Use the four brackets on the GuardPLC 1200 to mount it onto a back panel.



Use the following to mount the controller:

Top Brackets	Bottom Brackets
M4 screws (2)	M5 screws (2)
lock washer	lock washer
washers	washers
nut	nut

If the mounting brackets are not flat before the nuts are tightened, use additional washers as shims, so the controller does not bend when you tighten the nuts.

GuardPLC 1600, GuardPLC 1800, and Distributed I/O

IMPORTANT

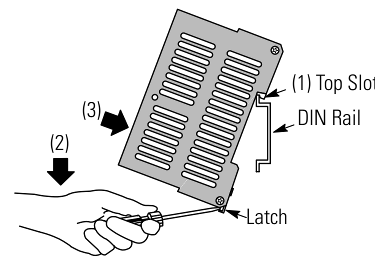
For effective cooling:

- Mount the device horizontally.
- Provide a gap of at least 100 mm (3.94 in.) above and below the device and at least 20 mm (0.79 in.) horizontally between devices.
- Wire duct can run in the 100 mm (3.94 in.) of free space above and below the controller if it is no deeper than 40 mm (1.58 in.). If the depth is greater than 40 mm (1.58 in.), the devices must be placed on stand-offs that match the depth of the duct. If stand-offs are not used, you must provide a gap of at least 80 mm (3.15 in.) between the device and the duct.
- Select a location where air flows freely or use an additional fan.
- Do not mount the controller or I/O module over a heating device.
- Do not block the ventilation slots on the side of the device.

GuardPLC 1600/1800 controllers and I/O cannot be panel-mounted. Mount the GuardPLC 1600/1800 controllers and distributed I/O to a DIN rail by following the steps below.

1. Hook the top slot over the DIN rail.

2. Insert a flathead screwdriver into the gap between the housing and the latch and pull the latch downward.



3. Hold the latch down as you push the housing back onto the DIN rail.

4. Release the latch to lock the device onto the rail.

TIP To remove the device from the DIN rail, insert a flathead screwdriver into the gap between the housing and the latch and pull the latch downward as you lift the device off of the rail.

GuardPLC 2000 Chassis

The GuardPLC 2000 provides two flanges with eyelets. Refer to the illustration below. Use bolts to mount the system to a back panel.

To mount the chassis flanges, you will need four M8 size bolts with lock washer, washer and nut with 13 mm (max.) head diameter. The bolts must be long enough to accept the chassis at its mounting place.

ATTENTION

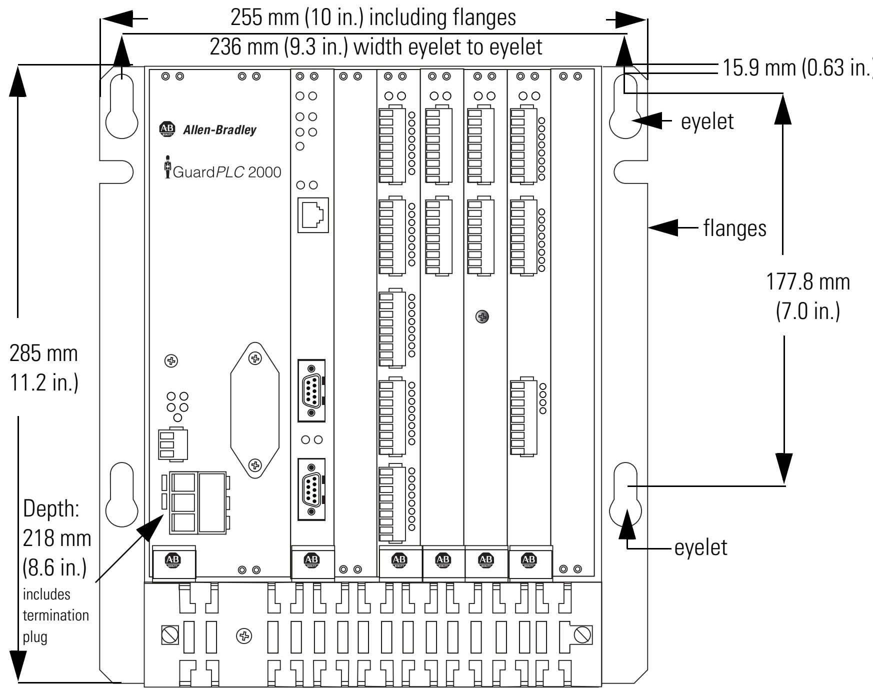
- Do not bend the chassis. Bending will damage the chassis and/or the backplane inside the GuardPLC 2000.

- If the rear side of the chassis does not lie flat before the nuts are tightened, use additional washers as shims so that the chassis does not bend when you tighten the nuts.

IMPORTANT

- The chassis must be installed without any modules inserted.
- Disconnect the supply voltage before mounting the chassis.
- The chassis must be vertically mounted with the cooling fans on the lower side.
- Do not obstruct ventilation openings.
- Provide a gap of at least 100 mm (3.94 in.) above and below the device and at least 20 mm (0.79 in.) horizontally between devices.

Modules are shown for illustration only. The chassis must be installed without any modules inserted.

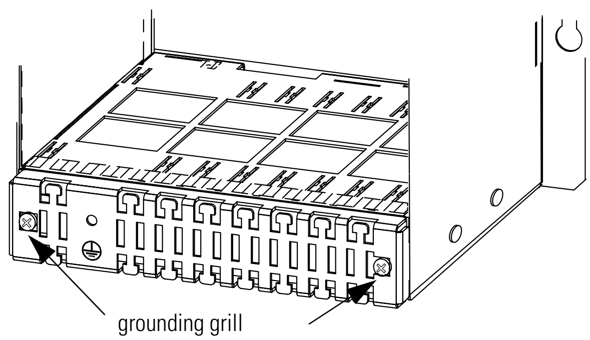


GuardPLC 2000 Controller, I/O, and Power Supply

Mount the GuardPLC 2000 chassis prior to installing the controller, I/O, and power supply modules.

IMPORTANT	Disconnect the power supply module, 1755-PB720, from the 24V dc supply voltage before you insert the module.

1. Before you insert the module, you must detach the grounding grill. To do this, remove the grounding grill screws (see figure below).



2. Remove the lower panel of the chassis and disconnect the fans.

3. **Power Supply:** Insert the power supply into the left-most slot of the chassis.

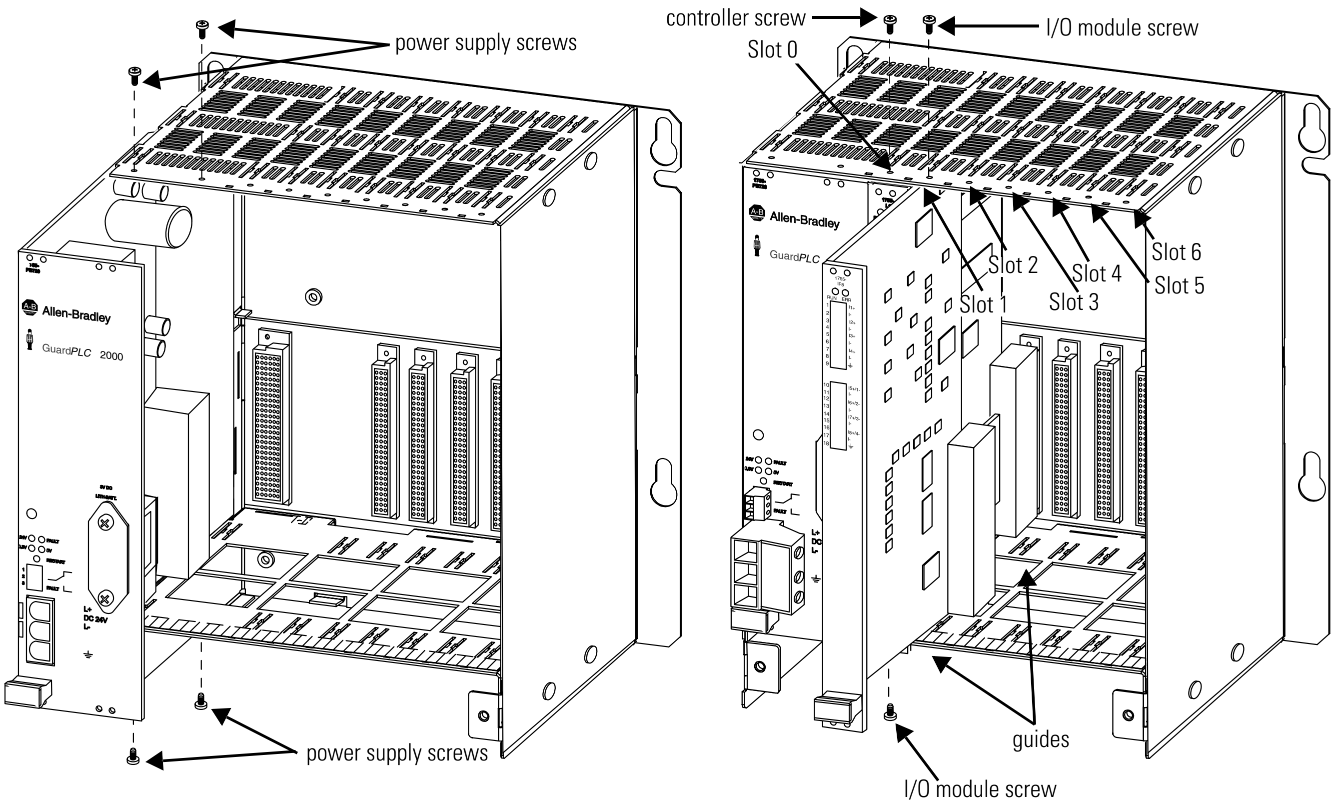
   Controller: Insert the controller into the slot directly to the right of the power supply module (slot 0).

   I/O Module: Insert the module into any unused slot from 1 to 6 (see figure on page 2-8). Keep the module in line with the guides so the module runs smoothly in the track.

4. Begin pushing the module into the chassis. If there is resistance when you push the module into the backplane, do not force the module because the pins will bend. Remove the module and start again at step 3.

5. Continue pushing the module into the chassis until the front of the module is flush with the other modules in the chassis.

6. Secure the module with the module screws on the top and bottom of the module (see figure on page 2-8).



TIP If you are installing other GuardPLC 2000 modules, follow their Installation Instructions up to this point before you complete the next 3 steps.

7. Reconnect the fans.
8. Replace the lower panel of the chassis, sliding it over the tabs on the sides of the chassis and under the tabs on the back of the chassis.
9. Use the grounding grill screws to attach the grounding grill.

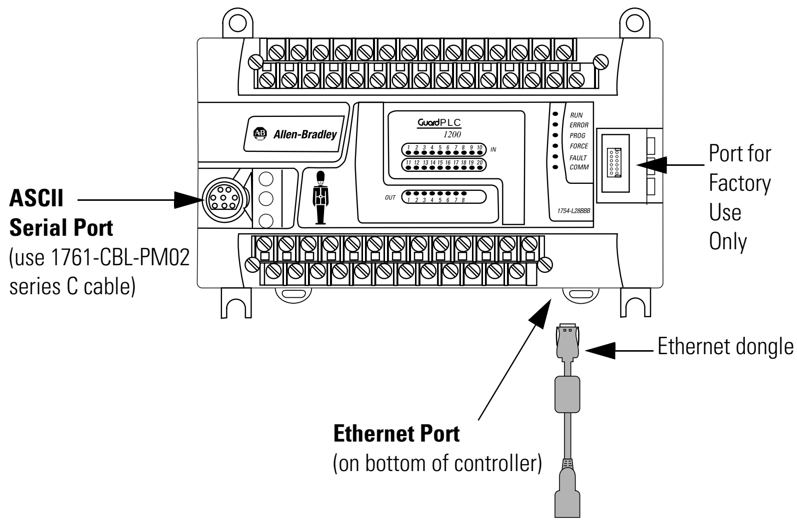
Communication Connections

GuardPLC 1200

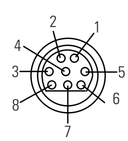
The GuardPLC 1200 has an ASCII serial port for non-safety-related communications and an Ethernet port for safety-related communications.

Connect the ASCII port to any RS-232 device that has the capability to send ASCII command strings to the controller. The controller replies with a data variable string. See Chapter 14 for more information on ASCII communications

Use the following illustration to connect the ASCII and Ethernet ports.



The pin assignment of the ASCII Serial port is shown below.



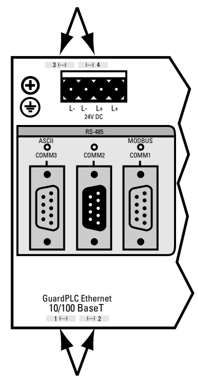
Pin	Function
1	24V dc
2	ground (GND)
3	request to send (RTS)
4	received data (RxD)
5	received line signal detector (DCD)
6	clear to send (CTS)
7	transmitted data (TxD)
8	ground (GND)
9	not applicable

GuardPLC 1600 and GuardPLC 1800

Connections for Safety-Related Communication

The controller has four 10/100BaseT, RJ-45 connectors to provide communications via GuardPLC Ethernet to other controllers, distributed I/O or RSLogix Guard PLUS. Connectors 1 and 2 are located on the bottom side on the left. Connectors 3 and 4 are located on the top side on the left.

Ethernet Ports 3 and 4



Ethernet Ports 1 and 2

All four connectors and the GuardPLC processor are connected together by an internal Ethernet switch. In contrast to a hub, a switch is able to store data packets for a short period of time in order to establish a temporary connection between two communication partners for the transfer of data. In this way, collisions (typical of a hub) can be avoided and the load on the network is reduced.

The switch automatically switches between transfer rates of 10 and 100 Mbit/s and between full- and half-duplex connections. This makes the full bandwidth available (full-duplex operation) in both directions.

A switch enables several connections to be established at the same time and can address up to 1000 absolute MAC addresses.

Auto-crossing recognizes that cables with crossed wires have been connected and the switch adjusts accordingly. Therefore, either cross-over or straight-through Ethernet cabling can be used.

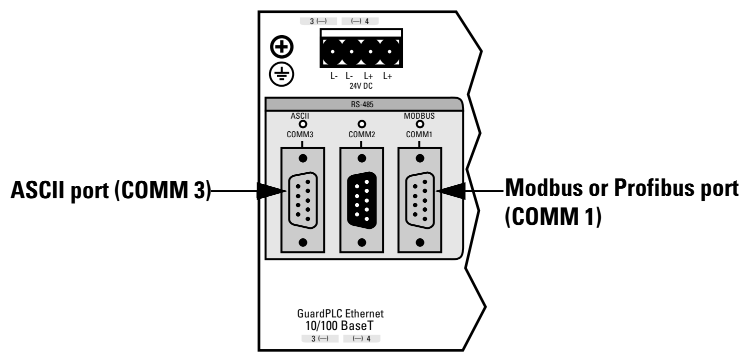
Star or line configurations are available. Make sure that a network loop is not generated. Data packets must only be able to reach a node via a single path.

See Chapters 4 and 12 for information on programming via Ethernet and on peer-to-peer communications.

Connections for Non-Safety-Related Communications

Three 9-pin Min-D connectors are located on the front of the controller, providing the following communications:

Designation	Function
COMM1 (RS-485)	Modbus RTU Slave (1753-L28BBBM or 1753-L32BBBM-8A) Profibus-DP-Slave 1753-L28BBBP or 1753-L32BBBP-8A)
COMM2	not used
COMM3	GuardPLC ASCII Protocol



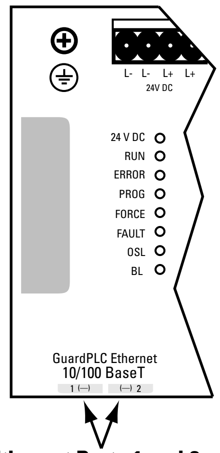
IMPORTANT The three Min-D connectors are RS-485. You must use an electrical interface device to connect the controller to an RS-232 device.

The pin assignment of the Min-D connectors is shown in the table below.

Connection	Signal	Function
1	---	---
2	RP	5V, decoupled with diodes
3	RxD/TxD-A	Receive/Transmit data A
4	CNTR-A	Control Signal A
5	DGND	Data reference potential
6	VP	5V, positive pole of supply voltage
7	---	---
8	RxD/TxD-B	Receive/Transmit data B
9	CNTR-B	Control Signal B

GuardPLC Distributed I/O Modules

Each module has two 10/100BaseT, RJ-45 connectors to provide safety-related communications via GuardPLC Ethernet. These two connectors and the GuardPLC DIO module are connected together by an internal Ethernet switch.

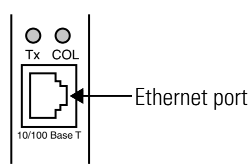


Ethernet Ports 1 and 2

GuardPLC 2000

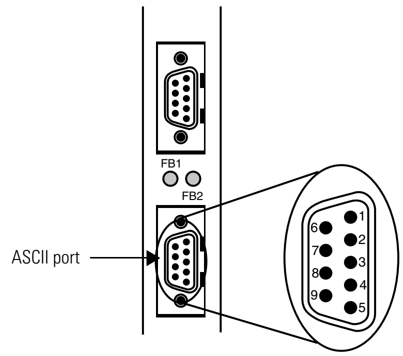
Connections for Safety-Related Communication

To configure/program the GuardPLC system, the controller must be connected on an Ethernet network to the RSLogix Guard PLUS programming terminal. GuardPLC Ethernet also provides for Peer-to-Peer communication to distributed I/O and to other controllers.



Connections for Non-Safety-Related Communication

Connect the ASCII port (FB2) to any RS-232 device that has the capability to send ASCII command strings to the controller. The controller replies with a data variable string. See Chapter 14 for more information on ASCII communications.



pin	function
1	none
2	send data
3	receive data
4	none
5	ground
6	none
7	RTS
8	CTS
9	none

Reset Pushbutton

GuardPLC 1600 and 1800 controllers and distributed I/O are equipped with a reset pushbutton. Reset via the pushbutton is necessary:

- if you forget the password to go online via the programming software, or
- if you are unable to determine the IP address and SRS of the controller

The pushbutton is accessible through a small round hole at the top of the housing, approximately 4 to 5 cm (1.6 to 2.0 in.) from the left rim and recessed approximately 9.5 mm (0.375 in.).

IMPORTANT	Activate the reset pushbutton using an insulated pin to prevent short-circuits.

To reset, press and hold the pushbutton while rebooting the controller by cycling power. Hold the reset pushbutton until the PROG LED stops flashing. Pressing the Reset pushbutton during operation has no affect.

After a reset, the IP address, SRS and login accounts are temporarily reset to their default settings:

- IP = 192.168.0.99
- SRS = 60000.1
- login Username = Administrator
- login Password = [none]

At the next power cycle, these settings will be reset to the last values stored into Flash. This means that either:

- the settings prior to the reset will be restored, **or**
- if any settings were changed after the reset, these new settings will still be in effect.

Wiring

Using This Chapter

IMPORTANT

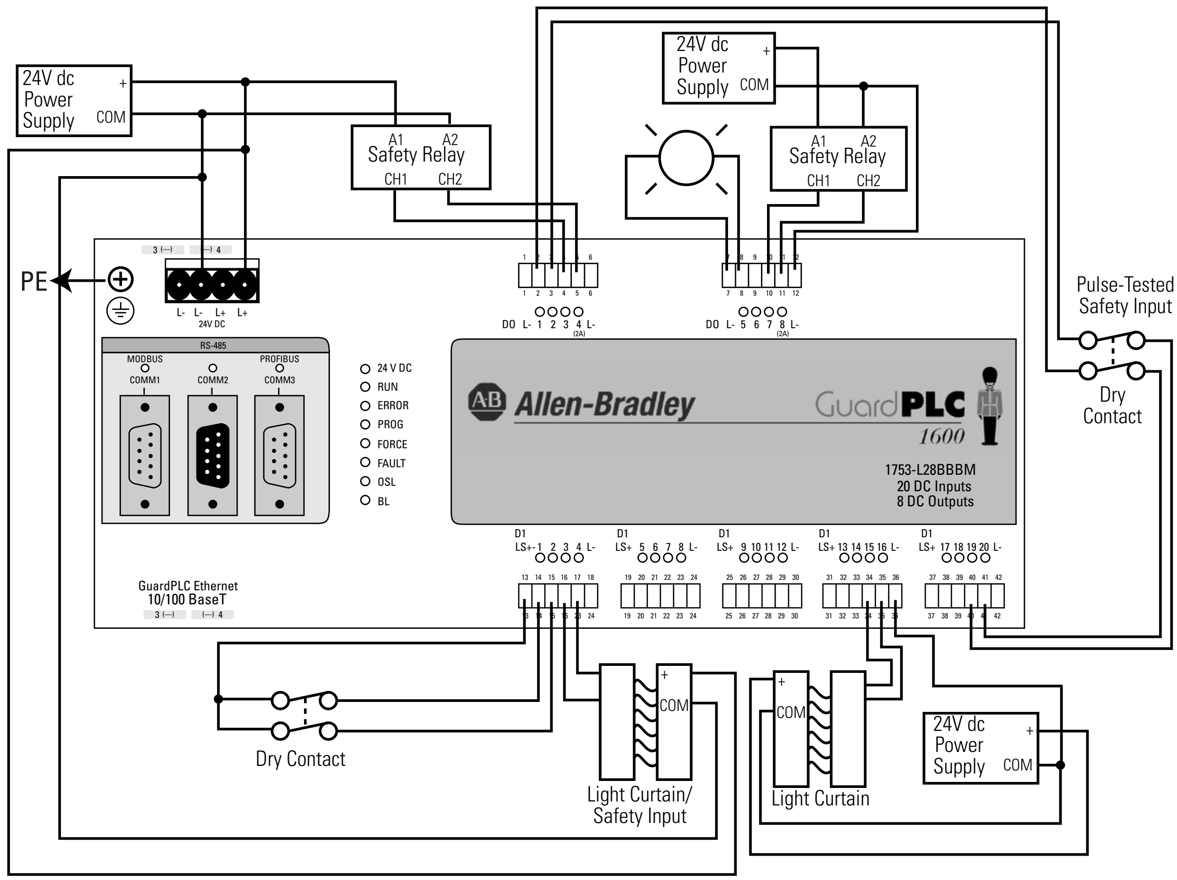
The wiring diagrams in this chapter detail only the wiring necessary to sense/control the I/O devices.

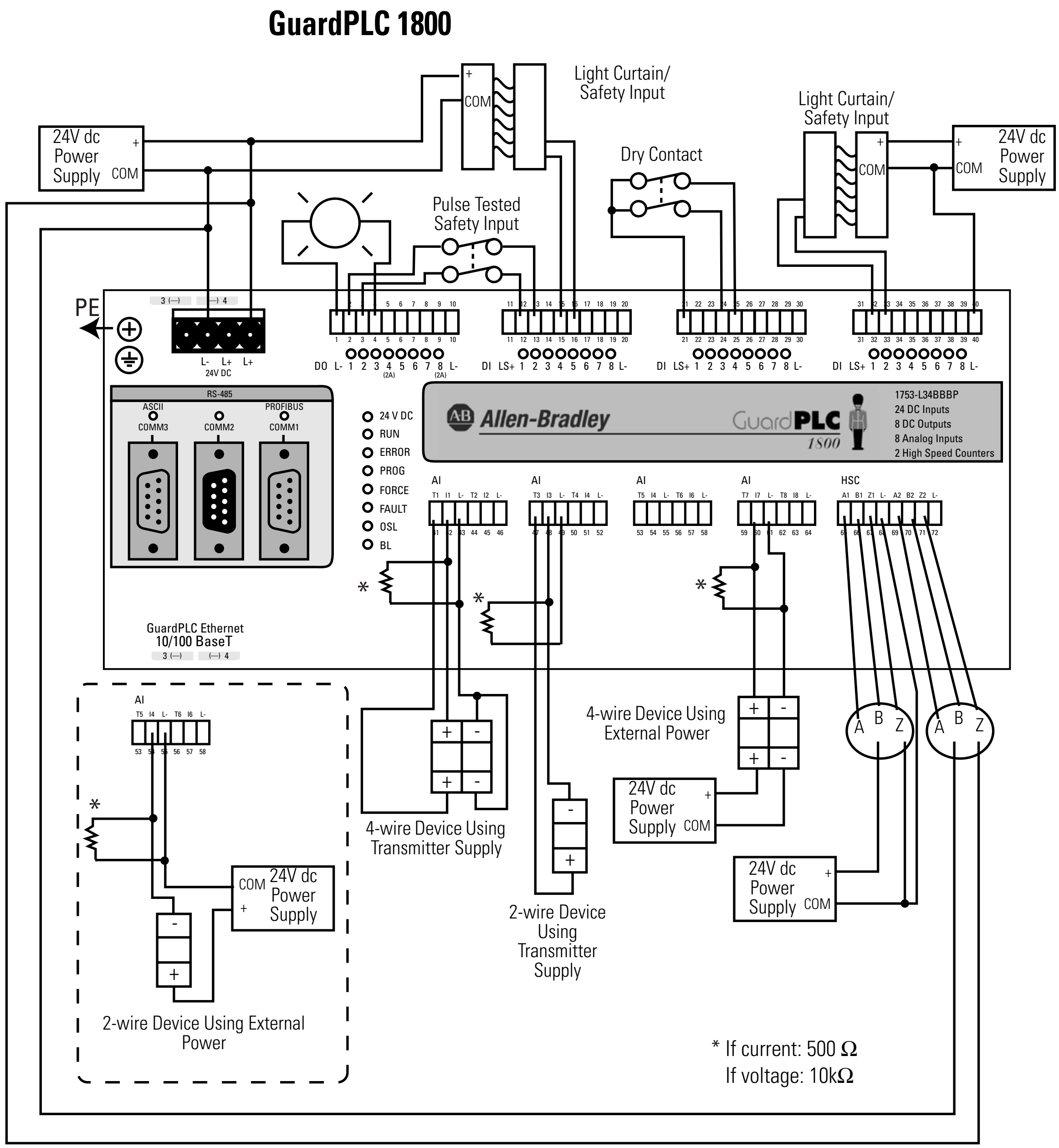
They do not show all of the wiring necessary to achieve CAT. 3 or CAT. 4 safety circuits.

For example, monitoring feedback signals is not illustrated.

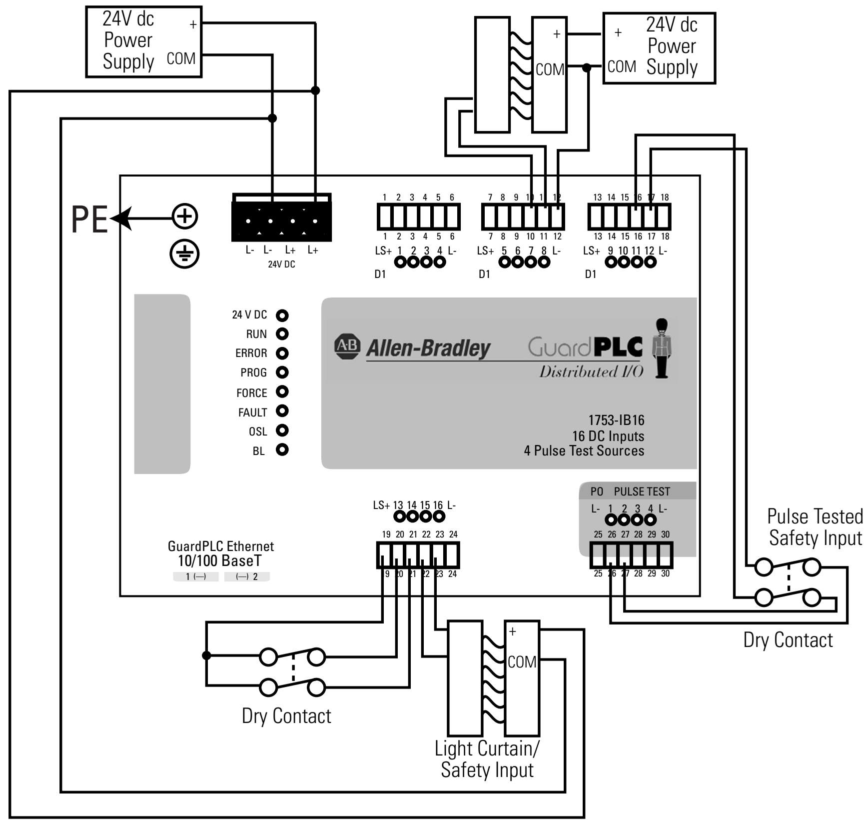
GuardPLC Wiring Examples

GuardPLC 1600

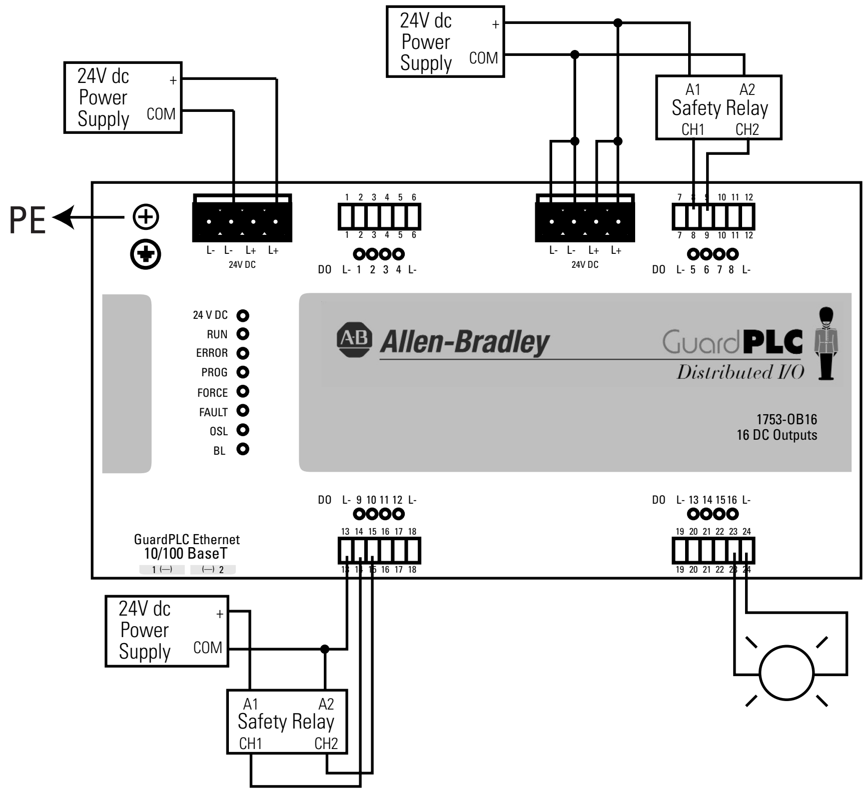



GuardPLC 1800
Light Curtain/ Safety Input
24V dc Power Supply
COM
Dry Contact
Pulse Tested Safety Input
PE
24 V DC
RUN
ERROR
PROG
FORCE
FAULT
OSL
BL
Allen-Bradley
GuardPLC 1800
1753-L34BBBP
24 DC Inputs
8 DC Outputs
8 Analog Inputs
2 High Speed Counters
RS-485
ASCII
PROFIBUS
COMM3
COMM2
COMM1
GuardPLC Ethernet
10/100 BaseT
AI
HSC
4-wire Device Using Transmitter Supply
2-wire Device Using Transmitter Supply
4-wire Device Using External Power
2-wire Device Using External Power
* If current: 500 Ω
If voltage: 10kΩ

1753-IB16

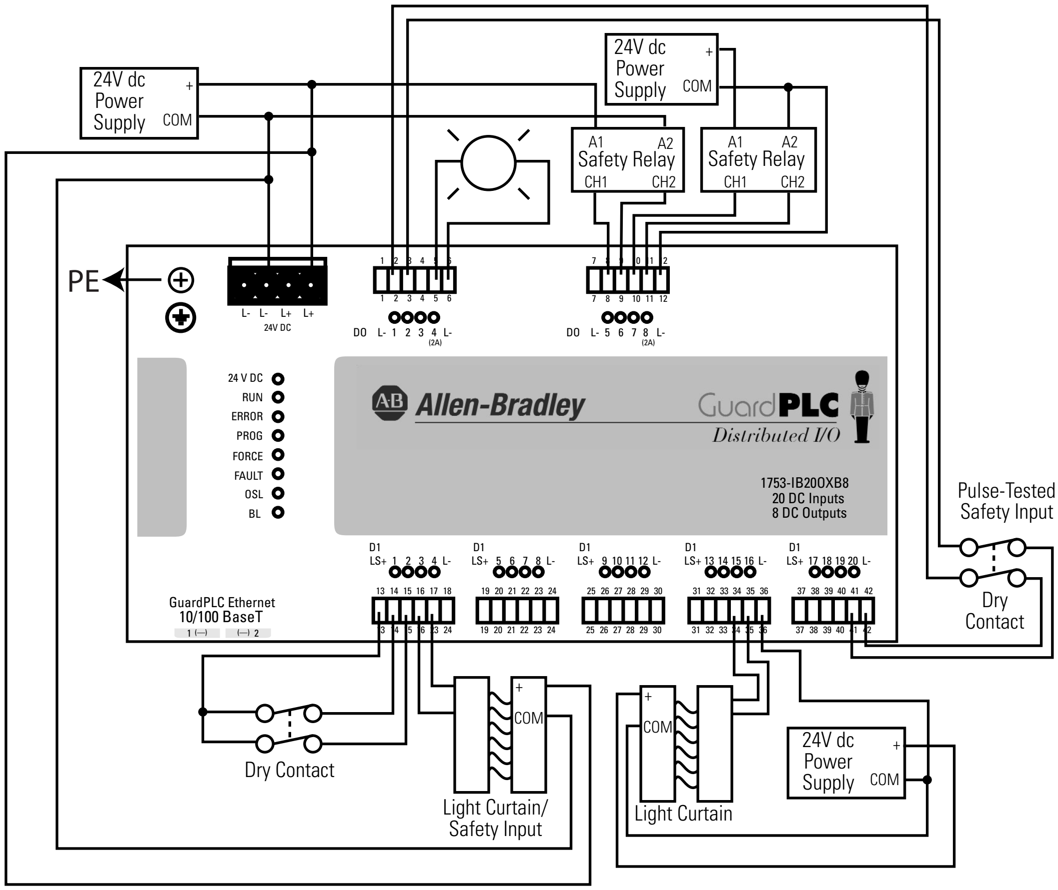

24V dc Power Supply
+
COM
24V dc Power Supply
PE
L- L- L+ L+
24V DC
LS+ 1 2 3 4 L-
LS+ 5 6 7 8 L-
LS+ 9 10 11 12 L-
24 V DC
RUN
ERROR
PROG
FORCE
FAULT
OSL
BL
Allen-Bradley
GuardPLC
Distributed I/O
1753-IB16
16 DC Inputs
4 Pulse Test Sources
PO PULSE TEST
LS+ 13 14 15 16 L-
GuardPLC Ethernet
10/100 BaseT
Pulse Tested Safety Input
Dry Contact
Dry Contact
Light Curtain/ Safety Input

1753-OB16

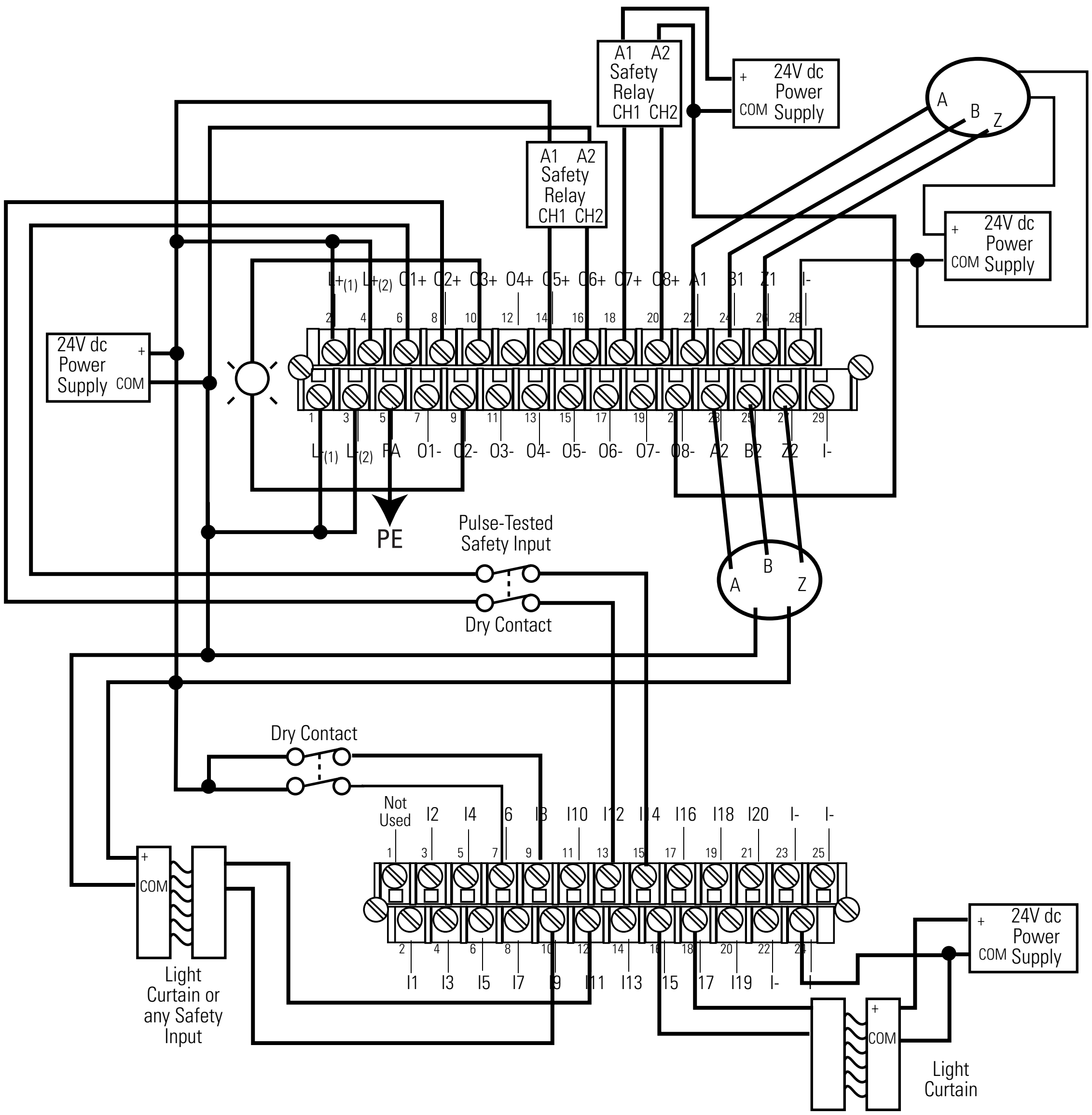


24V dc Power Supply + COM
24V dc Power Supply + COM
A1 A2 Safety Relay CH1 CH2
PE
L- L- L+ L+
24V DC
DO L- 1 2 3 4 L-
DO L- 5 6 7 8 L-
24 V DC
RUN
ERROR
PROG
FORCE
FAULT
OSL
BL
Allen-Bradley
GuardPLC
Distributed I/O
1753-OB16
16 DC Outputs
DO L- 9 10 11 12 L-
DO L- 13 14 15 16 L-
GuardPLC Ethernet
10/100 BaseT
24V dc Power Supply + COM
A1 A2 Safety Relay CH1 CH2

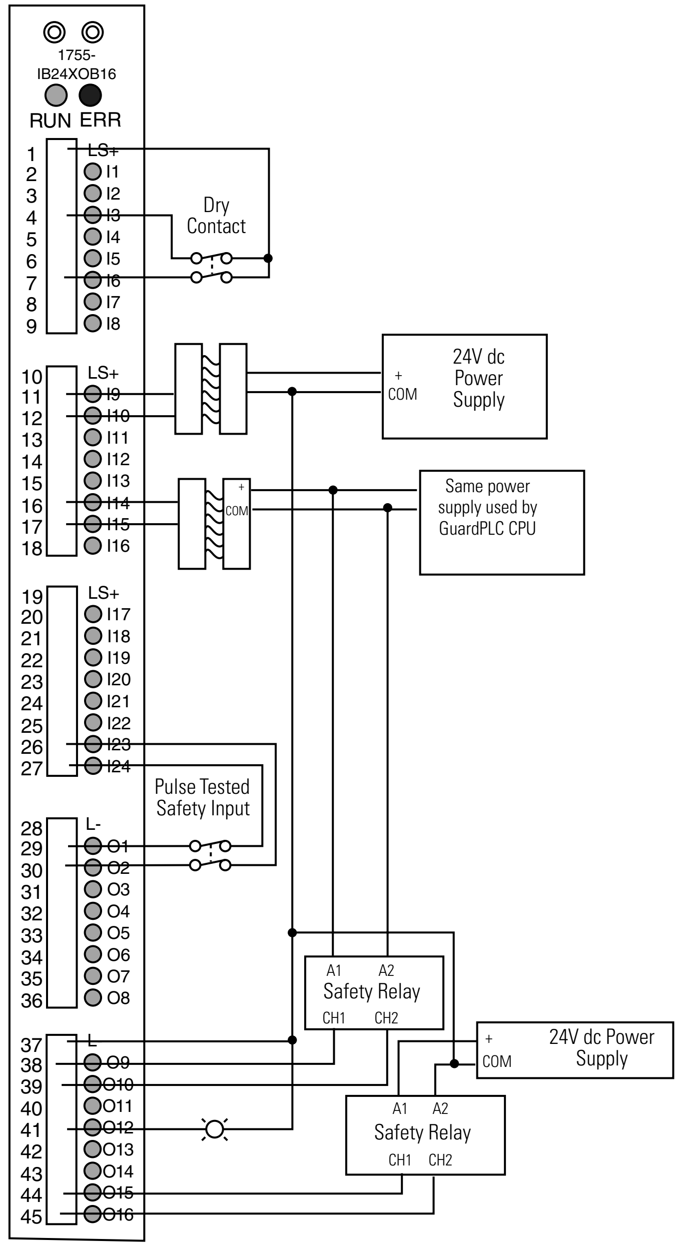
1753-IB20XOB8



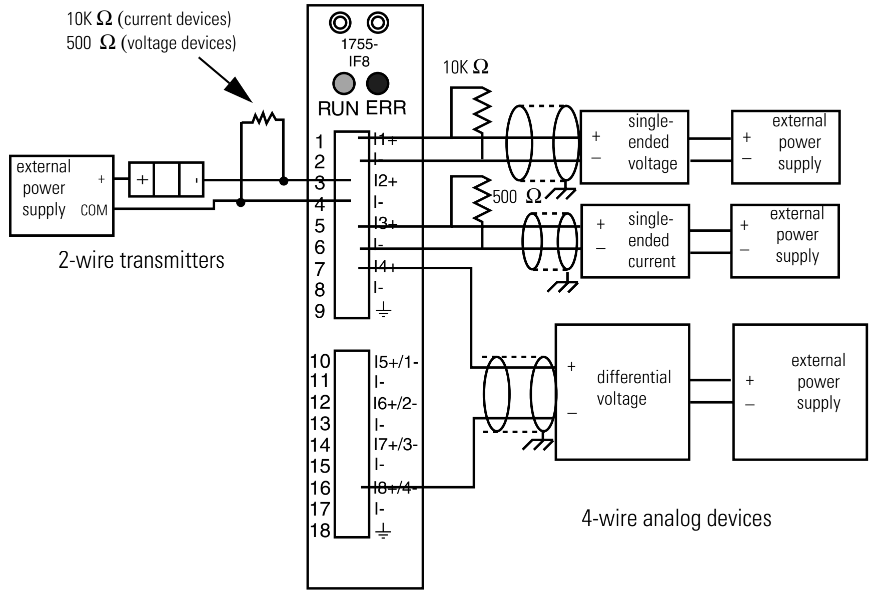
GuardPLC 1200



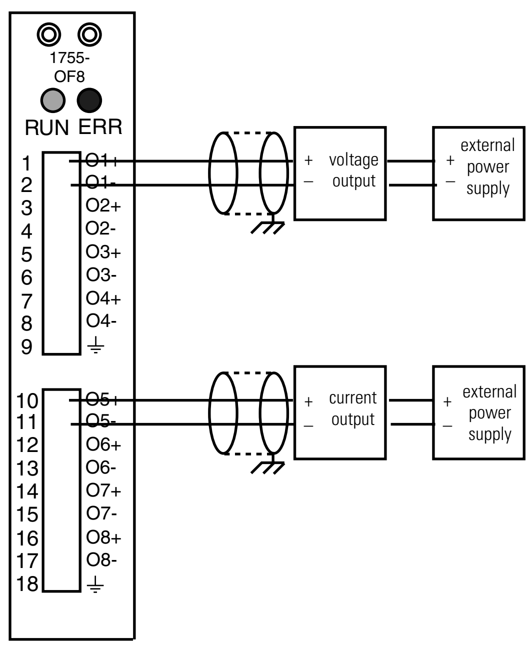
1755-IB24XO16 Digital Input/Output Modules



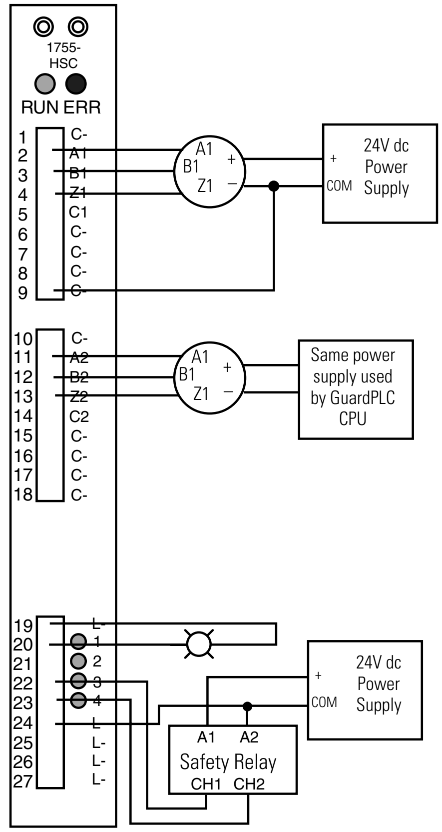
1755-IF8 Analog Input Modules



1755-OF8 Analog Output Modules



1755-HSC High Speed Counter Module



General Wiring Considerations

Terminals accommodate wire sizes up to 1.5 mm^2 (16 AWG) for input/output wiring and up to 2.5 mm^2 (14 AWG) for voltage supply connections.

Safety-Related Digital Inputs

The status of digital inputs is indicated via LEDs when the controller is in RUN mode.

Follow the closed-circuit principle for external wiring when connecting sensors. To create a safe state in the event of a fault, the input signals revert to the de-energized state (0). The external line is not monitored, but a wire break is interpreted as a safe (0) signal.

In general, the LS+ terminals, not L+ on the power supply connection, should be used to supply voltage for safety inputs. Each LS+ features individual short-circuit and EMC protection. Due to current limitations, use LS+ for only the safety inputs on the same terminal plug.

Safety-Related Digital Outputs

The status of digital outputs is indicated via LEDs when the controller is in RUN mode.

GuardPLC outputs are rated to either 0.5A or 1.0A at an ambient temperature of 60°C (140°F). At an ambient temperature of 50°C (122°F), outputs rated at 1.0A increase to 2.0A.

If an overload occurs, the affected outputs are turned off. When the overload is eliminated, the outputs are under the control of the controller and are energized based on the user program code.

An output is in the safe state when it is de-energized. Therefore, outputs are switched off when a fault that affects the safe control of those outputs occurs.

For connection of a load, the reference pole L- of the corresponding channel group must be used (2-pole connection). Although L- poles are connected internally to L- on the power supply input, it is strictly recommended to connect the L- reference poles only to their corresponding output group. EMC testing was performed in this manner.

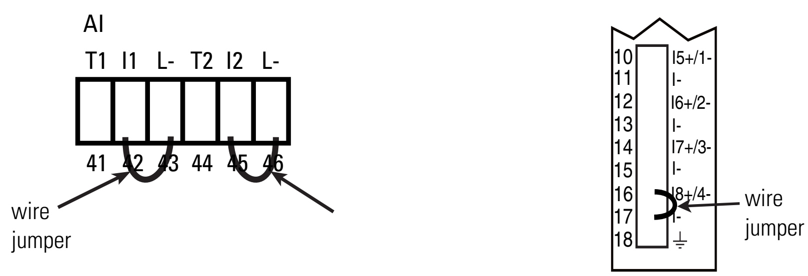
TIP Inductive loads can be connected without a protection diode on the load, because there is a protection diode located within the GuardPLC. However, Rockwell Automation strongly recommends that a protection diode be fitted directly to the load to suppress any interference voltage. A 1N4004 diode is recommended.

Safety-Related Analog Inputs

GuardPLC safety controllers use analog inputs for the unipolar measurement of voltages from 0 to 10V, referenced to L-. A 10 KΩ shunt is used for single-ended voltage signals. With a 500 Ω shunt resistor, currents from 0 to 20 mA can also be measured.

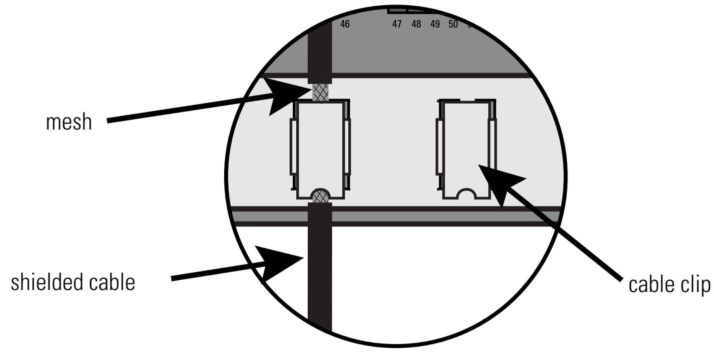
The feeder lines should be no more than 300 m (284 ft.) in length. Use shielded, twisted-pair cables, with the shields connected at one end, for each measurement input.

Unused analog inputs must be short-circuited. Place wire jumpers to ground on any inputs that are not used.



GuardPLC 1800 1755-IF8

Shielded cabling is fed in from below so that the shielding can be connected to the shield contact plate using a clip. Remove about 2 cm of the outer cable insulation so that the mesh is exposed at the point where the cable is clipped to the plate. Position the clip over the uninsulated cable shielding and push it into the slots of the shield contact plate until it fits firmly in place, as shown below.



IMPORTANT Make sure that the mesh comes in direct contact with the shield contact plate. If the mesh does not touch the plate, the cable is not grounded.

High Speed Counters

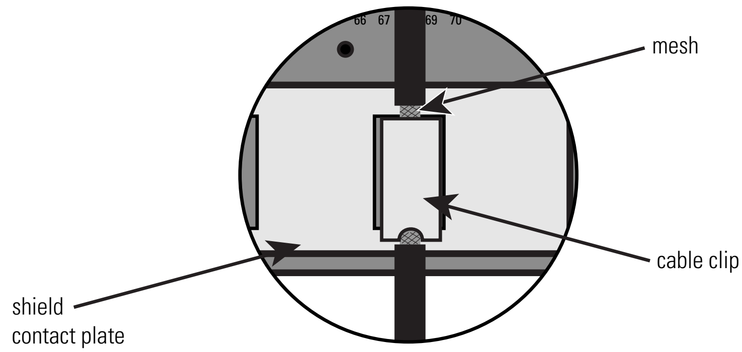
The GuardPLC safety controllers feature inputs for high speed counting up to 1 MHz. These counters are 24-bit, and are configurable for either 5V or 24V dc. The table below shows the maximum input frequency of the counters by controller type.

Maximum Input Frequency	Controller Type
100 KHz	GuardPLC 1200 GuardPLC 1800
1 MHz	GuardPLC 2000 (1755-HSC)

The counters can be used as a counter or as a decoder for 3-bit Gray Code inputs. As a counter, input A is the counter input, input B is the counter direction input, and input Z is used for a reset.

The counter inputs must be connected using shielded, twisted-pair cables for each measurement input. The shields must be connected at both ends. The input lines should be no more than 500m in length. All reference (L-, C-, or I- depending on the controller) connections are interconnected on the module in the form of common reference pole.

Cables are clipped to the shield contact plate when connecting counter inputs. Remove about 2 cm of the outer cable insulation so that the mesh is exposed at the point where the cable is clipped to the plate.



IMPORTANT	Make sure that the mesh comes in direct contact with the shield contact plate. If the mesh does not touch the plate, the cable is not grounded.

IMPORTANT	Do not terminate unused high speed counter inputs.

To ensure that counters are used in a safety-related manner (SIL3 in accordance to IEC 61508), the whole system, including connected sensors and encoders, must satisfy these safety requirements. Refer to the *GuardPLC Controllers Safety Reference Manual,* publication number 1755-RM001, for more detailed information.

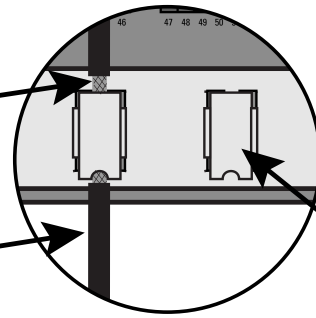
Safety-Related Analog Outputs

GuardPLC safety controllers use analog outputs to transfer analog values from the user program into outputs ranging from ±10V dc to 0 to 20 mA. The relationship between the value in the user program and the output value is linear and is displayed in the following table.

Logic Value	Output Voltage	Output Current
0	0.00V	0.0 mA
1000	10.00V	20.0 mA
-1000	-10.00V	na

Unused current inputs must be short-circuited.

Shielded cabling is fed in from below so that the shielding can be connected to the shield contact plate using a clip. Remove about 2 cm of the outer cable insulation so that the mesh is exposed at the point where the cable is clipped to the plate. Position the clip over the uninsulated cable shielding and push it into the slots of the shield contact plate until it fits firmly in place, as shown below.



IMPORTANT	Make sure that the mesh comes in direct contact with the shield contact plate. If the mesh does not touch the plate, the cable is not grounded.

Power Supply Considerations

The power supply must provide a voltage between 20.4 and 28.8V dc. You must supply enough power to drive the controller, inputs, and outputs. To operate, GuardPLC controllers typically draw less than 1A at 24V dc. They require additional power to operate the inputs and outputs connected to the controller. Consider the power draw of the I/O when specifying the size of the power supply and required fusing.

The 24V dc voltage supply must feature galvanic isolation since inputs and outputs are not electrically isolated from the processor.(1) It must also meet the requirements of the Safety Extra Low Voltage (SELV – EN60950) or Protective Extra Low Voltage (PELV – EN60204) guidelines.

IMPORTANT	Protect the controller with a slow-blowing fuse.

ATTENTION	Before connecting the power supply, check for correct polarity, value and ripple. Do not reverse the L+ and L- terminals or damage to the controller will result. There is no reverse polarity protection.

(1) The I/O and CPU are only isolated from one another on the GuardPLC 2000.

GuardPLC 1600/1800 Controllers and Distributed I/O

The supply voltage is connected via a 4-pin connector which accommodates wire sizes up to 2.5 mm^2 (14 AWG). You only need to connect one wire to L+ and one wire to L-. Both L+ and L- terminals are internally connected. The other terminal can be used to daisy-chain 24V dc to addtional devices. The power supply connector is rated to 10A.

GuardPLC 1200

Both terminals must be used in parallel to allow the maximum current of 8A. (Each terminal maximum is 4A so both are required for 8A.)

If the power supply has only one (+) lead, a short bridge jumper must be installed between $L+_{(1)}$ and $L+_{(2)}$.

TIP The GuardPLC 1200 requires approximately 0.5A to operate. The remaining 7.5A is used to source power for inputs and outputs.

GuardPLC 2000

The GuardPLC 2000 controller features several different modules. These modules and their current draw specifications are listed in the table below.

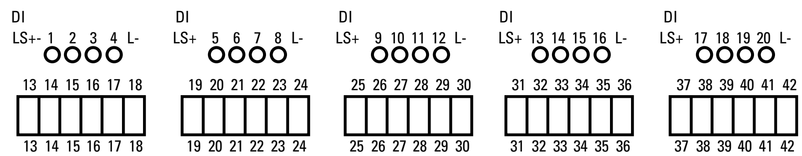
Module	Current Draw at 3.3V dc	Current Draw at 24V dc
1755-IB24XO16	0.3A	0.5A
1755-IF8	0.15A	0.4A
1755-OF8	0.15A	0.4A
1755-HSC	0.8A	0.1A
1755-L1	1.5A	1.0A

TIP The GuardPLC 2000 can draw up to 30A. The majority of this 30A is used to source inputs and outputs. Only 1A is required to operate the CPU module.

Connect the power supply unit, 1755-PB720, to the 24V dc supply voltage. Refer to the *GuardPLC 2000 Power Supply Installation Instructions*, publication number 1755-IN007, for detailed instructions.

GuardPLC 1600 Terminal Connections and Other Considerations

Safety-Related Digital Inputs



Digital inputs are connected to the following terminals:

Terminal Number	Designation	Function
13	LS+	Sensor supply for inputs 1 to 4
14	1	Digital input 1
15	2	Digital input 2
16	3	Digital input 3
17	4	Digital input 4
18	L-	Reference pole
19	LS+	Sensor supply for inputs 5 to 8
20	5	Digital input 5
21	6	Digital input 6
22	7	Digital input 7
23	8	Digital input 8
24	L-	Reference pole
25	LS+	Sensor supply for inputs 9 to 12
26	9	Digital input 9
27	10	Digital input 10
28	11	Digital input 11
29	12	Digital input 12
30	L-	Reference pole
31	LS+	Sensor supply for inputs 13 to 16
32	13	Digital input13
33	14	Digital input 14
34	15	Digital input 15
35	16	Digital input 16
36	L-	Reference pole

Terminal Number	Designation	Function
37	LS+	Sensor supply for inputs 17 to 20
38	17	Digital input 17
39	18	Digital input 18
40	19	Digital input 19
41	20	Digital input 20
42	L-	Reference pole

Safety-Related Digital Outputs.

Digital outputs are connected to the following terminals:

Terminal Number	Designation	Function	Current
1	L-	Reference pole	—
2	1	Digital output 1	0.5 A
3	2	Digital output 2	0.5 A
4	3	Digital output 3	0.5 A
5	4	Digital output 4 (for increased load)	2.0 A
6	L-	Reference pole	—
7	L-	Reference pole	—
8	5	Digital output 5	0.5 A
9	6	Digital output 6	0.5 A
10	7	Digital output 7	0.5 A
11	8	Digital output 8 (for increased load)	2.0 A
12	L-	Reference pole	—

GuardPLC 1800 Terminal Connections and Other Considerations

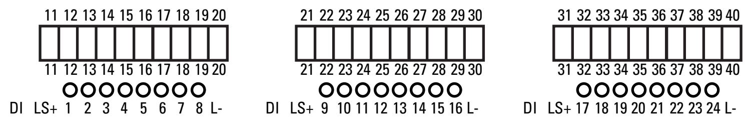
The controller has 24 digital inputs whose status is indicated via LEDs when in RUN mode. The digital inputs are actually analog inputs that provide the program with UINT values of 0 to 30V (0 to 3000), which are used to create limit values to calculate signals for the digital inputs. Default settings are:

- <7V = 0 signal
- >13V = 1 signal

The limit values are set using system variables. See page B-16 for more information on configuring these inputs.

TIP Since digital inputs are actually analog values, the .USED variable must be set HI in the output signal connections dialog to activate the digital input. See step 6 on page 5-7 for an example.

Safety-Related Digital Inputs

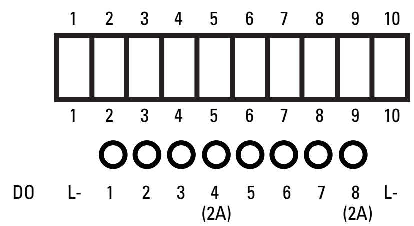


Digital inputs are connected to the following terminals:

Terminal Number	Designation	Function
11	LS+	Sensor supply for inputs 1 to 8
12	1	Digital input 1
13	2	Digital input 2
14	3	Digital input 3
15	4	Digital input 4
16	5	Digital input 5
17	6	Digital input 6
18	7	Digital input 7
19	8	Digital input 8
20	L-	reference pole
21	LS+	Sensor supply for inputs 9 to 16
22	9	Digital input 9
23	10	Digital input 10
24	11	Digital input 11
25	12	Digital input 12
26	13	Digital input 13
27	14	Digital input 14
28	15	Digital input 15
29	16	Digital input 16
30	L-	Reference pole

Terminal Number	Designation	Function
31	LS+	Sensor supply for inputs 17 to 24
32	17	Digital input 17
33	18	Digital input 18
34	19	Digital input 19
35	20	Digital input 20
36	21	Digital input 21
37	22	Digital input 22
38	23	Digital input 23
39	24	Digital input 24
40	L-	Reference pole

Safety-Related Digital Outputs



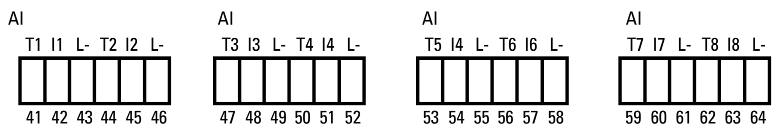
Digital outputs are connected to the following terminals:

Terminal Number	Designation	Function	Current
1	L-	Reference pole	—
2	1	Digital output 1	0.5 A
3	2	Digital output 2	0.5 A
4	3	Digital output 3	0.5 A
5	4	Digital output 4 (for increased load)	2.0 A
6	5	Digital output 5	0.5 A
7	6	Digital output 6	0.5 A
8	7	Digital output 7	0.5 A
9	8	Digital output 8 (for increased load)	2.0 A
10	L-	Reference pole	—

Safety-Related Analog Inputs

The GuardPLC 1800 features 8 single-ended analog inputs. Differential analog inputs cannot be used on the GuardPLC 1800. Two- or four-wire transmitters can be used. These devices can be powered from the transmitter supply terminal of the GuardPLC 1800 or from an external power supply.

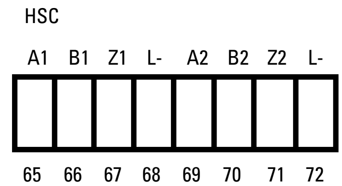
IMPORTANT Unused analog inputs must be short-circuited. See page 3-12.



The analog inputs are connected to the following terminals:

Terminal Number	Designation	Function
41	T1	Transmitter supply 1
42	I1	Analog input 1
43	L-	Reference pole
44	T2	Transmitter supply 2
45	I2	Analog input 2
46	L-	Reference pole
47	T3	Transmitter supply 3
48	I3	Analog input 3
49	L-	Reference pole
50	T4	Transmitter supply 4
51	I4	Analog input 4
52	L-	Reference pole
53	T5	Transmitter supply 5
54	I5	Analog input 5
55	L-	Reference pole
56	T6	Transmitter supply 6
57	I6	Analog input 6
58	L-	Reference pole
59	T7	Transmitter supply 7
60	I7	Analog input 7
61	L-	Reference pole
62	T8	Transmitter supply 8
63	I8	Analog input 8
64	L-	Reference pole

Safety Related High Speed Counter



Counters are connected to the following terminals:

Terminal Number	Designation	Counter Function	Gray Code Function
65	A1	Input A1	bit 0 (LSB)
66	B1	Input B1	bit 1
67	Z1	Input Z1	bit 2 (MSB)
68	L-	Common reference pole	
69	A2	Input A2	bit 0 (LSB)
70	B2	Input B2	bit 1
71	Z2	Input Z2	bit 2 (MSB)
72	L-	Common reference pole	

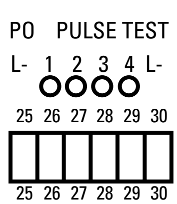
1753-IB16 Terminal Connections and Other Considerations

The 1753-IB16 input module features 16 digital inputs and 4 pulse test sources.

Since the 1753-IB16 is the only module/controller without digital outputs, it is equipped with four pulse test sources that can be software-configured for pulse testing of safety inputs, if required. Due to minimal current capacity, these pulse test sources cannot be used as outputs if they are not configured as pulse test sources.

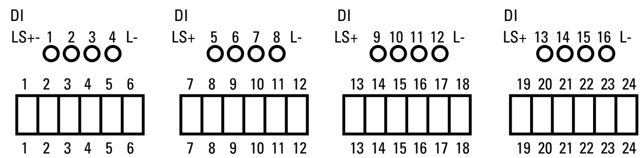
For information on configuring pulse test sources for line control, see Chapter 16.

Pulse Test Sources



Terminal Number	Designation	Function
25	L-	Reference pole
26	1	Pulse test source 1
27	2	Pulse test source 2
28	3	Pulse test source 3
29	4	Pulse test source 4
30	L-	Reference pole

Safety-Related Digital Inputs



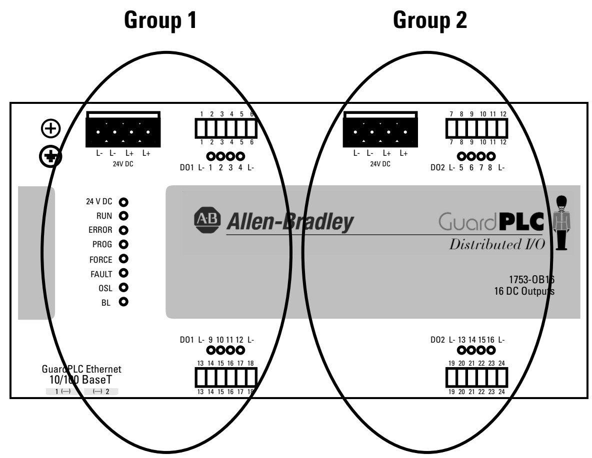
Digital inputs are connected to the following terminals:

Terminal Number	Designation	Function
1	LS+	Sensor supply for inputs 1 to 4
2	1	Digital input 1
3	2	Digital input 2
4	3	Digital input 3
5	4	Digital input 4
6	L-	Reference pole
7	LS+	Sensor supply for inputs 5 to 8
8	5	Digital input 5
9	6	Digital input 6
10	7	Digital input 7
11	8	Digital input 8
12	L-	Reference pole
13	LS+	Sensor supply for inputs 9 to 12
14	9	Digital input 9
15	10	Digital input 10
16	11	Digital input 11
17	12	Digital input 12
18	L-	Reference pole
19	LS+	Sensor supply for inputs 13 to 16
20	13	Digital input 13
21	14	Digital input 14
22	15	Digital input 15
23	16	Digital input 16
24	L-	Reference pole

1753-OB16 Terminal Connections and Other Considerations

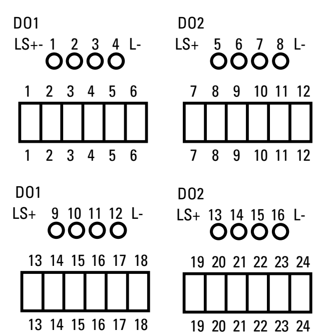
Operating Voltage Considerations

The 1753-OB16 is the only module/controller that has a total current capacity (16A) higher than the terminal block current limitation (10A). Therefore, it features two separate operating voltage supply connections. Each group of 8 outputs has a current capacity of 8A, and is powered by a separate voltage supply. Output groups are comprised of the following:



Group	Outputs
1	1, 2, 3, 4, and 9, 10, 11,12
2	5, 6, 7, 8 and 13, 14, 15, 16

The module has 16 digital outputs (DO1 to DO16) whose status is indicated via LEDs.



Each output is rated for up to 1A at 60° C (140°F) or 2A at 40°C (104°F). However, each group of 8 outputs may not exceed 8A total. For heat dissipation, intersperse high-current and low-current outputs so that all the high-current outputs are not next to each other.

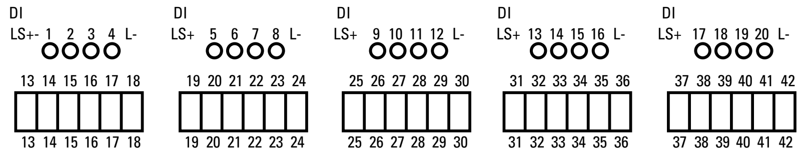
The digital outputs are connected to the following terminals:

Terminal Number	Designation	Function
1	L-	Reference pole
2	1	Digital output DO 1
3	2	Digital output DO 2
4	3	Digital output DO 3
5	4	Digital output DO 4
6	L-	Reference pole
7	L-	Reference pole
8	5	Digital output DO 5
9	6	Digital output DO 6
10	7	Digital output DO 7
11	8	Digital output DO 8
12	L-	Reference pole
13	L-	Reference pole
14	9	Digital output DO 9
15	10	Digital output DO 10
16	11	Digital output DO 11
17	12	Digital output DO 12
18	L-	Reference pole
19	L-	Reference pole
20	13	Digital output DO 13
21	14	Digital output DO 14
22	15	Digital output DO 15
23	16	Digital output DO 16
24	L-	Reference pole

1753-IB20XOB8 Terminal Connections and Other Considerations

The remote I/O module features 20 digital inputs and 8 digital outputs whose status is indicate via LEDs.

Safety-Related Digital Inputs



The digital inputs are connected to the following terminals:

Terminal Number	Designation	Function
13	LS+	Sensor supply for inputs 1 to 4
14	1	Digital input 1
15	2	Digital input 2
16	3	Digital input 3
17	4	Digital input 4
18	L-	Reference pole
19	LS+	Sensor supply for inputs 5 to 8
20	5	Digital input 5
21	6	Digital input 6
22	7	Digital input 7
23	8	Digital input 8
24	L-	Reference pole
25	LS+	Sensor supply for inputs 9 to 12
26	9	Digital input 9
27	10	Digital input 10
28	11	Digital input 11
29	12	Digital input 12
30	L-	Reference pole
31	LS+	Sensor supply for inputs 13 to 16
32	13	Digital input 13
33	14	Digital input 14
34	15	Digital input 15
35	16	Digital input 16
36	L-	Reference pole
37	LS+	Sensor supply for inputs 17 to 20
38	17	Digital input 17
39	18	Digital input 18
40	19	Digital input 19
41	20	Digital input 20
42	L-	Reference pole

Safety-Related Digital Outputs

The module has 8 digital outputs (DO1 to DO8) whose status is indicated via LEDs.

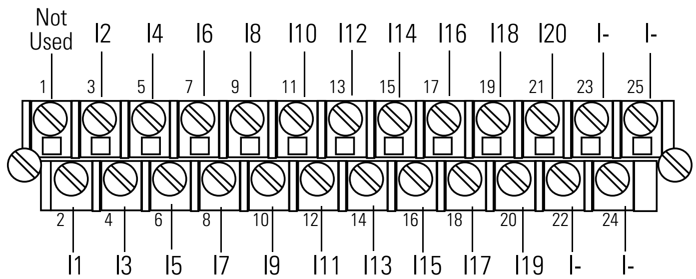
The digital outputs are connected to the following terminals:

Terminal Number	Designation	Function	Current
1	L-	Reference pole	—
2	1	Digital output 1	0.5 A
3	2	Digital output 2	0.5 A
4	3	Digital output 3	0.5 A
5	4	Digital output 4 (for increased load)	2.0 A
6	L-	Reference pole	—
7	L-	Reference pole	—
8	5	Digital output 5	0.5 A
9	6	Digital output 6	0.5 A
10	7	Digital output 7	0.5 A
11	8	Digital output 8 (for increased load)	2.0 A
12	L-	Reference pole	—

GuardPLC 1200 Terminal Connections and Other Considerations

The GuardPLC 1200 has no LS+ terminal for a safety input voltage source. Use the L+ supply terminal as the source for safety input voltage. The four reference terminals, labeled I-, should be used for the safety input voltage reference. This is a common reference for all 20 inputs.

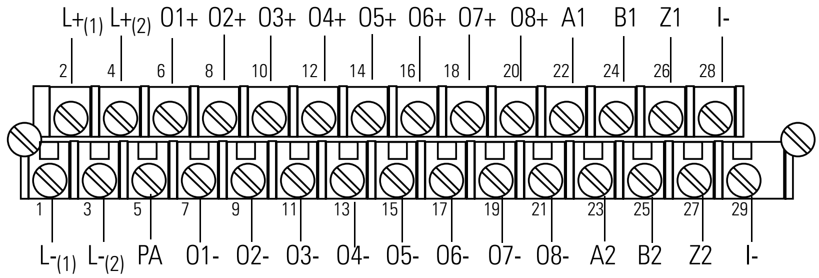
Lower Terminal Block



Terminal Number	Designation	Function
1	Not Used	None
2	I1	Digital input 1
3	I2	Digital input 2
4	I3	Digital input 3
5	I4	Digital input 4
6	I5	Digital input 5
7	I6	Digital input 6
8	I7	Digital input 7
9	I8	Digital input 8
10	I9	Digital input 9
11	I10	Digital input 10
12	I11	Digital input 11
13	I12	Digital input 12
14	I13	Digital input 13
15	I14	Digital input 14
16	I15	Digital input 15
17	I16	Digital input 16
18	I17	Digital input 17
19	I18	Digital input 18
20	I19	Digital input 19
21	I20	Digital input 20
22	I-	Reference pole
23	I-	Reference pole
24	I-	Reference pole
25	I-	Reference pole

All eight of the digital output zero-voltage reference terminals are common. Unlike the GuardPLC 1600/1800 or distributed I/O, which have an earth ground screw, the GuardPLC 1200 earth ground should be wired to the PA terminal.

Upper Terminal Block



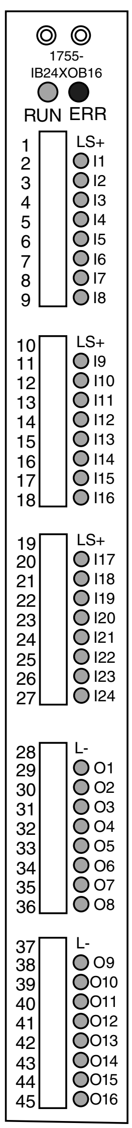
Terminal Number	Designation	Function
1	L-	24V dc return path
2	L+	24V dc power input
3	L-	24V dc return path
4	L+	24V dc power input
5	PA	Functional ground
6	O1+	Digital output 1
7	O1-	Voltage reference for digital output 1
8	O2+	Digital output 2
9	O2-	Voltage reference for digital output 2
10	O3+	Digital output 3
11	O3-	Voltage reference for digital output 3
12	O4+	Digital output 4
13	O4-	Voltage reference for digital output 4
14	O5+	Digital output 5
15	O5-	Voltage reference for digital output 5
16	O6+	Digital output 6
17	O6-	Voltage reference for digital output 6
18	O7+	Digital output 7
19	O7-	Voltage reference for digital output 7
20	O8+	Digital output 8
21	O8-	Voltage reference for digital output 8
22	A1	Universal signal input for counter 1
23	A2	Universal signal input for counter 2
24	B1	Signal input for counting direction for counter 1
25	B2	Signal input for counting direction for counter 2
26	Z1	Reset for counter 1
27	Z2	Reset for counter 2
28	I-	Signal ground for counters 1 and 2
29	I-	Signal ground for counters 1 and 2

GuardPLC 2000 Terminal Connections and Other Considerations

1755-IB24XOB16 Digital I/O Module

This module is a combination I/O module featuring 24 safety-related digital inputs and 16 safety-related digital outputs.

- **Inputs:** The sockets with pins 2 through 9, 11 through 18, and 20 through 27 provide the 24 digital inputs I1 to I24. Pins 1, 10, and 19 are the common positive poles (LS+). Each group of 8 inputs has current limits of 100mA.
- **Outputs:** The sockets with pins 29 through 36 and 38 through 45 provide the 16 digital outputs O1 to O16. Pins 28 and 37 are the common negative poles (L-) for the output loads.
- Each output channel can be loaded with 2A, but the total load of all 16 outputs must not exceed 8A.



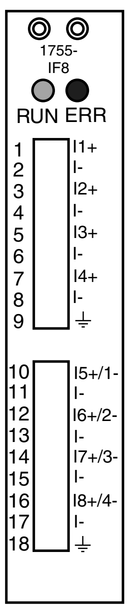
Terminal Number	Designation	Function	Terminal Number	Designation	Function
1	LS+	Digital input supply for inputs 1 to 8	24	I21	Digital input 21
2	I1	Digital input 1	25	I22	Digital input 22
3	I2	Digital input 2	26	I23	Digital input 23
4	I3	Digital input 3	27	I24	Digital input 24
5	I4	Digital input 4	28	L-	Reference pole for outputs 1 to 8
6	I5	Digital input 5	29	O1	Digital output 1
7	I6	Digital input 6	30	O2	Digital output 2
8	I7	Digital input 7	31	O3	Digital output 3
9	I8	Digital input 8	32	O4	Digital output 4
10	LS+	Digital input supply for inputs 9 to 16	33	O5	Digital output 5
11	I9	Digital input 9	34	O6	Digital output 6
12	I10	Digital input 10	35	O7	Digital output 7
13	I11	Digital input 11	36	O8	Digital output 8
14	I12	Digital input 12	37	L-	Reference pole for outputs 9 to 16
15	I13	Digital input 13	38	O9	Digital output 9
16	I14	Digital input 14	39	O10	Digital output 10
17	I15	Digital input 15	40	O11	Digital output 11
18	I16	Digital input 16	41	O12	Digital output 12
19	LS+	Digital input supply for inputs 17 to 24	42	O13	Digital output 13
20	I17	Digital input 17	43	O14	Digital output 14
21	I18	Digital input 18	44	O15	Digital output 15
22	I19	Digital input 19	45	O16	Digital output 16
23	I20	Digital input 20			

1755-IF8 Analog Input Module

This module features 8 single-ended analog inputs or 4 differential analog inputs. Two-wire or four-wire transmitters can be used. The devices cannot be powered from the GuardPLC module. An external power supply is required for all analog transmitters. Single-ended transmitters connect between the I*x* and I- terminals. For example: pins 1 and 2, 3 and 4, 5 and 6. Differential transmitters connect between I*x* and *x*- terminals. For example: pins 1 and 10, 3 and 12, 5 and 14.

IMPORTANT Unused channels must be short-circuited. See page 3-12.

All reference poles (I-) are internally connected.



Terminal Number	Designation	Function
1	I1+	Analog input 1
2	I-	Reference pole for input 1
3	I2+	Analog input 2
4	I-	Reference pole for input 2
5	I3+	Analog input 3
6	I-	Reference pole for input 3
7	I4+	Analog input 4
8	I-	Reference pole for input 4
9	shield connection	signal ground
10	I5+/1-	Analog input 5
11	I-	Reference pole for input 5
12	I6/2-	Analog input 6
13	I-	Reference pole for input 6
14	I7+/3-	Analog input 7
15	I-	Reference pole for input 7
16	I8+/4-	Analog input 8
17	I-	Reference pole for input 8
18	shield connection	signal ground

1755-OF8 Analog Output Module

This module features 8 analog outputs. Devices cannot be powered from the 1755-OF8 module. An external power supply is required for all analog output devices.

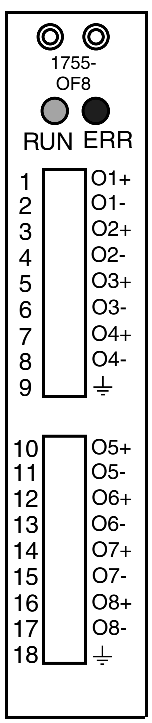
There are 4 reference poles for the 8 outputs. A pair of outputs share a reference pole as shown below.

These outputs:	Share these Reference Poles:
1 and 2	O1- and O2-
3 and 4	O3- and O4-
5 and 6	O5- and O6-
7 and 8	O7- and O8-

Each group of 2 outputs is electrically isolated from the others.

IMPORTANT If an unused channel is defined as a current output (software configuration set to "current output"), the output channel has to be short-circuited. Place jumpers into these outputs and tighten the screws.

IMPORTANT If an unused channel is defined as a voltage output (software configuration set to "voltage output"), the unused outputs must be left open. Short-circuiting a unused voltage output may cause damage to the output.



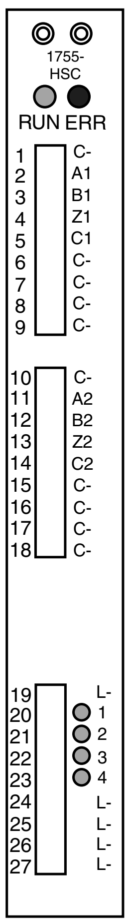
Terminal Number	Designation	Function
1	O1+	Analog output 1
2	O1-	Group 1 reference pole
3	O2+	Analog output 2
4	O2-	Group 1 reference pole
5	O3+	Analog output 3
6	O3-	Group 2 reference pole
7	O4+	Analog output 4
8	O4-	Group 2 reference pole
9	shield connection	signal ground
10	O5+	Analog output 5
11	O5-	Group 3 reference pole
12	O6+	Analog output 6
13	O6-	Group 3 reference pole
14	O7+	Analog output 7
15	O7-	Group 4 reference pole
16	O8+	Analog output 8
17	O8-	Group 4 reference pole
18	shield connection	signal ground

1755-HSC Counter Modules

This module contains 2 high speed counters and 4 digital outputs. Although the 4 digital outputs are located on the 1755-HSC module, they cannot be driven by counter presets. The 4 digital outputs are driven by software, just as on the 1755-IB24XOB16 module.

The nominal current per output is limited to ≤ 0.5A. Currents > 0.5A are regarded as overload. The overload is limited to ≤ 11A per output, or ≤ 2A if all four outputs are overloaded at the same time. With an overload of 2A, the output voltage drops to 18V.

All counter common reference poles, C-, share the same path. All digital output common reference poles, L-, share the same path, but are electrically isolated from the C- pins.



Terminal Number	Designation	Function
1	C-	Common reference pole
2	A1	Signal input for counter 1
3	B1	Counting direction input for counter 1
4	Z1	Reset input for counter 1
5	C1	no function
6	C-	Common reference pole
7	C-	Common reference pole
8	C-	Common reference pole
9	C-	Common reference pole
10	C-	Common reference pole
11	A2	Signal input for counter 2
12	B2	Counting direction input for counter 2
13	Z2	Reset input for counter 2
14	C2	no function
15	C-	Common reference pole
16	C-	Common reference pole
17	C-	Common reference pole
18	C-	Common reference pole
19	L-	Reference pole for digital outputs
20	1	Digital output 1
21	2	Digital output 2
22	3	Digital output 3
23	4	Digital output 4
24	L-	Reference pole for digital outputs
25	L-	Reference pole for digital outputs
26	L-	Reference pole for digital outputs
27	L-	Reference pole for digital outputs

Grounding

You must provide an acceptable grounding path for each device in your application. For more information on proper grounding guidelines, refer to the *Industrial Automation Wiring and Grounding Guidelines*, publication number 1770-4.1.

Grounding Considerations for All Controllers

- To improve EMC conditions, ground the controller.
- Run the ground connection from the ground screw of the controller to a good earth ground. Use a minimum of 2.5 mm^2 (14 AWG) wire.
- Keep the connection to earth ground as short as possible to minimize resistance.
- Grounding is required even if the control system does not have shielded cables.
- If shielded cables are used to connect the controller to the external 24V dc source, connect the shield to the grounding contact of the power supply.
- No protective grounding (against hazardous shock) is required.

GuardPLC 1200

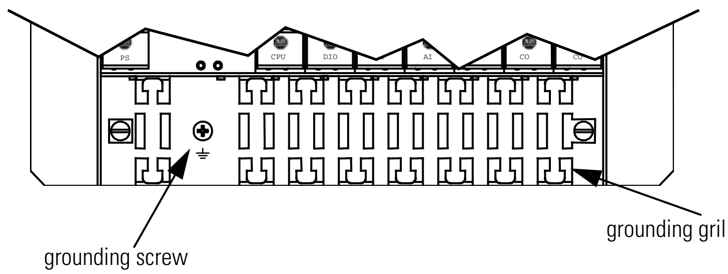
Ground the GuardPLC 1200 by connecting the PA terminal to earth ground. See page 3-27 for GuardPLC 1200 terminal connections.

GuardPLC 1600 and GuardPLC 1800 Controllers and Distributed I/O

The controllers and I/O have a grounding screw located on the upper left of the housing and marked with the grounding symbol ⏚ . This grounding screw is common to the DIN rail connection. Attach an appropriate earth ground to the grounding screw.

GuardPLC 2000

Ground the GuardPLC 2000 chassis and cables using the grounding screw located on the left side of the grounding grill. Ground the chassis via the grounding grill.



Preventing Electrostatic Discharge

ATTENTION

Electrostatic discharge can damage integrated circuits or semiconductors. Follow these guidelines when you handle the module:

- Touch a grounded object to discharge static potential.
- Wear an approved wrist-strap grounding device.
- If available, use a static-safe workstation.
- When not in use, keep the GuardPLC controller in its static-shield box.

Connecting to the GuardPLC Controller

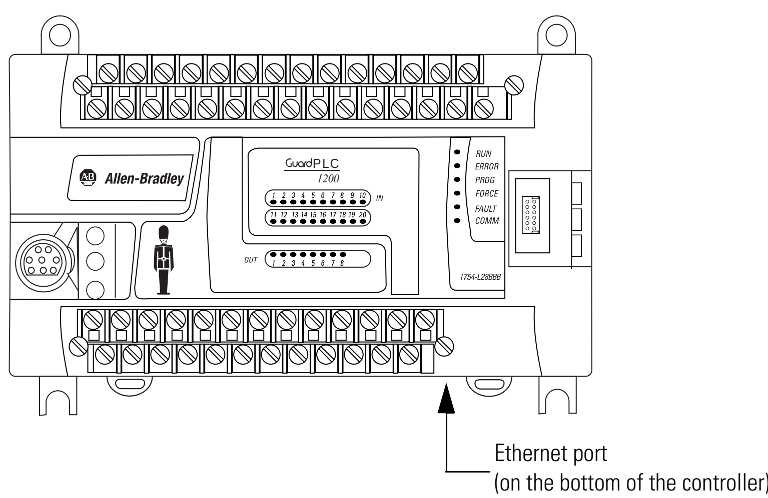
Using This Chapter

Connecting to the Controller via RSLogix Guard PLUS

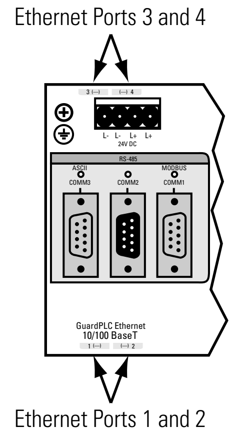
You connect the controller to the programming terminal via an Ethernet port on the controller. The programming terminal must have an Ethernet port or Ethernet communication card.

To directly connect the programming terminal to the controller, use a cross-over Ethernet cable. The GuardPLC 1600 and 1800 feature auto-sensing ports so that a cross-over or straight-thru cable may be used.

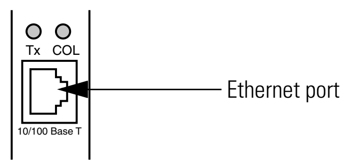
Connecting to a GuardPLC 1200 Controller



Connecting to a GuardPLC 1600 or 1800 Controller



Connecting to a GuardPLC 2000 Controller



GuardPLC Factory Defaults

IP Address	192.168.0.99
Subnet Mask	255.255.252.0
SRS[1]	60000

(1) The SRS code is compiled with the program. It guarantees that the program can only be downloaded to a GuardPLC with a matching SRS stored in non-volatile memory.

Understanding Ethernet Addressing

As with any connection between devices on Ethernet, the IP address and subnet mask determine if the connection can take place. Every device on Ethernet has an IP address and subnet mask.

The IP address and subnet mask are made up of four (4) octets (001.002.003.004) The IP address is made up of the Network ID (octets 001 and 002) and the Host ID (octets 003 and 004). The Network ID portion of the IP address is derived from the subnet mask.

When any two devices attempt to talk on Ethernet, a check is made to see if the Network ID of both the originator and the destination address match. If they match, then the message is sent on the local network. If they do not match, then the message is sent to the Gateway to route the message to the destination. The subnet masks of all the devices on a local network should be the same.

The example below illustrates how to derive the Network ID based on the GuardPLC IP address and subnet mask defaults.

EXAMPLE

Determining the Network ID

GuardPLC Defaults:

IP Address 192.168.0.99	= 11000000 . 10101000 . 00000000 . 01100011
Subnet Mask 255.255.252.0	= 11111111 . 11111111 . 11111100 . 00000000
Network ID	= 11000000 . 10101000 . 000000xx . xxxxxxxx

Set up the programming terminal's IP address so that it has the same Network ID as the GuardPLC. Octets one and two have to be the same because the subnet mask octets are 255. The third subnet mask octet is 252, which means that only the last two bits can be different.

If the factory default settings above are used, the allowable IP addresses for the programming terminal running RSLogix Guard PLUS are:

- 192.168.0.xxx (xxx represents any value between 000-255)
- 192.168.1.xxx
- 192.168.2.xxx
- 192.168.3.xxx

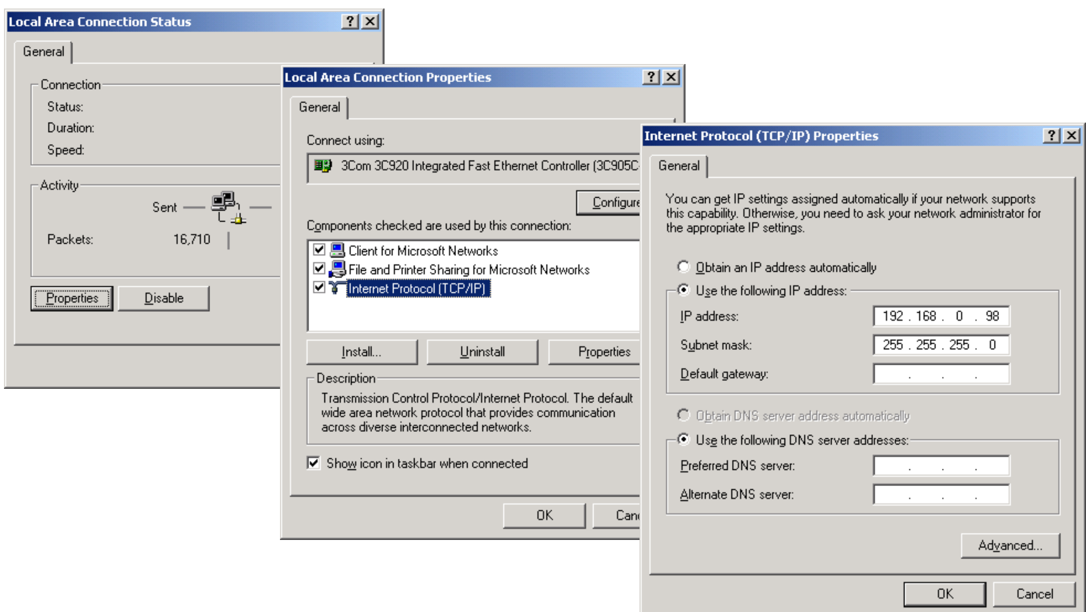
Configure the IP Address of Your Programming Terminal

IMPORTANT The first time you connect to a controller, you must use the factory-set IP address of 192.168.0.99 and the default SRS of 60000. After you establish communications with the controller (using the steps on the following pages), you can change the IP address and SRS to better accommodate your Ethernet network.

Change the IP address of your programming terminal running RSLogix Guard PLUS so that the GuardPLC and programming terminal can communicate on a local network.

TIP	If you suspect the GuardPLC has the factory-set default IP address of 192.168.0.99 and the default subnet mask of 255.255.252.0, set your programming terminal's IP address to 192.168.0.98 with a subnet of 255.255.252.0 to establish communications.

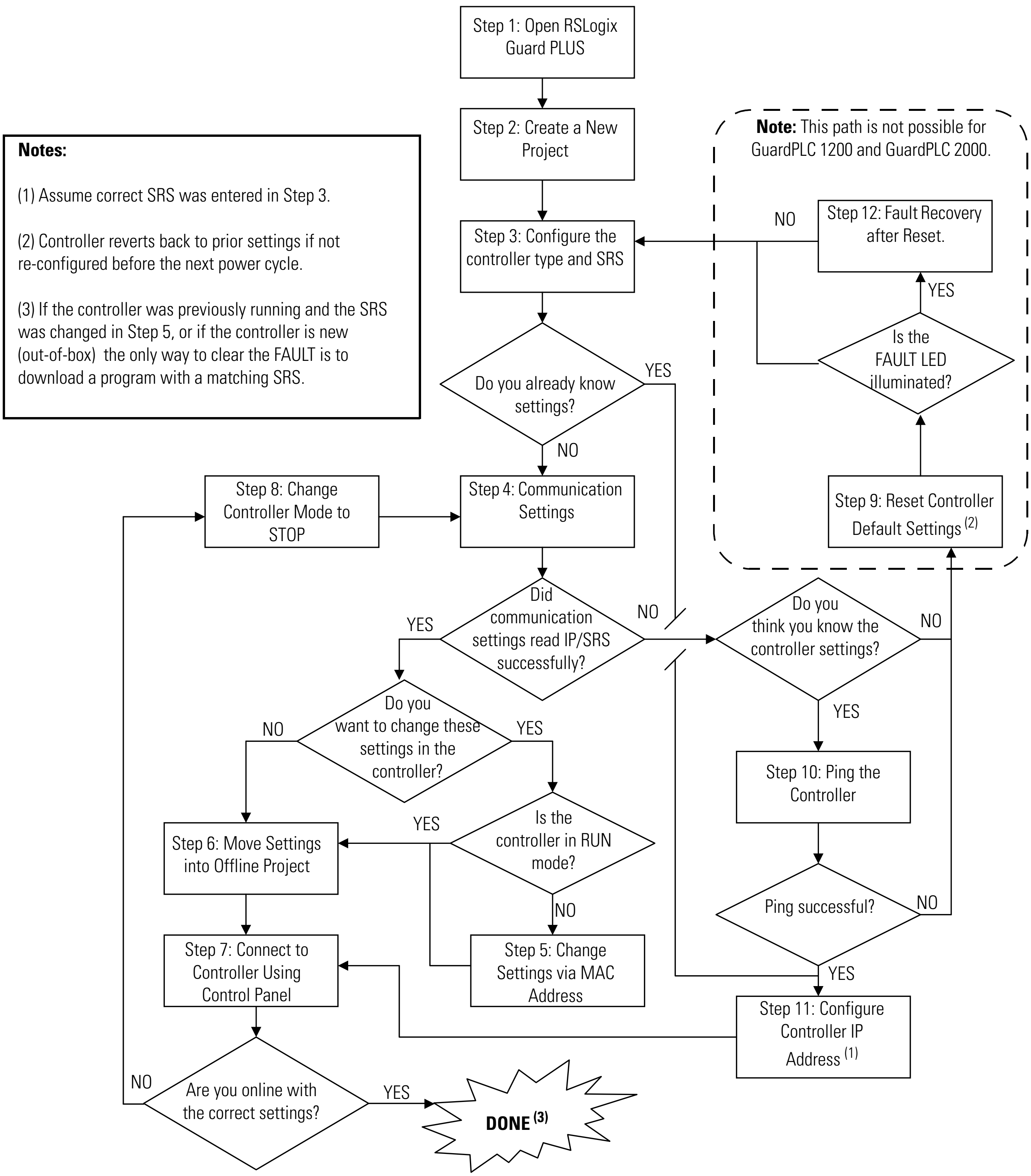
Change the IP address (via Windows 2000) by selecting *Start* → *Settings* → *Control Panel* → *Network and Dial-up Connections*. Open *Local Area Connections* and select *Properties*. Select *TCPIP* and *Properties*. Set the *General TCP/IP Properties* as shown below.



Confirm your settings by clicking *OK* in both dialog boxes.

Going Online with the GuardPLC Controller

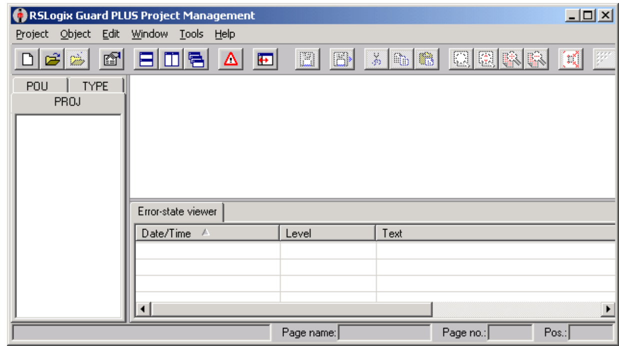
The following flowchart illustrates the steps required to successfully go online with the GuardPLC controller.



The steps are described in detail below.

Step 1: Open RSLogix Guard PLUS

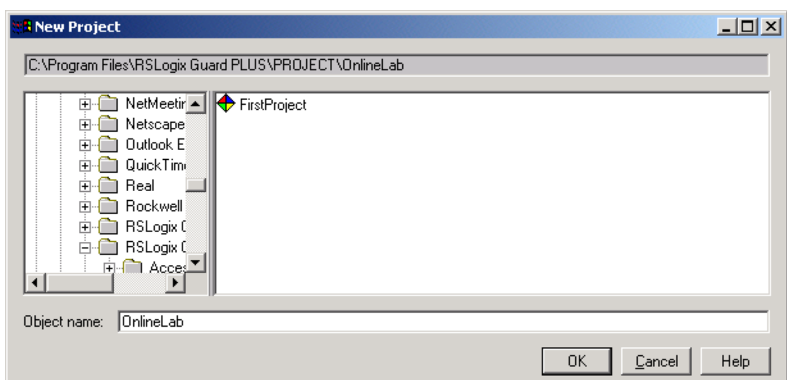
Select *Start* →*Programs* →*RSLogix Guard PLUS* →*RSLogixGuardPLUS*.



Step 2: Create a New Project

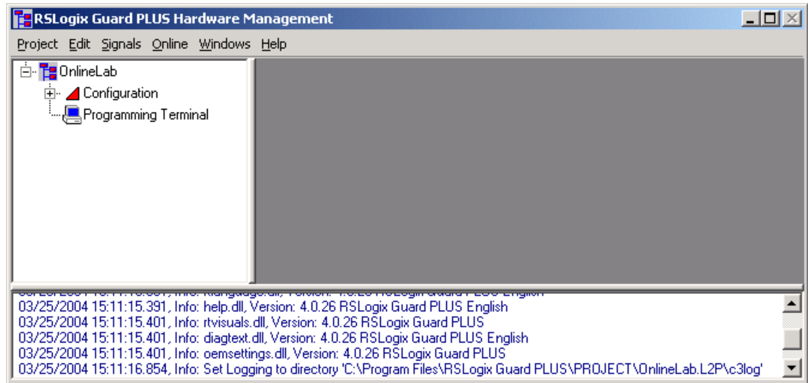
Open an existing project or create a new project that contains a GuardPLC controller.

1. To create a new project, select *Project* →*New* from the main menu or click on the *New* icon.
2. Enter the name of the project in the *Object Name* field.



3. Click *OK*.

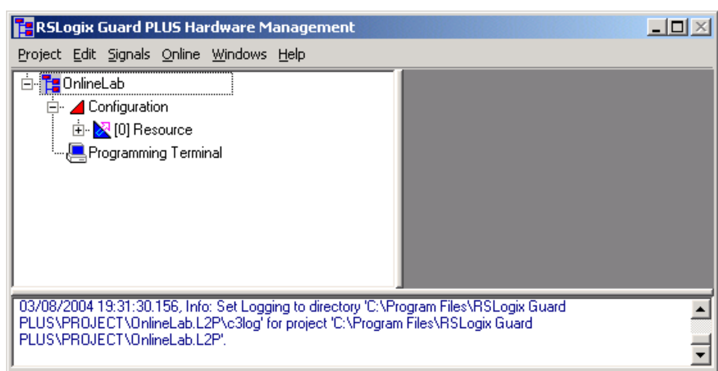
The RSLogix Guard PLUS Hardware Management window opens. Both the Project Management and Hardware Management windows are normally open when running RSLogix Guard PLUS.



Step 3: Configure the Controller Type and SRS

To go online, you must specify the controller type and change the default SRS. The software defaults to an SRS of zero (0), which is the only illegal SRS value. To accept the controller type, the SRS must be changed to a value between 2[1] and 65535.

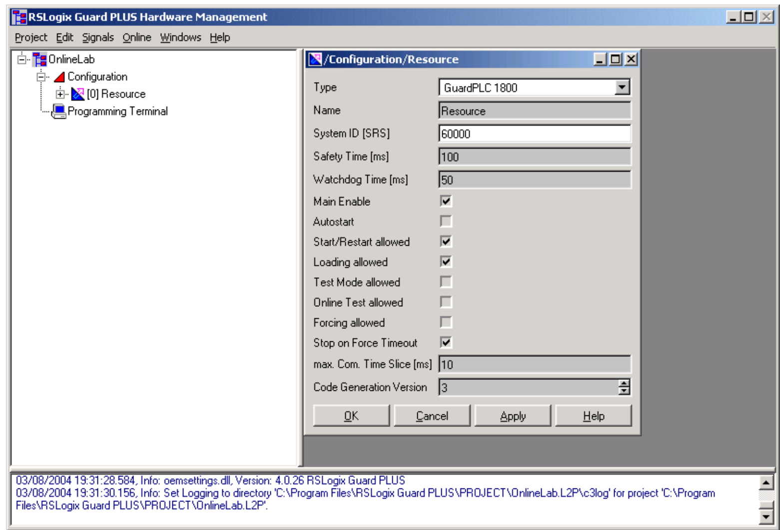
1. Expand the project tree in the Hardware Management window until *[0] Resource* is visible.



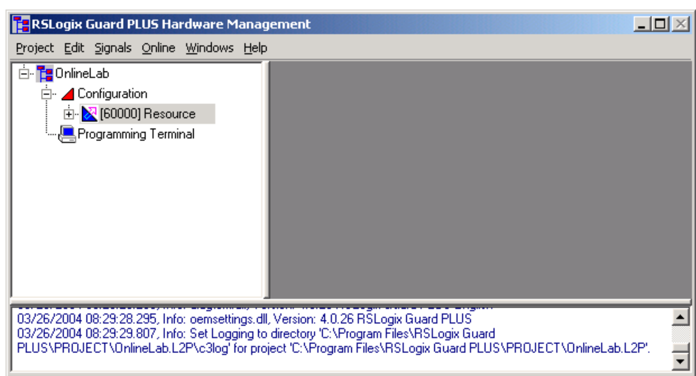
2. Right-click on *[0] Resource* and select *Properties*.

(1) The programming terminal defaults to 1.

3. Specify the controller type and enter an SRS of 60000. You must use the default SRS of 60000 the first time you connect to a GuardPLC controller.

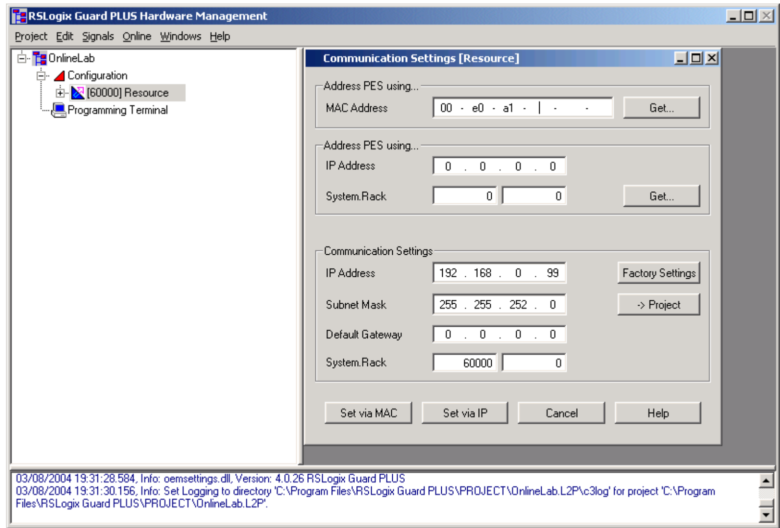


4. Click *OK*. The Hardware Management window should appear as shown below. Notice that the SRS has changed to 60000.

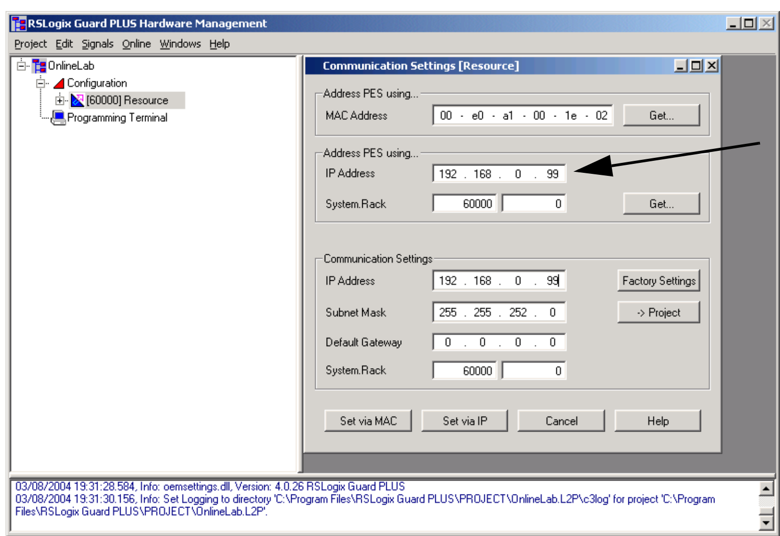


Step 4: Communication Settings

1. Select *Online* → *Communication Settings* from the pull-down menu.

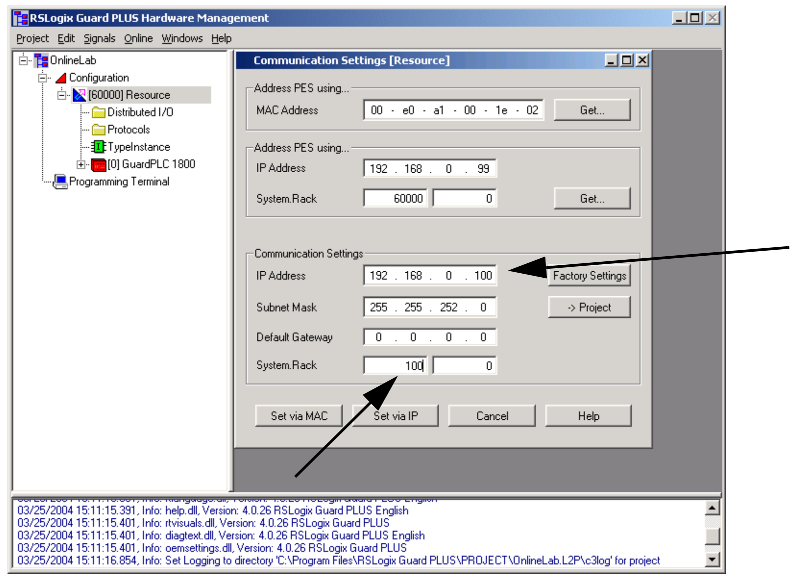


2. The MAC address is on the sticker on the side of a GuardPLC 1200 controller, on the label positioned over both lower RJ-45 connections on GuardPLC 1600/1800 controllers and I/O, or on the front bezel of the AB-CPU module of a GuardPLC 2000 controller. Enter the last three elements of the MAC address into the *MAC Address* field and click on *Get*. The IP address and SRS of the GuardPLC should appear in the *Address PES using…* fields.



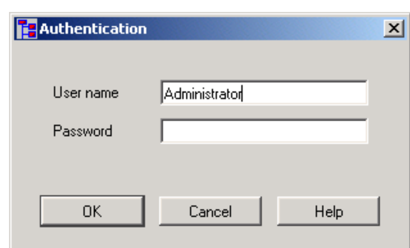
Step 5: Change Settings via MAC Address

1. Enter desired settings for the IP and SRS in the Communication Settings fields indicated by the arrows below.



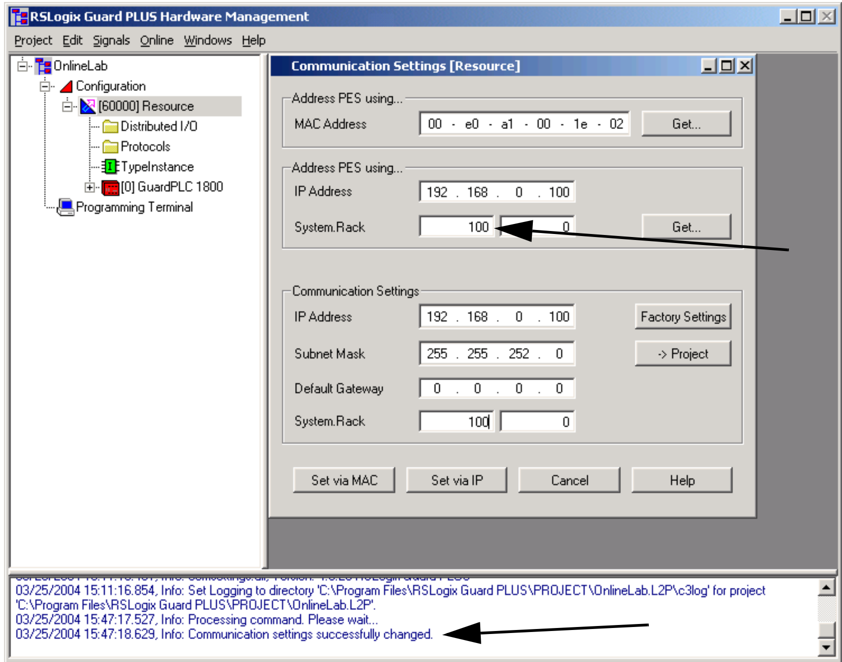
2. Click on the *Set via MAC* button.

3. The Authentication window appears. Enter the default username “Administrator” as shown below.



4. Click *OK*.

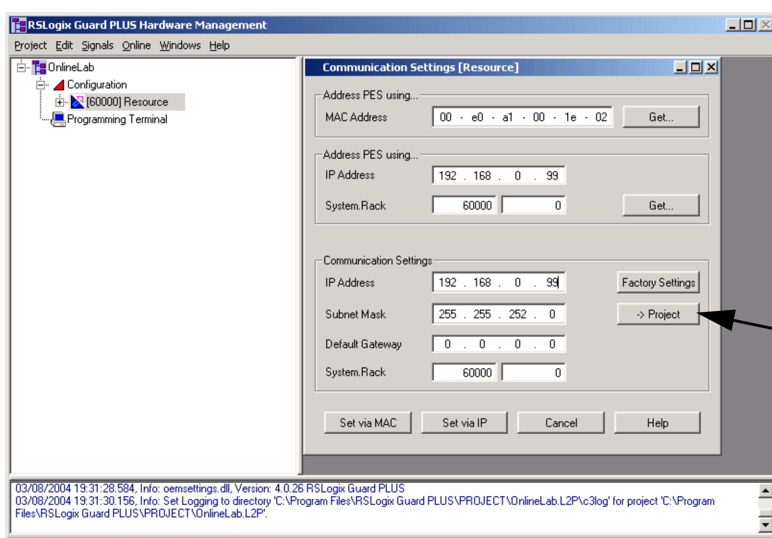
The IP address and SRS should have changed. If successful, a prompt appears at the bottom of the window and the settings in the middle fields change.



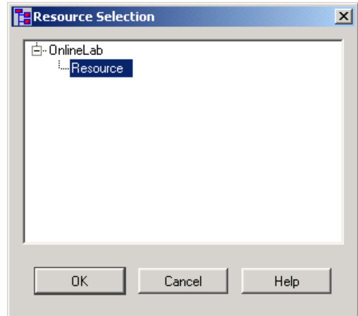
Step 6: Move the Settings Into Your Offline Project

If you wish to connect using the current GuardPLC settings, move the settings into your offline project.

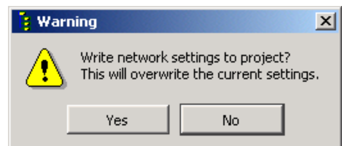
1. Left-click on *"-> Project"*.



2. The Resource Selection window appears. Make sure *Resource* is selected and click *OK*.



3. Answer *YES* to the warning prompt.



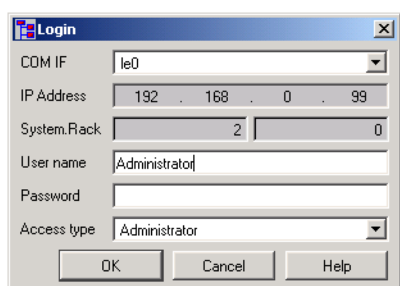
This moves the IP address and SRS of the GuardPLC to your offline project and overwrites the existing values. These new values will be used in the login screen to connect with the GuardPLC.

Step 7: Use the Control Panel to Connect to the GuardPLC

1. Right-click on *[60000] Resource*.

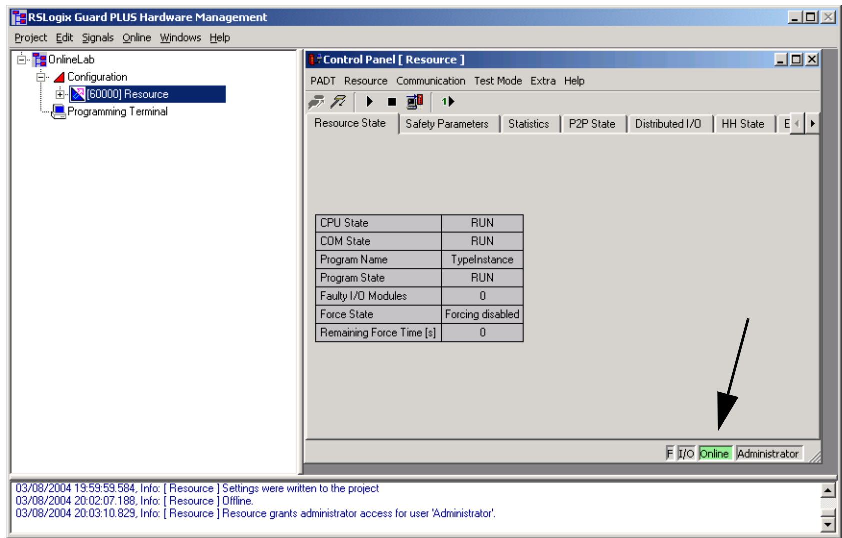
2. Select *Online →Control Panel.*

3. The Login Window appears. Type [Ctrl]+[A] to fill in the default *Username, Password,* and *Access Type.*



4. Click *OK*.

The Control Panel Online indicator will be GREEN if the controller is in RUN Mode. The Online indicator may also be yellow, white, or red based on its current state.

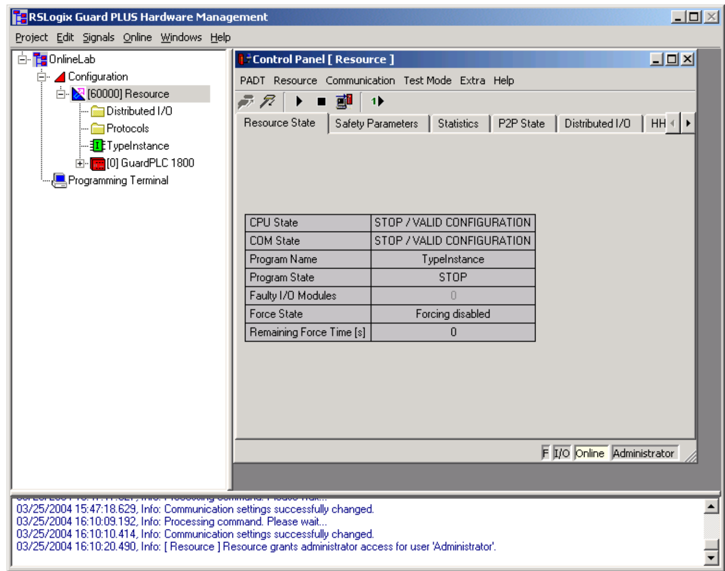


TIP If you are successfully online with the GuardPLC and in RUN mode (Green Online indicator as shown above), you do not need to continue with the steps 8 through 12 below. However, if you are not online and in RUN mode, consult the flowchart on page 4-5, and perform the appropriate steps.

Step 8: Change the Controller to STOP Mode

To change the controller to STOP mode, select *Resource →Stop* from the Control Panel or use the *Stop* icon.

When in STOP Mode, the Control Panel appears as follows:



Close the Control Panel.

Step 9: Reset the Controller to the Default Settings

In some cases, you may have to reset the GuardPLC to its default IP address and SRS.

GuardPLC 1600 and 1800 controllers have a Reset button that is accessible via a small hole directly to the right of the Ethernet ports on top of the controller. The Reset button returns the IP address, SRS and Password settings to:

IP Address	192.168.0.99
SRS	60000
Username	Administrator
Password	[none]

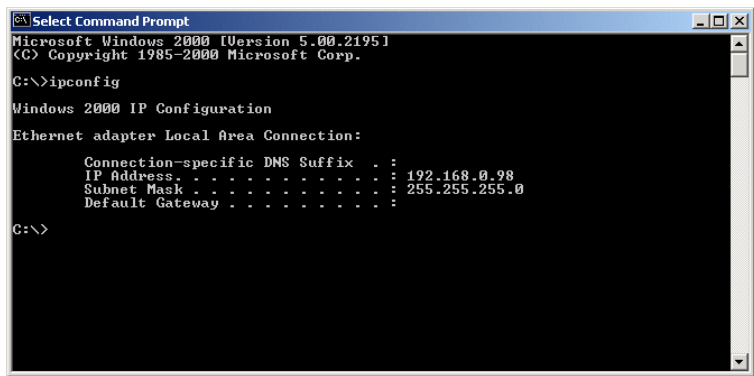
To reset the controller, hold down the RESET button, then power cycle the GuardPLC. Continue to hold down the Reset button until the PROG led stops flashing.

At the next power cycle, the settings will revert back to the last configured settings. These could be the settings in place prior to the Reset operation, if you did not reconfigure them after resetting the controller.

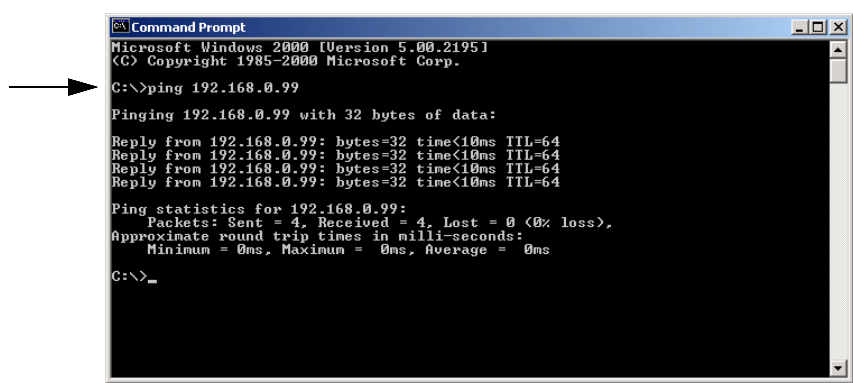
Step 10: Ping the Controller

Use the Start menu to open the RSLogix Guard PLUS Command Prompt by selecting *Start →Programs →RSLogix Guard PLUS → RSLogix Guard PLUS Command Prompt*.

Run IPCONFIG at the DOS Command prompt to verify your computer's IP address. It must be on the same local network as the GuardPLC.



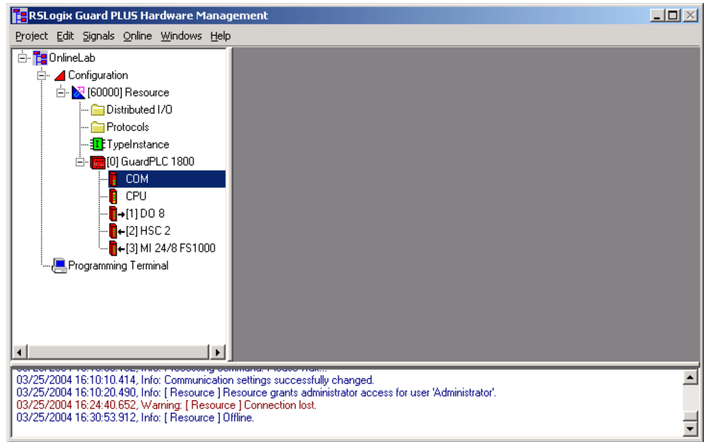
Ping the GuardPLC using the command shown at the C:\> below. If the ping is successful, the IP address of the GuardPLC has been verified and the Ethernet link is operating. If the ping was not successful either the IP address, subnet mask, or Ethernet link is not correct. The picture below is the result of a successful ping.



Type "exit" at the command prompt to close the Command Prompt window.

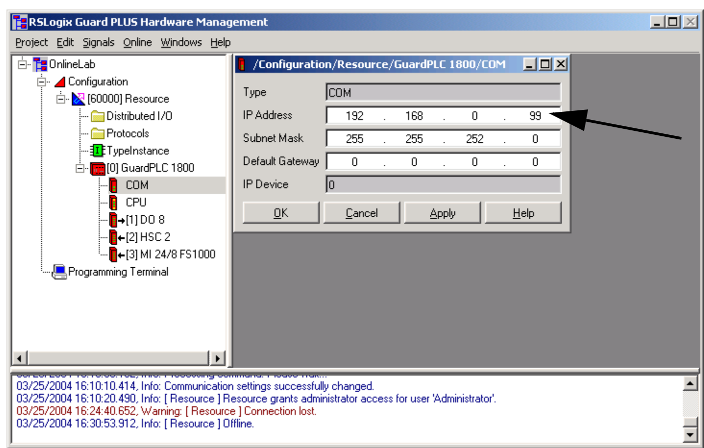
Step 11: Configure the GuardPLC Controller's IP Address

1. Expand the project tree in the Hardware Management window until the controller *COM* icon is visible.



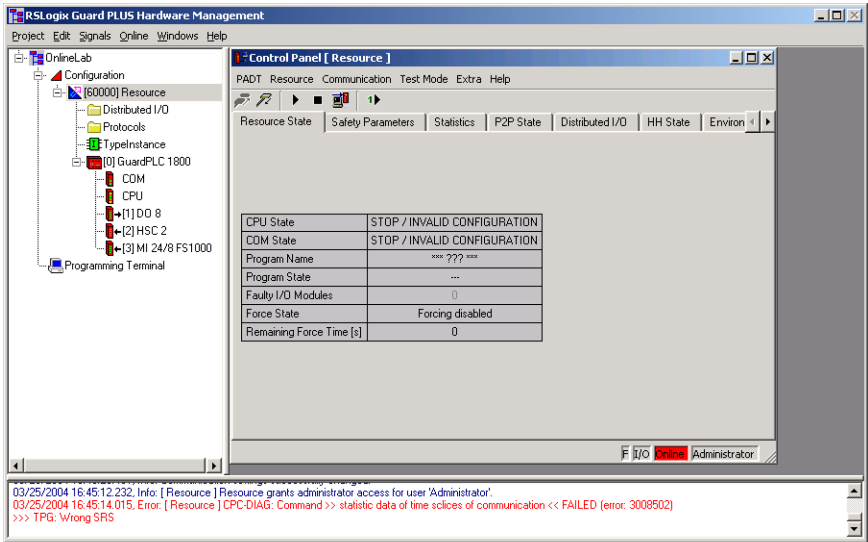
2. Right-click on *COM* and select *Properties*.

3. Edit the IP address to match the GuardPLC controller and click *OK*.



Step 12: Recovering from a Controller Fault After Using the RESET Button

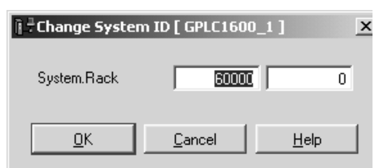
After using the Reset button, the Control Panel will appear as follows if the SRS was not originally 60000 prior to the Reset. The Fault LED on the front of the GuardPLC is illuminated, and the CPU State of the Resource is STOP/INVALID CONFIGURATION.



To recover from this fault:

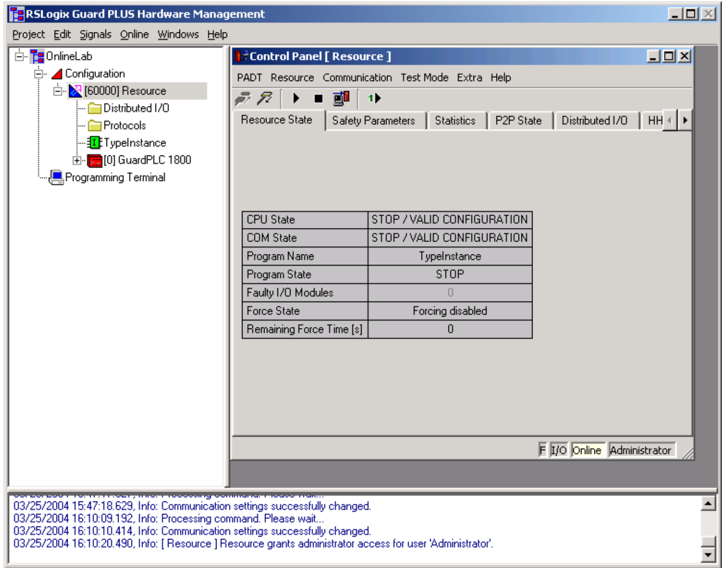
1. Select *Extra →Change System ID* from the Control Panel.

2. Verify that 60000 appears in the first window with 0 in the second, as shown below.



3. Click *OK*.

The Fault LED should turn off, and the Control Panel should show that the CPU State has changed to STOP/VALID CONFIGURATION, as shown below.



TIP The Online indicator is white because the GuardPLC is in STOP/VALID mode.

TIP This is the end of the steps related to the flowchart on page 4-5.

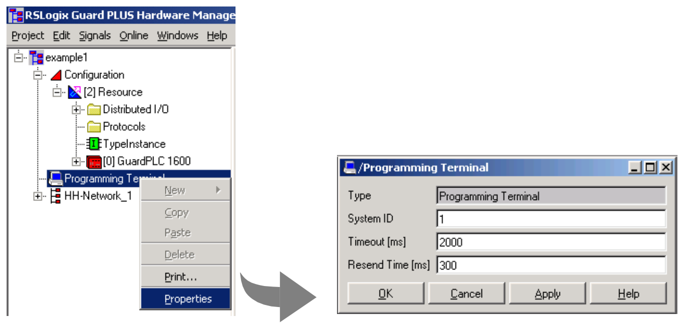
Configuring the Programming Terminal

Specify Host SRS

From the Hardware Management window, you can specify the host SRS of the programming terminal.

1. Right-click on *Programming Terminal* and select *Properties*.

2. Enter the host SRS (1 to 65535) for the programming terminal.



Make sure the host SRS of the programming terminal is not identical to the system ID (SRS) of any other controllers or programming terminals.

In a network, as many as five programming terminals can connect to the same controller at the same time. However, only one programming terminal can have read/write access.

If another controller logs in with read access, that additional user can query controller states and parameters (RUN, STOP, controller switches, etc.) with the Control Panel. The additional user can also display data values if the programming terminal has the same configuration as the controller.

If there are multiple programming terminals in one network, each programming terminal must have a unique host SRS.

Login Dialog

The Login dialog defines the communication parameters between the controller and the programming terminal.

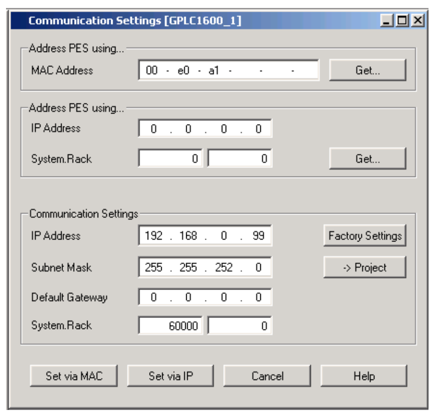
Field:	**Description:**
IP address	the IP address of the controller on the Ethernet network. The factory-set IP address is 192.168.0.99.
SRS	SRS stands for **"S**ystem, **R**ack, **S**lot". The rack and slot IDs are already preset by the controller, so you only need to enter the system ID. You can enter any number from 1 to 65,535. However, the number must be unique from the programming terminal and from any other GuardPLCs on the same Peer-to-Peer Ethernet. The default (factory-set) SRS is 60000.
Username (default = Administrator)	your username. The Administrator assigns a username. The username is sensitive to upper and lower case characters. A username can only contain letters, numbers, and underscore characters. You can define as many as 10 usernames per GuardPLC controller.
Password (default = <blank>)	your password. An Administrator assigns a password. The password is case sensitive. A password can only contain letters, numbers, and underscore characters.
Access Type	your access level. Login as one of these options: Administrator: highest privileges manage usernames and passwords read data from controller write routines and data into controller force tags stop, start, freeze, and force a routine download an operating system change IP address and system ID reboot the controller can also login under read/write and read levels Read/Write: read data from controller write routines and data into controller force tags start, stop, freeze, and force a routine can also login under read level Read: lowest privileges only read data from controller As many as five users can login to the same controller at the same time; however, only one of those users can login as Administrator or Read/Write. The others must login with READ access. If you login while someone else is logged in with Administrator or Read/Write access, you automatically get READ access, regardless of the access type you select. For new controllers, and if the backup battery was removed from a GuardPLC 1200 or 2000 controller, access is available using the following system defaults: Username: Administrator Password <blank> Access Type Administrator

Determining the IP Address and SRS of the Controller

The default IP address of a new controller is 192.168.0.99. The default SRS of a new controller is 60000. To check the current IP address and SRS of a controller:

1. Select *Online →Communication Settings*.

2. In the *MAC address* field, enter the MAC address of the controller. The MAC address is on the sticker on the side of a GuardPLC 1200 controller, on the label positioned over both lower RJ-45 connections on GuardPLC 1600/1800 controllers and I/O, or on the front bezel of the AB-CPU module of a GuardPLC 2000 controller.



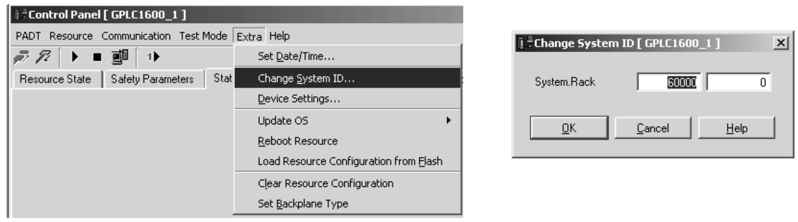
3. Click *Get*. The controller responds back with the IP address and the SRS it is currently using.

TIP	This function works if the IP address of the GuardPLC passes through the subnet mask of your computer.

Changing the SRS of the Controller

There are two ways to change the SRS of the controller:

1. From the Control Panel, select *Extra →Change System ID* (SRS). Enter the SRS and click *OK*.



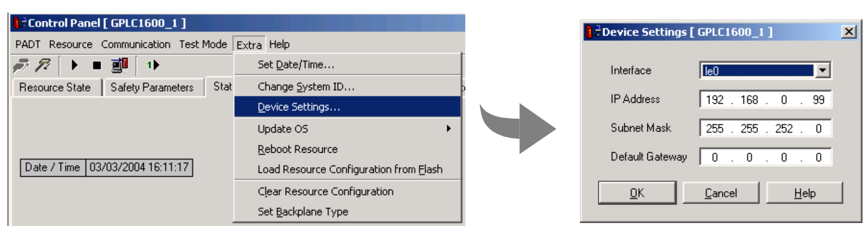
2. Or follow "Step 4: Communication Settings" on page 4-9 and "Step 5: Change Settings via MAC Address" on page 4-10.

TIP Typically, you change the SRS of the GuardPLC to match that of the controller/routine that you wish to download to it. Recall that the SRS is compiled into the executable and ensures that this .EXE can only be downloaded to a GuardPLC with a matching SRS.

Changing the IP Address of the Controller

After you establish communications with the controller, you can change the IP address of the controller to match your Ethernet network. There are two ways to change the IP address of the controller:

1. From the Control Panel, select *Extra →Device Settings*. Enter the new IP address and click *OK*.



2. Or follow "Step 4: Communication Settings" on page 4-9 and "Step 5: Change Settings via MAC Address" on page 4-10.

TIP To re-establish communications with the "new" IP address and subnet of your GuardPLC, you may need to change the IP and subnet address of your programming terminal. Use the Network section of the Windows Control Panel to change the programming terminal's IP address and subnet mask.

Creating Your First GuardPLC Project

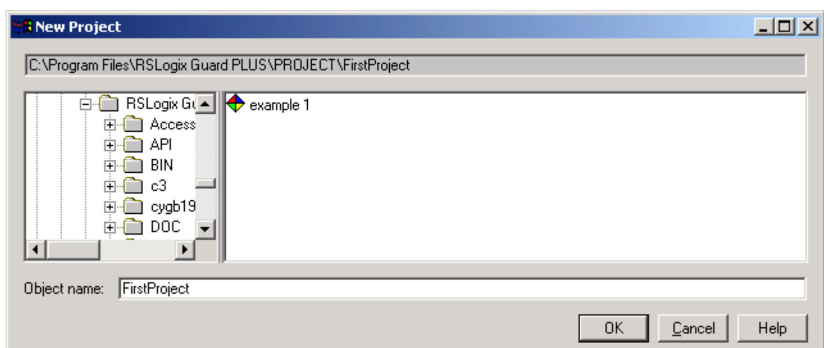
Using This Chapter

This chapter is a tutorial that guides you through the following basic steps required to create a project:

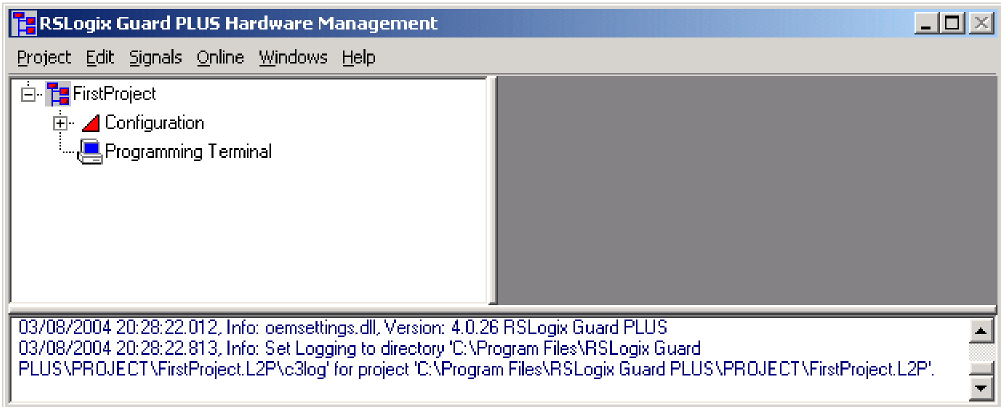
1. Start a new project.
2. Configure the project and hardware.
3. Create signals and connect them to the I/O points.
4. Create a Function Block program using the signals.
5. Save, compile, test, and download the program to the GuardPLC.
6. Monitor the project online.

Start a New Project

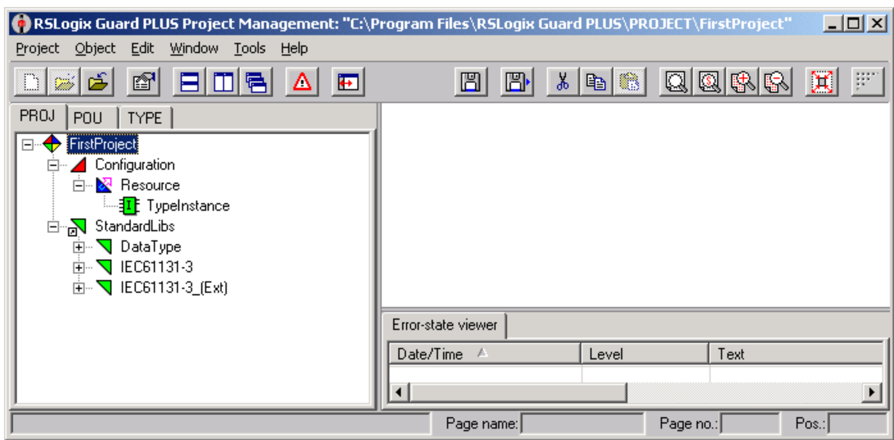
Start RSLogix Guard PLUS. Create a new project using the *New* icon or by selecting *Project* →*New*. Enter "FirstProject" in the *Object Name* field as shown below and click *OK*.



The Hardware Management window opens. This window is used to configure the project, controller, I/O, and signals.



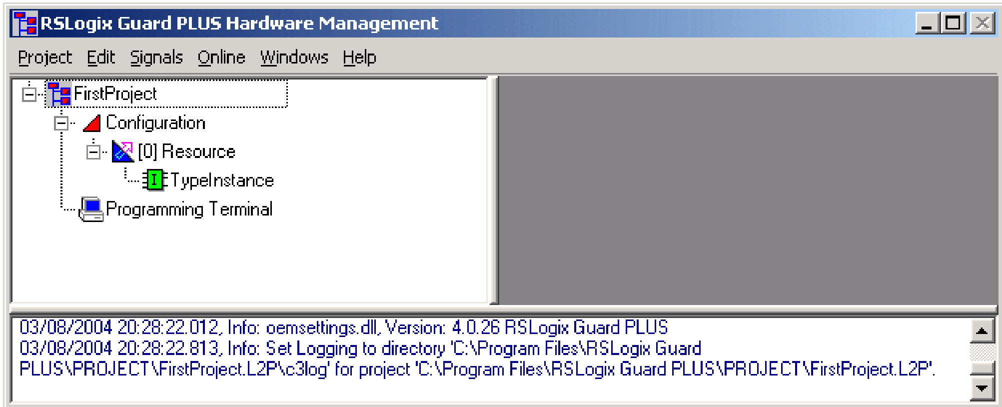
Return to the Project Management window and expand the project tree until it matches the example below.



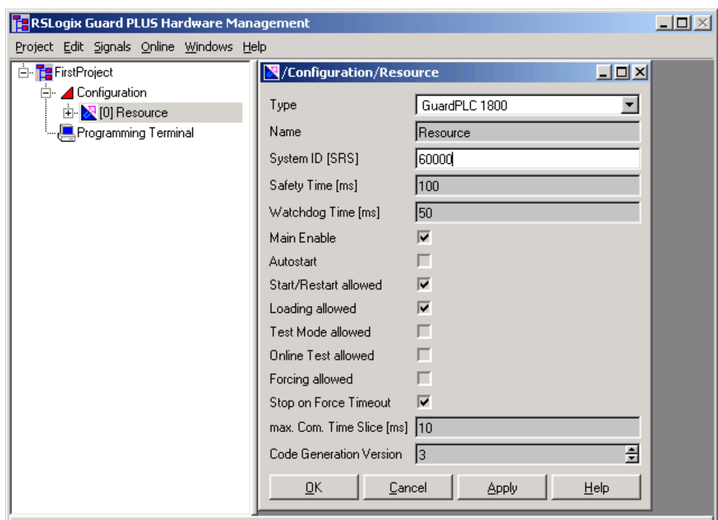
When the project is expanded, you can see that it contains a *Configuration* and under the *Configuration* there is a *Resource*, which is the actual GuardPLC Controller. Under the *Resource* is the program *TypeInstance* that will run on the GuardPLC Controller.

Configure the Project and Hardware

1. In the Hardware Management window, expand the project tree so that the *Configuration*, *Resource*, and *TypeInstance* are visible, as shown below:



2. Right click on *[0] Resource* and select *Properties*. Edit the properties as shown below.



The Resource Type should match the type of GuardPLC to which you want to connect. The SRS is a code that is compiled with the function block routine. The routine can be downloaded only to a GuardPLC with a matching SRS code stored in its non-volatile memory. For more information on configuring the controller, see page 8-5.

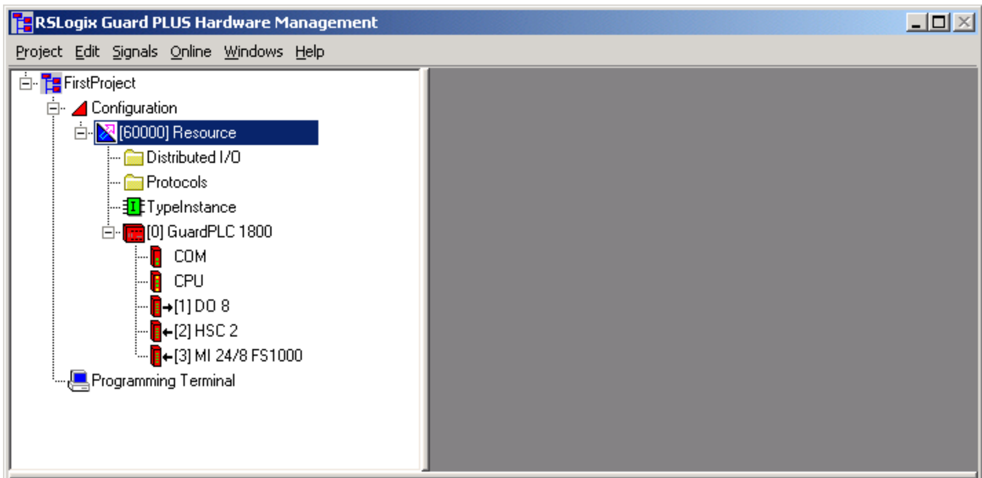
TIP	The default SRS of a new controller is 60000. You must use this SRS to initially establish communications with the controller. Once you have established communications, you can change the SRS.

3. Click *Apply* to move these values into the project.

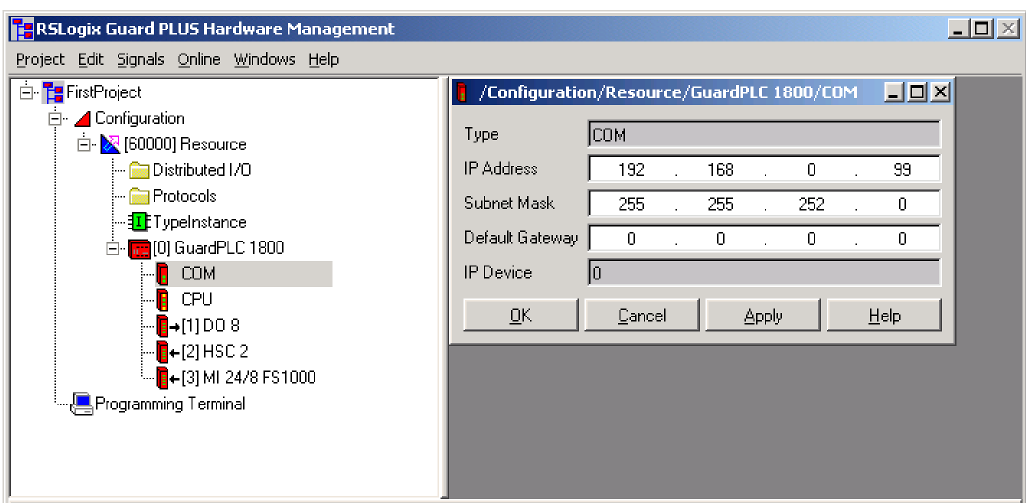
4. Select the four (4) unchecked boxes and click *OK*.

TIP You can rename the controller using the Program Management window. Expand the project tree and the *Configuration*. Right-click on *Resource* and select *Rename*.

5. Expand the *Resource* so that the Hardware Management project tree appears as shown below.



6. Right click on *COM* under *GuardPLC 1800*, and select *Properties*. The following window appears.



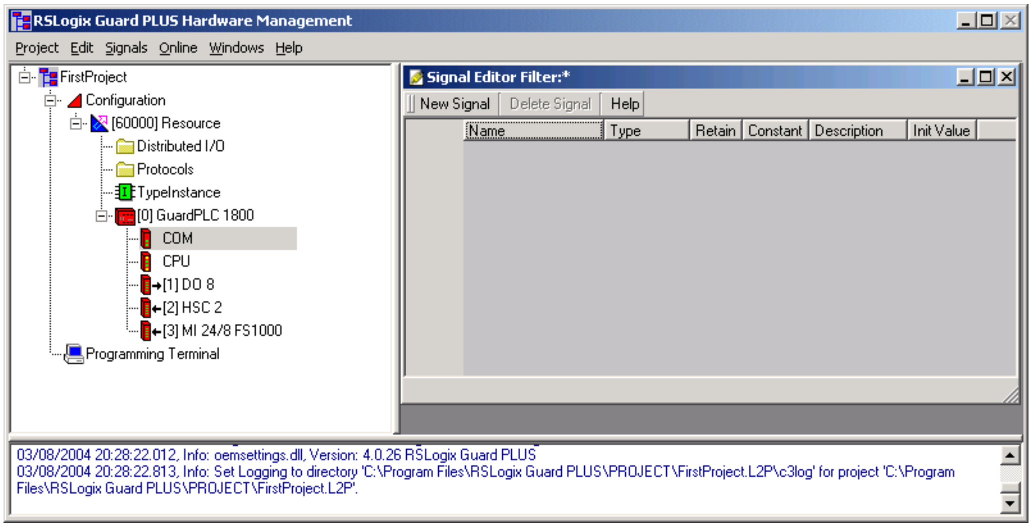
7. Enter the IP address of your GuardPLC. Neither the Subnet Mask nor the Default Gateway should require changes. Click *OK*.

TIP The GuardPLC controller's default IP address is 192.168.0.99.

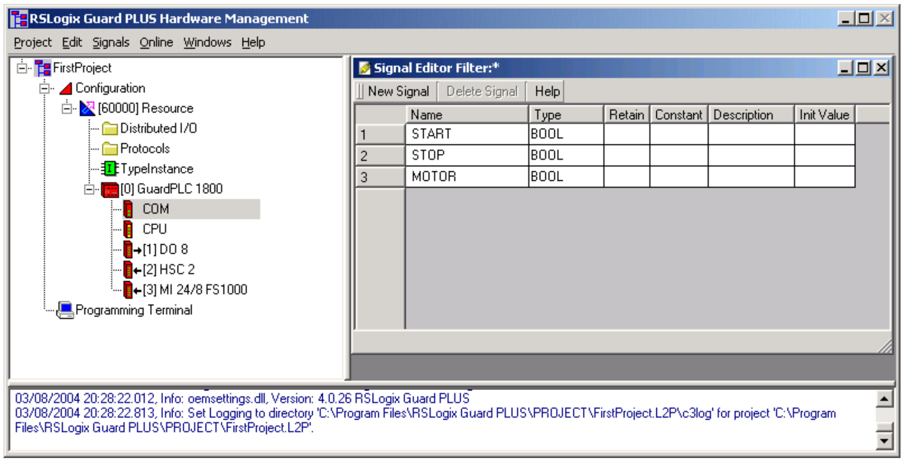
Create Signals and Connect Them to I/O Points

Because the example in this chapter uses the GuardPLC 1800 controller, there are predefined I/O listed under the controller in the project tree. The 1200/1600/1800 are fixed controllers with pre-configured I/O. If you use a GuardPLC 2000, the I/O must be configured, since it is a modular controller.

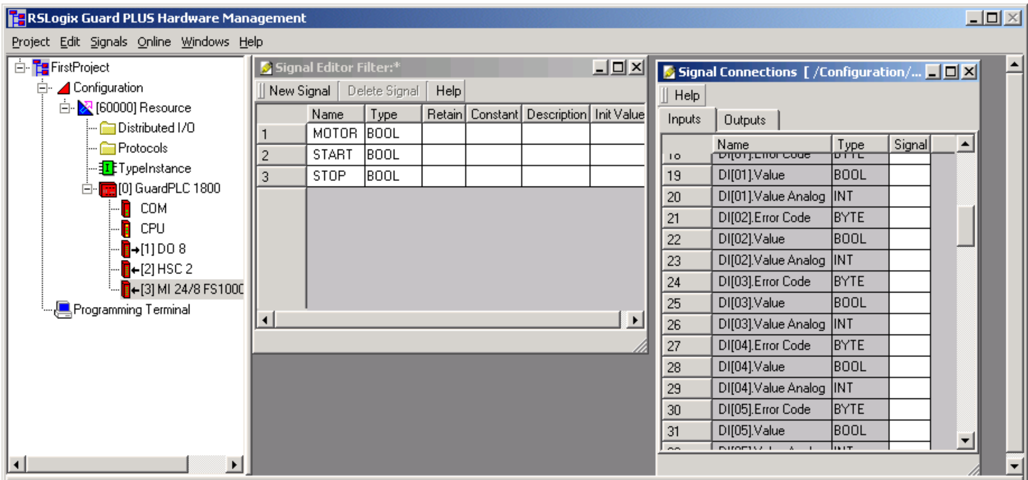
1. Select *Signals* from the Hardware Management menu bar, and select *Editor*.



2. Create 3 new signals, START, STOP, and MOTOR:
   a. Left-click on *New Signal* in the Signal Editor. Type START in the *Name* field and press the Enter key.
   b. Left-click on *New Signal* again. Type STOP in the *Name* field and press the Enter key.
   c. Left Click on *New Signal* again. Type MOTOR in the *Name* field and press the Enter key.



3. Connect these three signals to physical I/O terminals. Right-click on the controller's I/O (*DI 20* for 1600 or *MI 24/8 FS1000* for 1800) and select *Connect Signals*.



TIP Set up your screen so that you can easily drag signals from the Signal Editor window to the Signal Connections window. Both the *Name* fields in the Signal Editor and the *Signal* fields in the Signal Connections window must be visible, as shown above.

4. Verify that the *Inputs* tab is selected on the Signal Connections window.

   Two signals exist for each input: Value and Error Code. The GuardPLC 1800 adds another signal called Value Analog.

   Error Code is a status signal that can be used for point-level diagnostics. The Value contains the actual field state of the input: ON (1) or OFF (0).

5. Connect the START and STOP signals to *DI[17].Value* and *DI[18].Value* by dragging START and STOP from the *Name* field in the Signal Editor to the *Signal* field in the Signal Connections window.

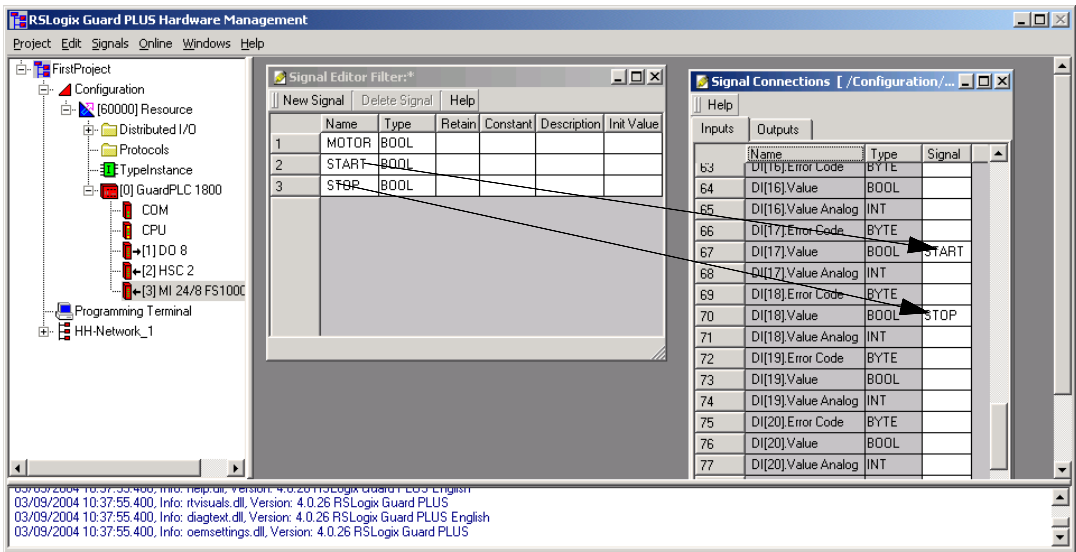
   a. Make sure the cursor is not active in any field in either the Signal Editor or the Signal Connections window.

   b. Left-click and hold on the *Name* field. Drag the signal to the *Signal* field in the Signal Connections window.

c. Release when over the proper field.

TIP Signals can only be dragged and dropped onto *Signal* fields of the same data type. Dropping a BOOL signal onto a BYTE field is not permitted.

When both signals have been connected, the screens should appear as follows:

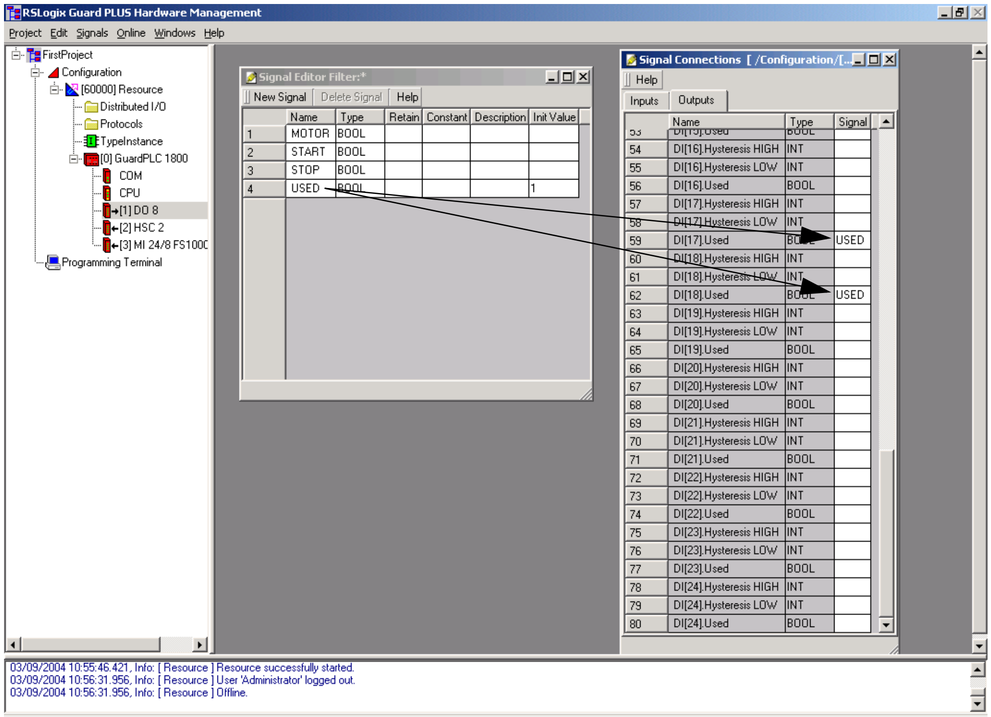


6. If your controller is a GuardPLC 1800, an additional step is required. The digital inputs on a GuardPLC 1800 are actually analog circuits with a resolution of one (1) bit. Any voltage greater than 13V dc will be a 1. Any voltage less than 7V dc will be a 0. Since GuardPLC analog circuits require the user to specify which channels are being used, this is also required for the 24 digital inputs on the GuardPLC 1800.

   a. Add a new signal, called USED, to the Signal Editor. Give this signal an initial value of 1. You will never change this value in your program, so USED will always be 1.

   b. Select the *Outputs* tab of *MI 24/8 FS1000*.

   c. Connect USED to the DI channels being used: *DI[17].Used,* and *DI[18].Used,* as shown on the following page.

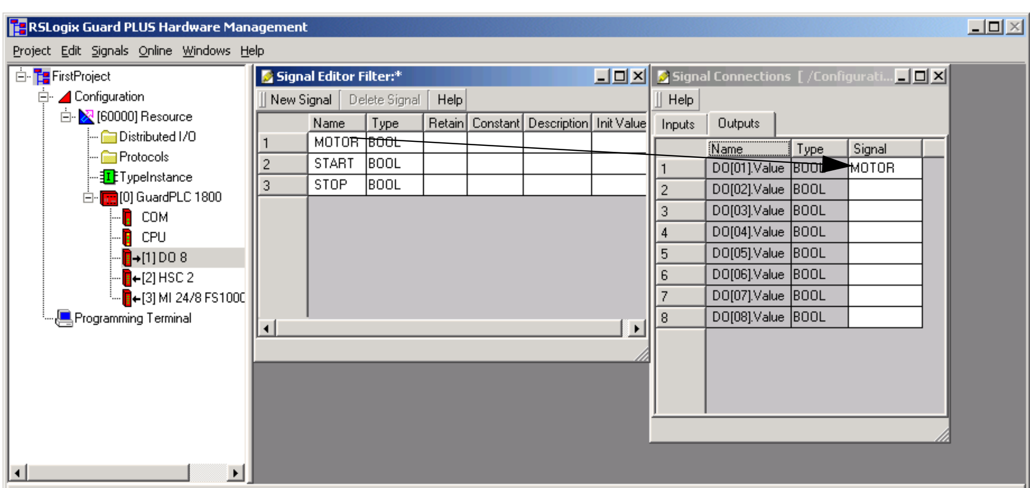


7. Close the DI Signal Connections window.

8. Right-click on *DO8* in the project tree and select *Connect Signals*.

9. The Signal Connections window defaults to the *Inputs* tab. Select the *Outputs* tab to view the output fields.

10. Connect the MOTOR signal to the first output, as shown below.



The signals have now been connected to physical I/O points.

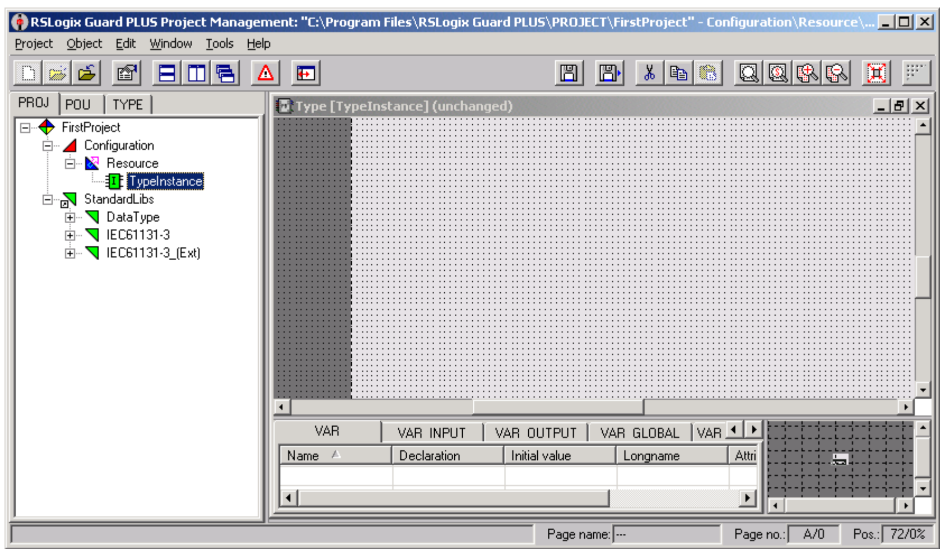
Create a Function Block Program

The following example creates code to start and stop a motor using the two input signals we created earlier.

TIP For more information on Function Block programming, consult the online Help and Chapter 17, "Creating User-Defined Function Blocks".

1. Close the Signal Connections window. Leave the Signal Editor active, and restore the Project Management window.

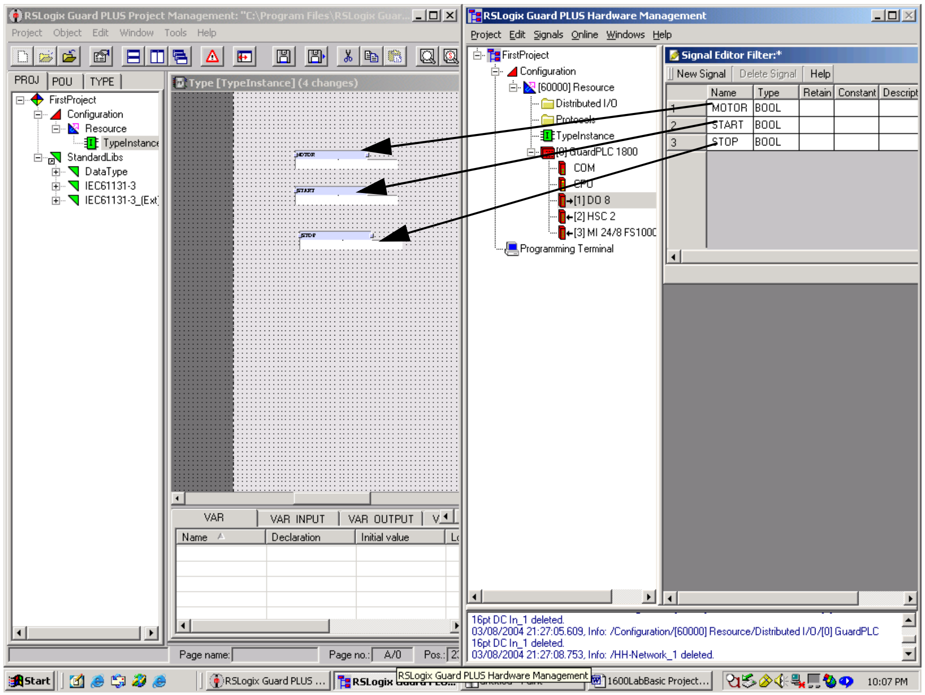
2. If necessary, expand the project tree in Project Management until *[I] TypeInstance* is visible and double left-click on *[I] TypeInstance*. A Function Block Editor program page opens.



1. Drag all three tags from the Signal Editor (in Hardware Management) to any location on the FB Editor program page.

TIP To make the Signal Editor and the FB Editor fit comfortably on your screen, restore both the Project Management and Hardware Management windows. Then, select *Tile Windows Vertically* from the Windows task bar located on the bottom of your screen.

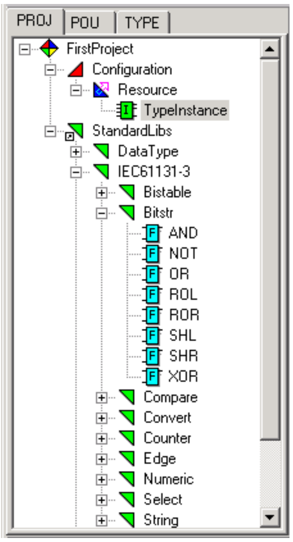
Whenever a page is edited for the first time, a window appears asking for a page name. You do not need to name the page. Click *OK* to close this dialog box.



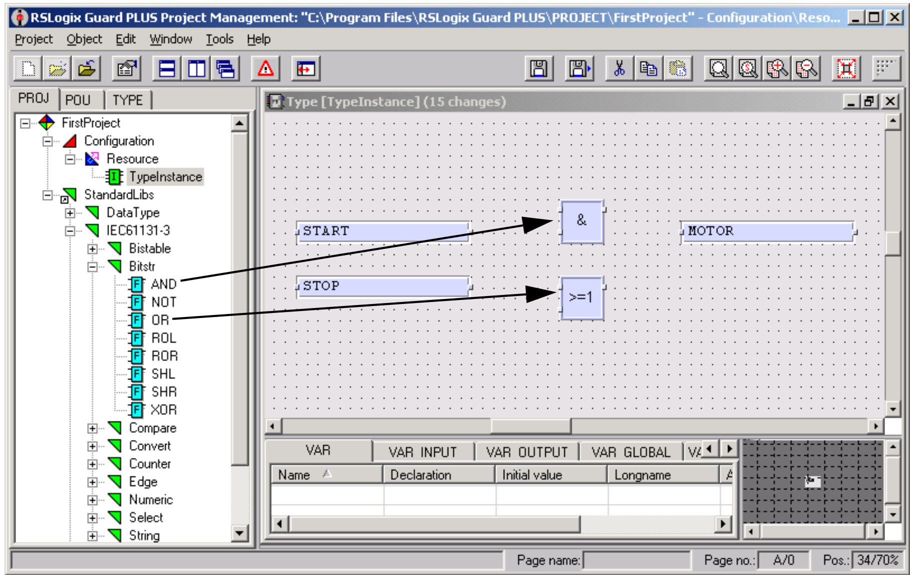
2. Minimize the Hardware Management window.

TIP You can delete the white areas under the signals, which are used for descriptions, by clicking on the white area and hitting the Delete key.

3. In the Project Management project tree, expand *StandardLibs*, *IEC61131-3*, and *Bistr* as shown below.



4. Drag an *AND* and an *OR* block onto the routine (Left-click, hold, drag and release).

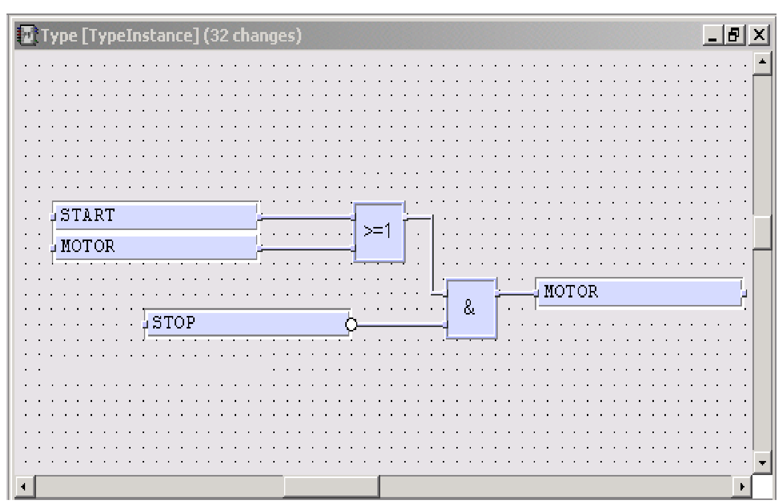


5. Connect the blocks with lines by left-clicking and holding the very small dots on the edges of the boxes. Drag and release over the destination dot.

TIP Use the *Zoom In* tool on the toolbar to zoom in to see the dots on the edge of the boxes.

6. Create a duplicate MOTOR signal by right-clicking on MOTOR and selecting *Duplicate*. Drag and drop the signal on the page.

7. Invert the STOP signal by right-clicking on the dot and selecting *Invert*.



Save, Compile, Test, and Download the Program

Save the Program

Left-click on the *Save* button to save your program edits. A window appears, which you can use to document your changes. Select *OK*.

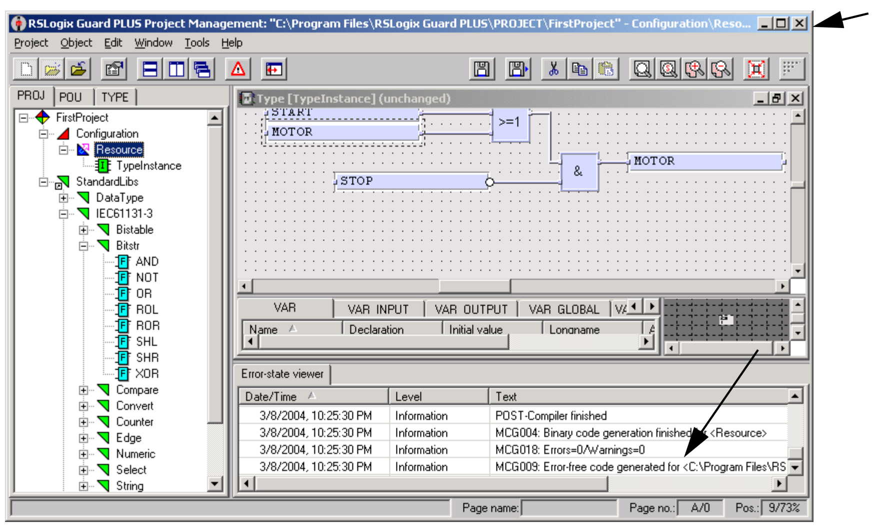
TIP The FB editor menu bar displays the number of edits since the last save. Following a save, it displays "(unchanged)".

Compile the Code

1. Close the *Type Instance* Program. (See the arrow below.)

2. Right-click on *Resource* and select *Code Generation.*

3. The results of the code generation are shown on the Error State Viewer. If the Error State Viewer is not visible, click on the red triangle to make it visible.



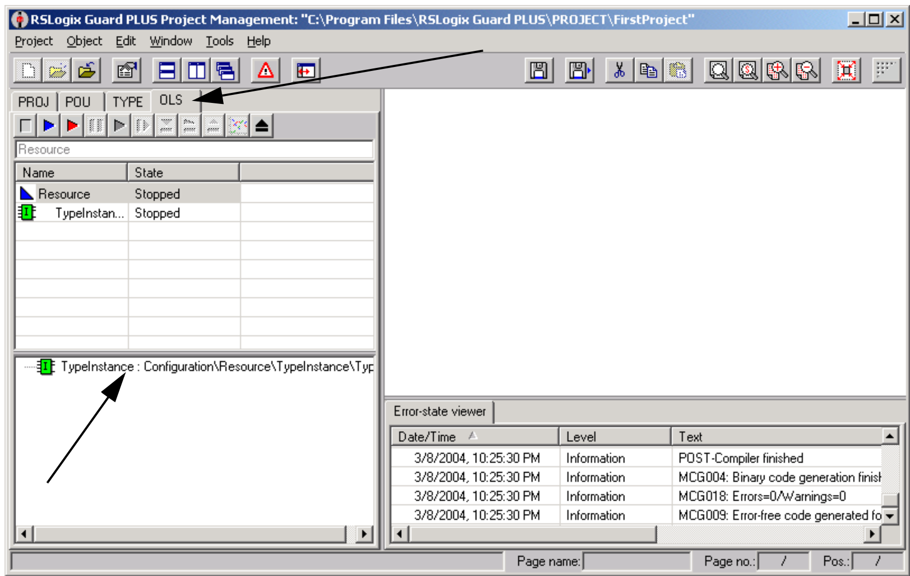
If the compile was successful, "Error Free code generated" appears in the Error-State Viewer.

TIP If you are using a GuardPLC 1800, you will see a warning in the Error State Viewer. Go to the Hardware Management window to view the warning, which reads "'USED' has an initial value, but no source." Disregard this warning, because the "USED" signal has an initial value of 1, but no source drives its value.

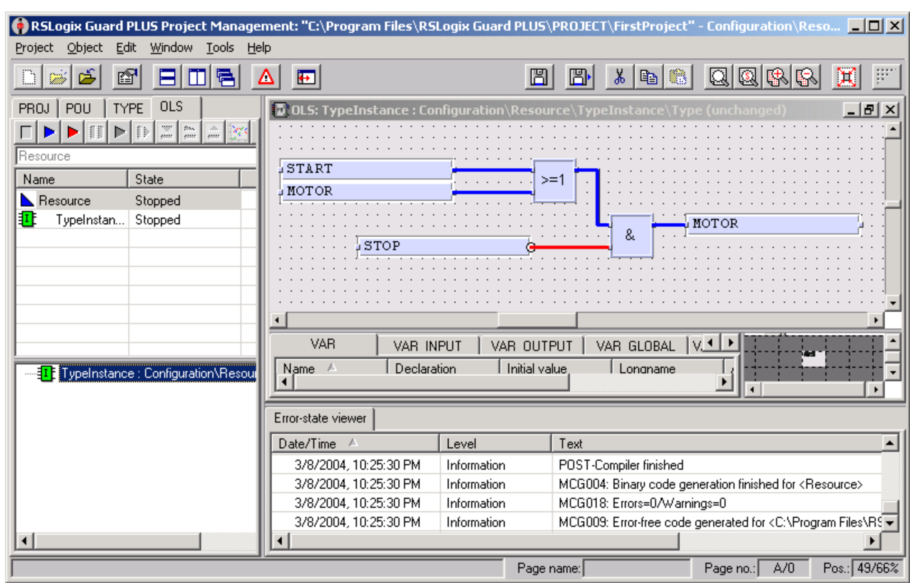
Run an Off-Line Simulation

To test the code before downloading it to the GuardPLC, run an off-line simulation:

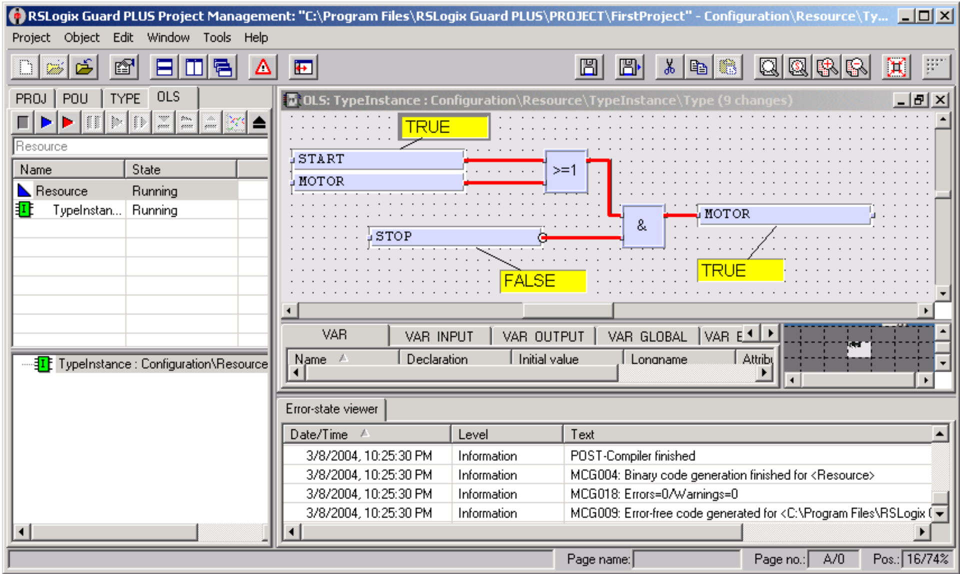
1. Right-click on *Resource* and select *OFF-Line-Simulation*. The *OLS* tab appears.



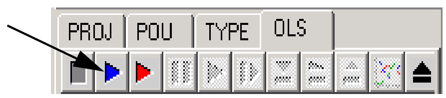
2. Double left-click on *[I] TypeInstance* above.



3. Select points to toggle/view during the simulation. To select a point, left-click on a point, drag, release, and left-click again to activate.

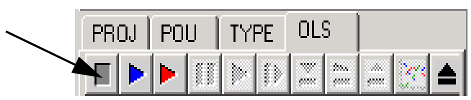


4. Start the simulation by left clicking on the blue flag button.

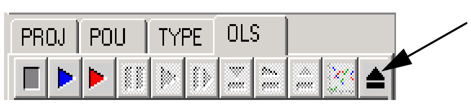


5. Double left-click on the yellow field to toggle TRUE/FALSE. Blue lines represent OFF. Red lines represent ON.

6. When finished testing, stop the simulation by selecting the *Stop* icon.



7. Close the Off-line simulation using the *Close OLS* icon.



TIP If you do not save your changes, you will have to re-select the points to simulate.

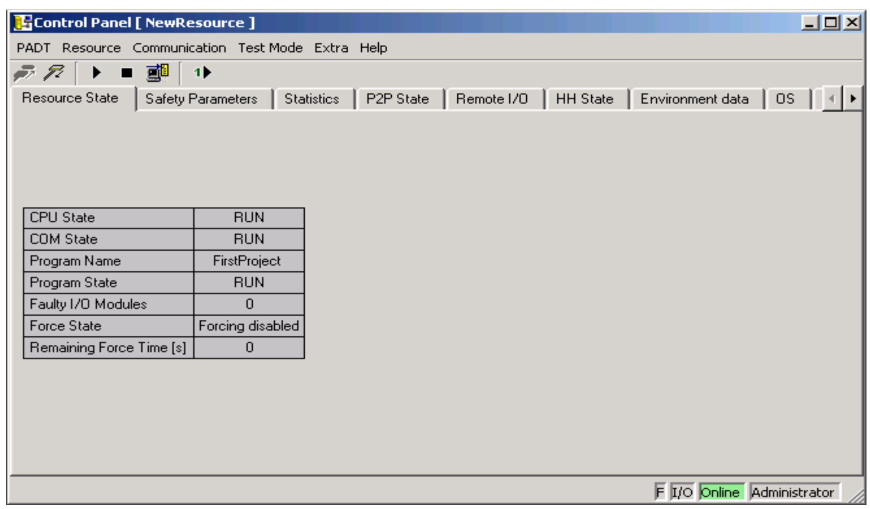
8. Click on the *PROJ* tab to return to the project tree.

Download the Program

1. Connect the GuardPLC to your PC's Ethernet port using a Cat. 5 Ethernet cable.
2. In the Hardware Management window, close the Signal Editor.
3. Right-click on *[60000] Resource*.
4. Select *Online* → *Control Panel*.
5. The default Username is Administrator with no password. Click *OK*. The Control Panel opens.

TIP You can use the [Ctrl]+[A] shortcut to enter the default Username and Password.

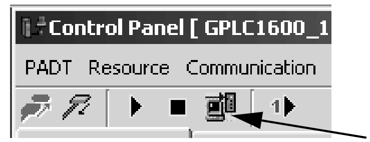
If you are unable to go online, see Chapter 4 for information on determining the IP address and SRS of the GuardPLC and for information on the appropriate setting for your PC's IP address.



6. If the GuardPLC is in RUN mode, change to STOP mode. Left-click on the *Stop* icon on the Control Panel.

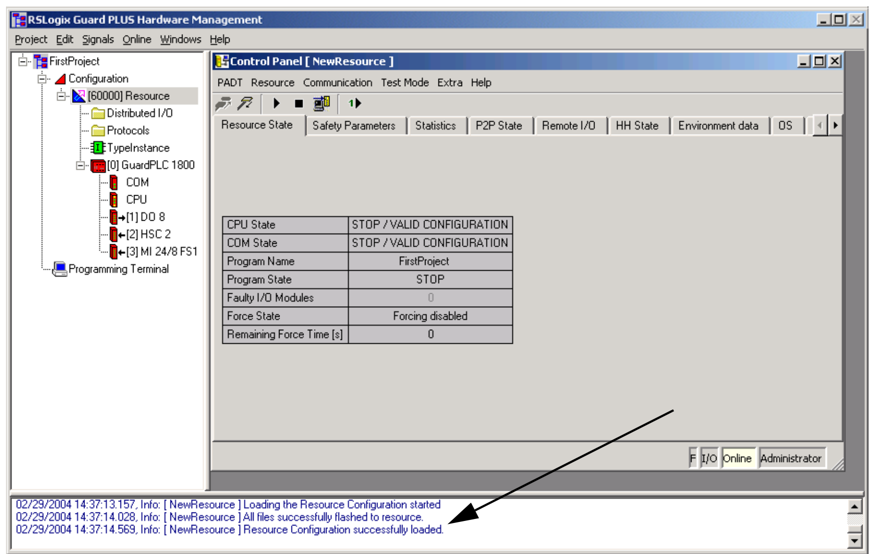
Answer *Yes* to the warning prompt.

7. Left-click on the *Download* icon.



Answer *Yes* to the warning prompt.

8. Make sure the download was successful by checking the Status Field for a "Resource Configuration successfully loaded" message.

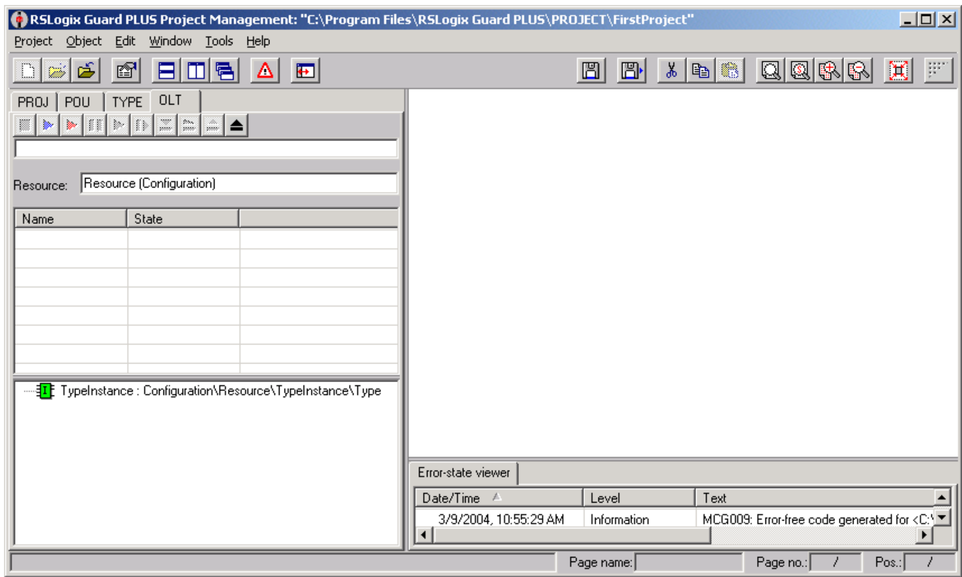


9. Put the GuardPLC into RUN mode by clicking on the *Coldstart* button and answering *Yes* to the warning prompt.

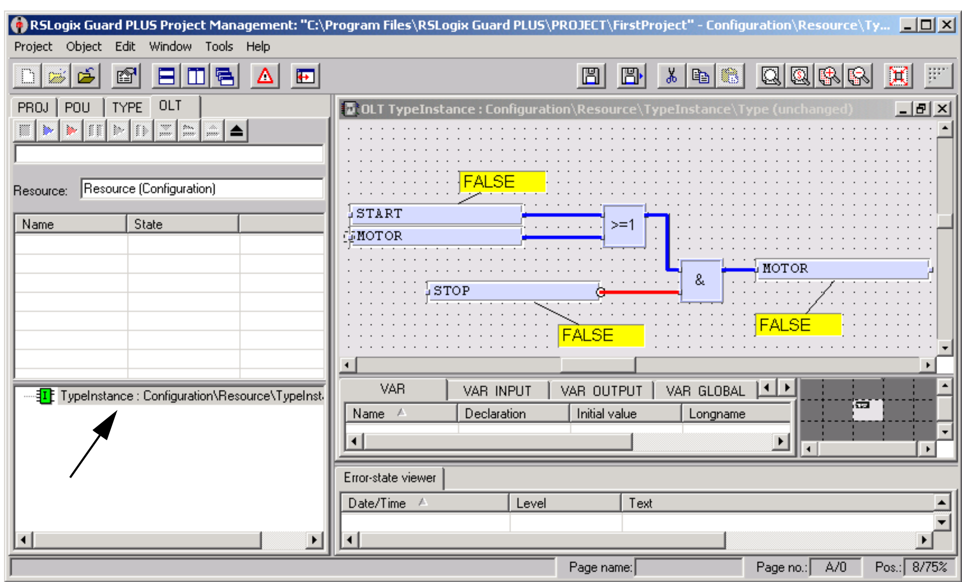
How to Monitor the Routine Online

To monitor the routine online, you must be online with the controller, and the controller must be in RUN mode.

1. In the Project Management window, right-click on *Resource* and select *ON-Line Test*. The Project Manager appears as shown below.



2. Double left-click on *[I] TypeInstance*.



If the lines appear RED and BLUE, then the monitor is active. Test the routine and monitor the function code.

3. Close the On-Line Test when finished testing.

4. Select the *PROJ* tab to return to the project tree.

If the lines are RED/BLACK striped, then the Control Panel is NOT online with the GuardPLC or the GuardPLC is not in RUN mode.

See Chapter 4 for information on going online with the GuardPLC controller.

Check, Download, Start, and Test a Routine

Using This Chapter

To download and run a routine, you must first:

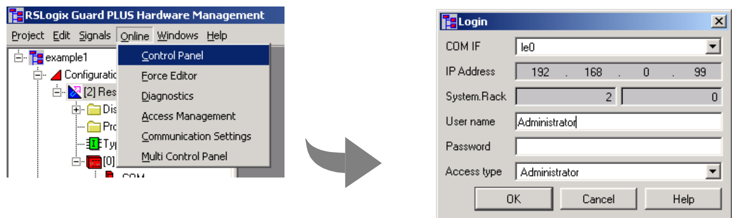
1. Complete your system configuration and your routine logic.
2. Save your logic by selecting *Object →Save* on the Project Management menu bar.
3. Generate code. Make sure all your system configuration is complete before you generate code.
4. Connect the programming terminal (running RSLogix Guard PLUS software) to the controller. See Chapter 4 for information on programming over an Ethernet link.
5. Download the routine to the controller. See page 6-2.
6. Start the routine. See page 6-4.

Checking Consistency

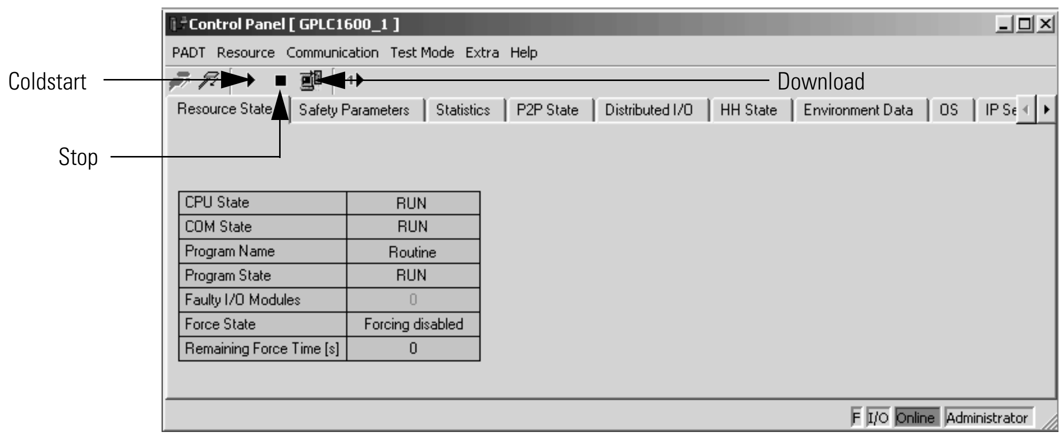
To determine whether or not you need to download your routine, you can use the Check Consistency feature to verify whether the routine running in the controller is the same routine you are editing in RSLogix Guard PLUS. Select *Resource → Check Consistency* to compare the two programs. If all the codes match, your offline routine has been previously downloaded to the controller.

Downloading a Routine

1. Select *Online* → *Control Panel*. The software automatically asks you to log in.

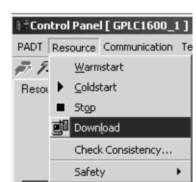


2. After you successfully log in, the Control Panel opens.



3. The routine must be stopped before downloading is permitted. Select *Resource* →*Stop*.

4. Select *Resource* →*Download* to load the routine into the controller.



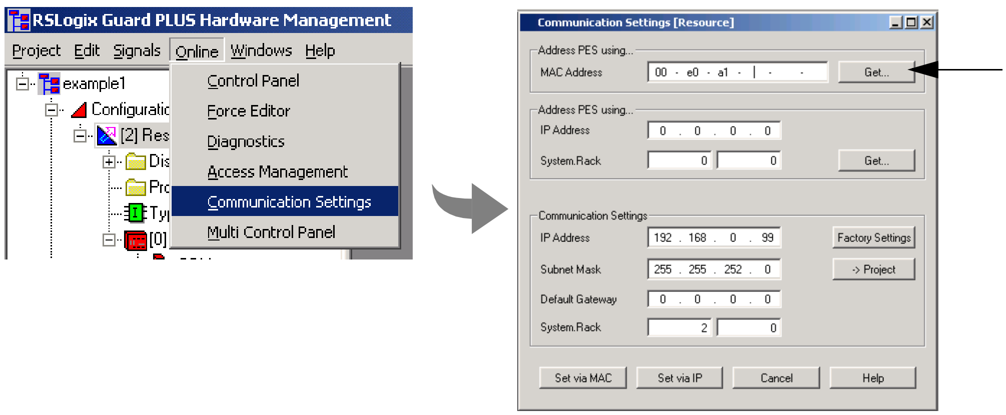
IMPORTANT	If your controller is in FAILURE_STOP, it must be rebooted before you can download a routine. See page 8-4.

Troubleshooting the Download Process

The SRS of the controller must match the SRS saved in the routine in order to download the routine. When you specify an SRS for a controller in a project, that SRS gets saved in the routine when you generate code.

Checking the SRS of the Controller

1. Select *Online →Communication Settings*.



2. In the *MAC address* field, enter the MAC address of the controller. The MAC address is on the sticker on the side of a GuardPLC 1200 controller, on the label positioned over both lower RJ-45 connections on GuardPLC 1600/1800 controllers and I/O, or on the front bezel of the AB-CPU module of a GuardPLC 2000 controller.

3. Click *Get*. The controller responds back with the IP address and the SRS it is currently using.

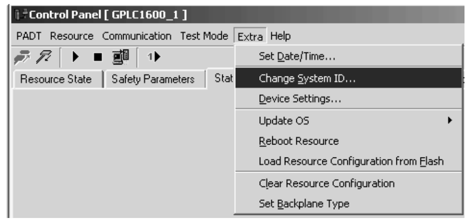
Now you know the correct SRS to use. Change the SRS and generate code again. Then the download should work.

Updating the SRS in the Controller

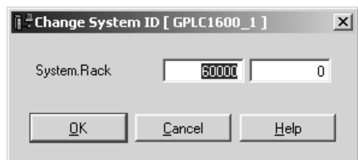
In some cases, most likely after a reboot due to a FAILURE_STOP, the SRS of the controller might be the same as the SRS in the routine, but the routine still will not download. If this happens, change the SRS to the same number and click *OK*, as shown on the following page. This updates the SRS in the controller and corrects the issue. You should now be able to download the routine.

TIP	For more information on recovering from FAILURE_STOP, see page 8-4.

1. Select *Extra →Change System ID* (SRS).



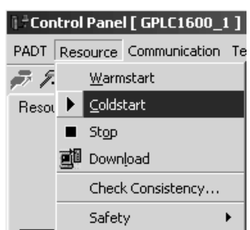
2. Enter the SRS and click OK.



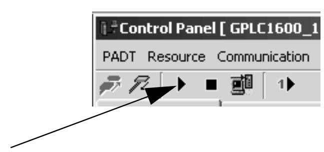
3. Try the download again.

Starting a Routine

After you successfully download a routine, you can start the routine. From the Control Panel, select *Resource →Coldstart* or use the *Coldstart* button on the menu bar.



or

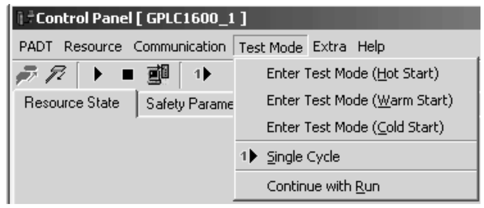


Options:	Description:
Warmstart	Allows the user routine to be started by the programming terminal and to continue with the previously saved Retain signals. You must have Administrator or Read/Write access to initiate a warmstart.
Coldstart	If a routine is in STOP or FREEZE mode, it can be started using this cold start option. The cold start option re-initializes the routine and available process values are lost.
Stop	Use this option to stop a routine that is in RUN or FREEZE mode.

Testing a Routine

Test a routine to check for and eliminate errors. You must have Administrator or Read/Write access to test a routine. Test options are discussed in the table below.

From the *Test Mode* menu on the Control Panel, select the test option you want.



Test Option:	Description:
Enter Test Mode (Hot Start)	To enter test mode hot, a routine must be loaded and started in the controller. After a security query, the routine is paused (FREEZE) while retaining the current process data after terminating the cycle. No input signals are processed. The output signals retain their current state.
Enter Test Mode (Warm Start)	Halts the execution of the routine with the signals declared as Retain retaining their values and with all other signals being reset.
Enter Test Mode (Cold Start)	A routine must be loaded in the controller to allow you to “enter test mode” cold. After a security query, the routine is initialized, started, and immediately enters FREEZE mode. No input signals are processed, and all the output signals stay in their basic state. If the routine was in RUN mode when enter test mode cold was selected, the cycle in progress is terminated and the process data is re-initialized.
Single cycle	Single cycle can only be executed when the controller is in the test mode. Use single cycle to manually trigger the execution of a single cycle of the routine. The routine is executed exactly once. The input signals are read in, processed, and the resulting output signals are transferred. Use the force editor to perform a step-by-step check of the data. See Chapter 9 for information on monitoring signals.
Continue with Run	This option terminates the test mode. The routine mode changes from FREEZE to RUN without re-initialization. The current process data are retained. (This corresponds to a routine hot start.)

How a Routine Executes

A controller has only one routine. A routine can be in any one of these states:

Routine State:	Description:
RUN_RUN	The controller is in the RUN mode. • The routine is executed cyclically by the controller. • Input data are processed in the routine. • Output data of the routine are operated.
RUN_FREEZE	The controller is in the RUN mode. • The routine is not executed. • No input data are processed by the routine. • No output data of the routine are operated. *This mode is not permissible for safety-related operation!*
STOP	The controller is in the STOP mode. • The routine is not (no longer) executed. • All outputs have been reset.
FAILURE_STOP	The controller is in the STOP mode. • The routine was stopped due to an error. • All outputs have been reset.

Controlling a Routine

You can control a routine using the actions described below:

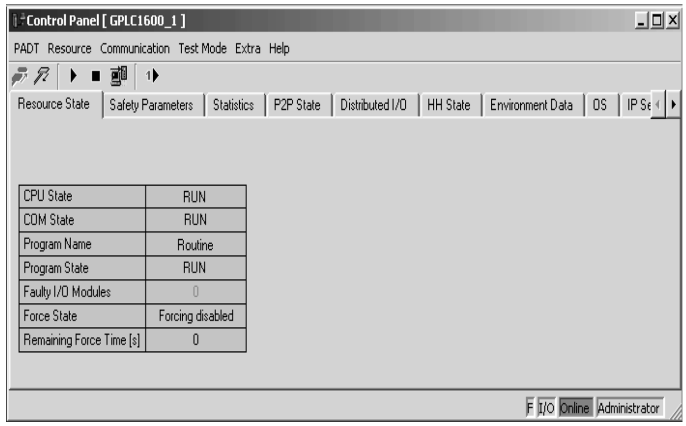
Control Action:	Description:
Start the routine from STOP	Starting the routine is the same as transferring the controller from the STOP mode into the RUN mode. The routine is then transferred into the RUN_RUN mode. If Freezing is activated while starting, the routine will be in the RUN_FREEZE mode. However, freeze operation is only possible if the software switch “Freeze Enable” has been enabled. In addition to starting in freeze mode, cold start is also possible. Starting a routine is only possible when both the controller restart switch and the routine restart switch are enabled.
Start the routine from RUN	The routine is transferred into the RUN_RUN mode if it has not already been operating in this mode. Starting is also possible in cold start, hot start, and no-freeze modes. *This function is not allowed for safety operations of the controller!*
Single cycle the routine	The routine must be in the RUN_FREEZE mode. Exactly one RUN cycle of the routine is executed, and the routine is then put back into the RUN_FREEZE mode. The command for the single cycle is the start command with the attributes hot start and freeze. This does not have any effect on the mode of the controller. Single cycle is only performed by the controller for the routine if freeze mode is enabled. *This function is not allowed for safety operations of the controller!*
Restart the routine	If the routine is in the FAILURE_STOP mode, it can be restarted via the programming software using a start command. After the restart, the entire routine is checked again.
Stop the routine	Stopping the routine is the same as transferring the controller from RUN mode into STOP mode. The routine is then transferred from RUN into the STOP mode.
Freeze the routine	The routine is transferred from the RUN_RUN mode into the RUN_FREEZE mode. This does not affect the mode of the controller. Freeze mode must be enabled for the routine. *This function is not allowed for safety operations of the controller!*

Using the Control Panel

Using This Chapter

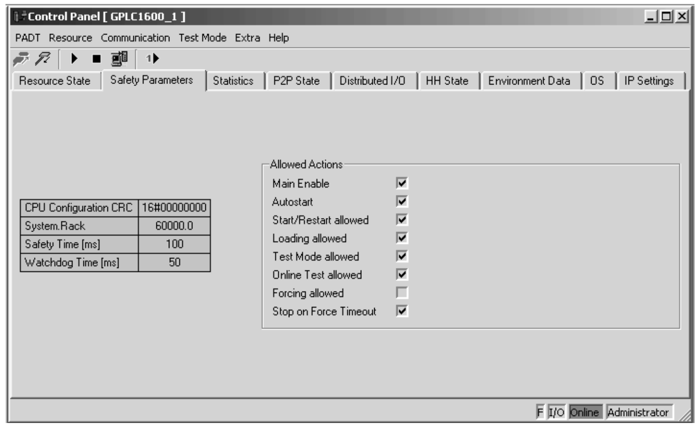
The Control Panel is your window into the online functionality of the controller. Use the tabs to modify or monitor controller status.

Resource State Tab



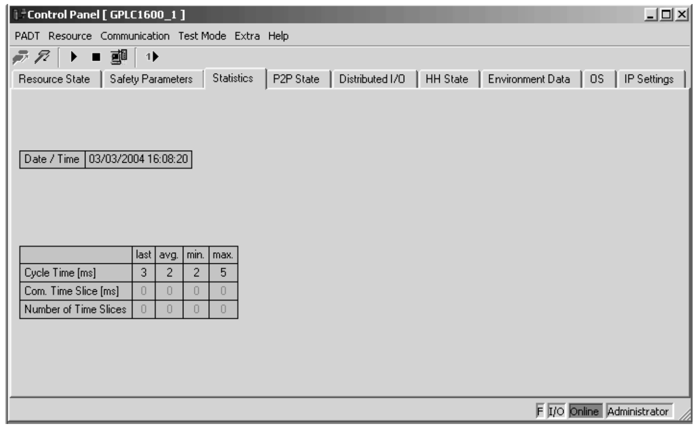
This field:	Displays:
CPU State	the current state of the controller. Possible states are INIT, RUN, STOP/VALID_CONFIGURATION, STOP/INVALID_CONFIGURATION, and FAILURE_STOP. See “Controller Modes” on page 8-2.
COM State	state of the communication portion of the controller. Possible states are RUN, STOP, and OS_LOADING.
Program Name	the routine name. The name assigned by the user to the routine. The default name is “Routine.”
Program State	the current state of the routine. Possible states are RUN, STOP, FREEZE, and FAILURE_STOP. See “Routine Modes”on page 8-8.
Faulty I/O Modules	the number of faulty I/O modules, when the controller is in RUN.
Force State	the force status. 0 – forcing is disabled 1 – ready for forcing (the controller is in stop but is set for forcing) 2 – forcing is active
Remaining Force Time [s]	the remaining force time in seconds.

Safety Parameters Tab



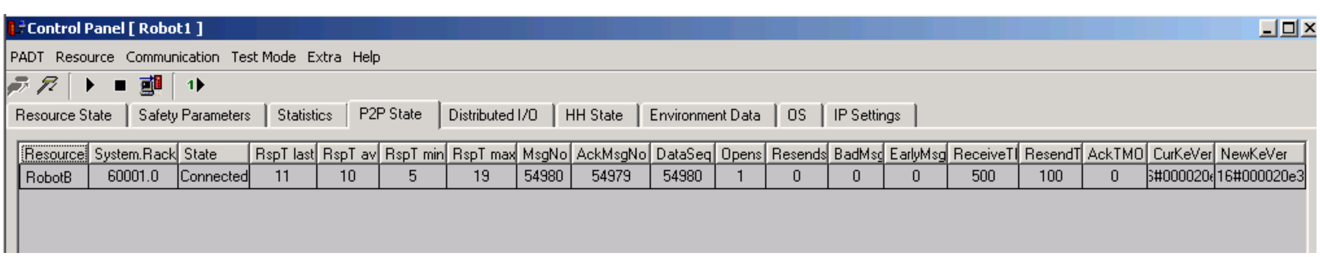
This field:	Displays:
CPU configuration CRC	cyclic redundancy check (CRC) selection for the configuration in the CPU (in hexadecimal notation). This identifies the configuration loaded in the controller.
System ID	the system ID.
Safety Time [ms]	the safety time in milliseconds.
Watchdog Time [ms]	the watchdog time in milliseconds.
Main Enable	whether controller switches can be changed while the controller is executing.
Autostart	whether the controller automatically starts up after rebooting the controller or applying power to the controller.
Start/Restart allowed	whether you can start a controller manually.
Loading allowed	whether you can load new configuration information to the controller.
Test Mode allowed	whether you can freeze the routine.
Forcing allowed	whether you can force tags.
Stop on Force Timeout	whether to stop executing the routine when the force time expires.

Statistics Tab



This field:	Displays:
Cycle Time [ms] average	the average cycle time (in milliseconds) of the last 50 cycles.
Cycle Time [ms] last	the cycle time (in milliseconds) of the last cycle.
Cycle Time [ms] min.	the fastest cycle time (in milliseconds).
Cycle Time [ms] max.	the slowest cycle time (in milliseconds). If this value exceeds the Watchdog Time, the controller goes to FAILURE_STOP.
Com. Time Slice [ms]	the time required to process all Peer-to-Peer communication tasks within a CPU cycle.
Number of Time Slices	the number of time slices required to process all communication tasks. This value should always be 1 to avoid having multiple CPU cycles to complete all communication tasks.
Date/Time	the date and time in the controller.

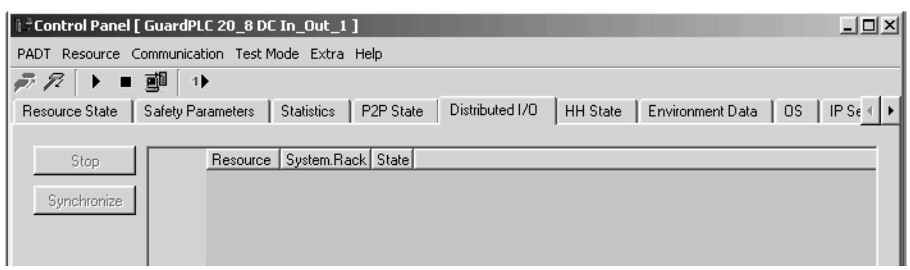
P2P (Peer-to-Peer) State Tab



This field:	Displays:
Resource	the name of the controller.
System ID	the network ID of the controller.
State	the status of the communication.

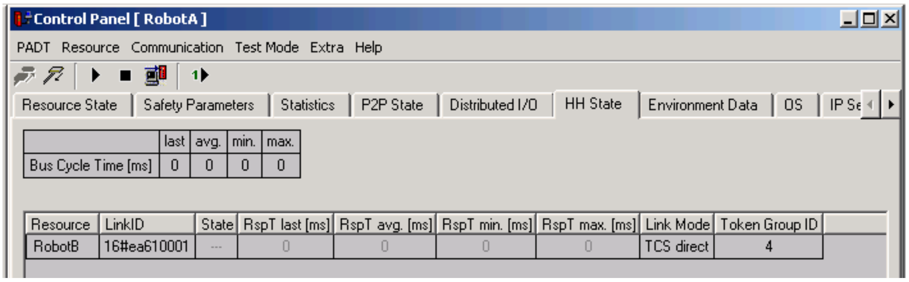
This field:	Displays:
RspT (last, avg, min, max)	the Measured ResponseTime for a message from $PES_1 \rightarrow PES_2 \rightarrow PES_1$, based on the network hardware, CPU cycle time, and Peer-to-Peer profile. This parameter will be optimized later.
MsgNr	the Counter (32-bit resolution) for all messages sent to a controller.
AckMsgNr	the number of the received message that the controller has to acknowledge.
DataSeq	the Counter (16-bit resolution) for sent messages, which contain process data.
Opens	the number of successful connects to a controller. A figure higher than 1 indicates that a controller dropped out and has been reconnected.
Resends	the Counter (32-bit resolution) for messages that have been resent due to an elapsed ResendTMO.
BadMsgs	the Counter (32-bit resolution) for received messages that are corrupted, or are not expected at that instant. A corrupt message, for example, is a message with a wrong sender or with a faulty CRC. An unexpected message, for example, is an "Open" command, when the controllers are already connected.
EarlyMsgs	the Counter (32-bit resolution) for received messages that are not in the correct sequence. If a message drops out and is lost at the addressee, there is a gap in the received messages, and the next message comes early.
Receive Tmo	Receive Timeout as entered by the user.
ResendTMO	Resend Timeout as set by the profile.
AckTmo	Acknowledge Timeout as set by the profile.
CurKeVer	CRC for the Peer-to-Peer configuration. Identical to the Peer-to-Peer system signal.
NewKeVer	Reserved for future use.

Distributed I/O Tab



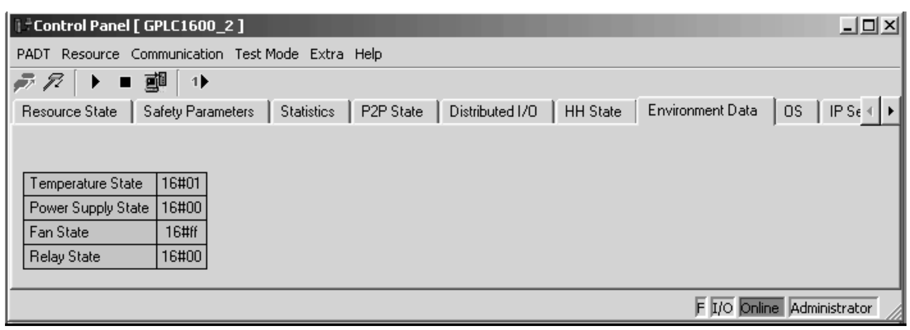
This field:	Displays:
Resource	the name of the module.
System.Rack	the System.Rack ID of the module.
State	the status of the I/O module: • RUN • STOP/VALID_CONFIGURATION • STOP/INVALID_CONFIGURATION • ERROR_STOP • not connected

HH (High-Level High-Speed) State Tab



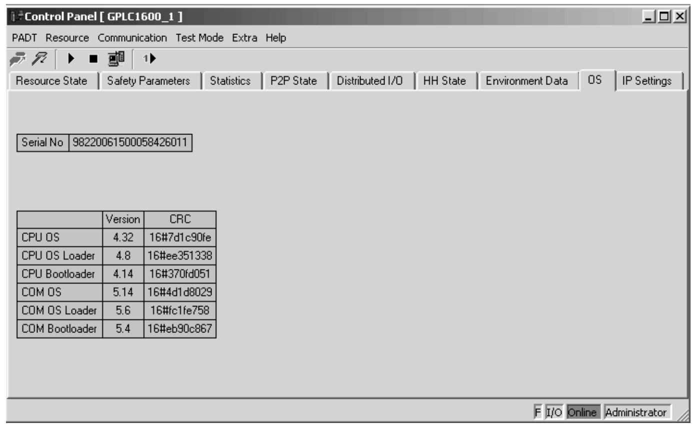
This field:	Displays:
Bus Cycle Time	the time in milliseconds for a Token cycle. The value is 0, if Token Passing is off (any Cleanroom profile).
Resource	the name of the controller.
LinkID	the controller network ID.
State	the status of communication.
RspT	• If Link Mode is "TCS direct" (Token Passing OFF), RspT is the ResponseTime of the HH profile for a message from $PES_1 \rightarrow PES_2 \rightarrow PES_1$, based on the network hardware and topology. This parameter cannot be changed by the user. • If Link Mode is "TCS TOKCYC" (Token Passing ON), RspT is part of the Bus Cycle Time.
Link Mode	• "TCS direct" when Token Passing is OFF. • "TCS TOKCYC" when Token Passing is ON.
Token Group ID	the ID of the Token Group.

Environment Data Tab



This tab displays status messages in hexadecimal form for Temperature State, Power Supply State, Fan State, and Relay State. See "Programming Controller Data" on page B-1 for an explanation of the error bits.

OS Tab



This field:	Displays:
Serial Number	the serial number of the communication module of the controller.
CPU OS	the version of the operating system and the cyclic redundancy check of the operating system (in hexadecimal). (Version 2.4 or later is required for Peer-to-Peer communication.)
CPU Loader	the version of the operating system loader and the cyclic redundancy check of the operating system loader (in hexadecimal).
CPU BootLoader	the version of the boot loader and the cyclic redundancy check of the boot loader (in hexadecimal).
COM OS	the version of the communication operating system and the cyclic redundancy check of the communication operating system (in hexadecimal). (Version 2.4 or later is required for Peer-to-Peer communication.)
COM OS Loader	the version of the communication operating system loader and the cyclic redundancy check of the communication operating system loader (in hexadecimal).
COM BootLoader	the version of the communication boot loader and the cyclic redundancy check of the communication boot loader (in hexadecimal).

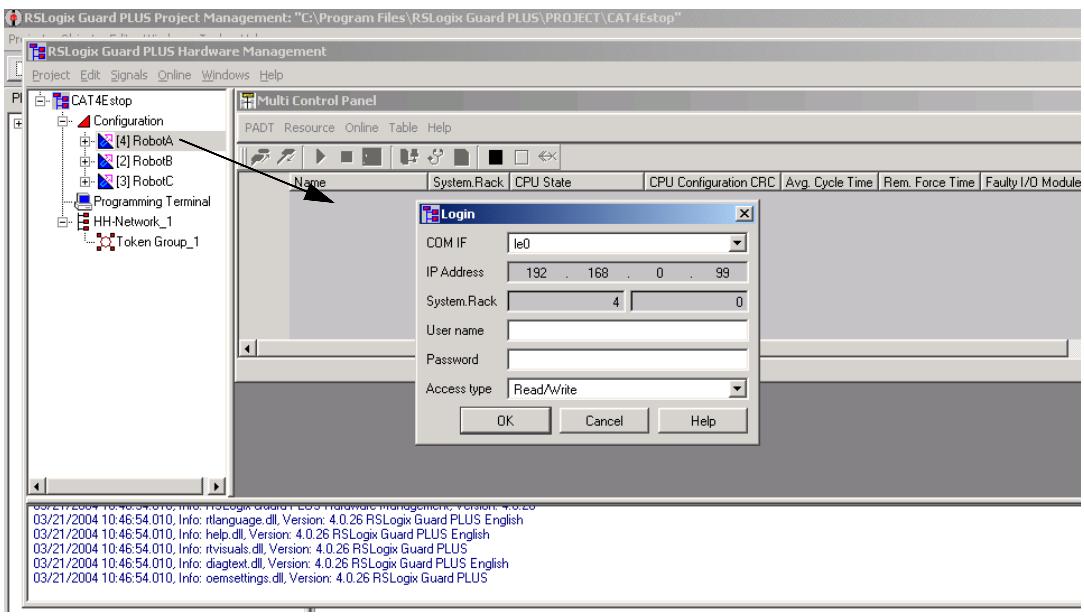
Using the Multi-Control Panel

The Multi Control Panel allows you to connect the programming terminal to more than one controller in the project in one window and to perform actions such as downloads, controller starts, invoking the force editor, etc. simultaneously.

1. Open the Multi Control Panel by selecting *Online →Multi Control Panel.*

   When the Multi Control Panel is opened for the first time, it does not contain any controllers.

2. Add a controller to the Multi Control Panel by dragging and dropping the Resource from the project tree into the Multi Control Panel.

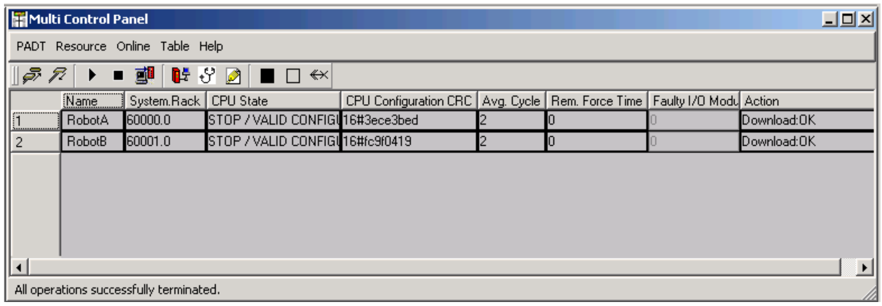


3. After a controller has been dropped in the Multi Control Panel, the Login window opens. Enter the correct Username and Password to connect the controller to the programming terminal.

   You must have Read/Write or Administrator rights (Access type) to download a routine into the controller.

4. Add as many controllers to the Multi Control Panel as you need. The list of controllers in the Multi Control Panel can be sorted by clicking on the column headlines.

The Multi Control Panel displays the following controller information:

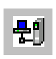
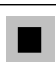
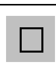


This field:	**Displays:**
Name	Controller name
System.Rack	Controller ID
CPU State	Status of the controller CPU, such as RUN, STOP, STOP/VALID CONFIGURATION, STOP/INVALID CONFIGURATION, etc.
CPU Configuration CRC	Checksum (cyclic redundancy check) of the CPU configuration, displayed in hexadecimal.
Avg. Cycle Time	Average CPU cycle time in milliseconds. This figure depends on the complexity of the logic and, because of the Schedule Time Slice, on the network load.
Rem. Force Time	Remaining force time in seconds (time until forcing is deactivated). Value is "0" when forcing is not active or disabled.
Faulty I/O Modules	Number of faulty IO modules. A fault can result from a hardware malfunction or from incorrect configuration.
Action	Display of a Multi Control Panel command and command status (e.g. Start, Start:Ok). The field is cleared after five seconds.

You can perform a Multi Control Panel command on one or more controllers. To select a single controller, click on the line number left of the controller name. The boundaries of this line become thicker. Hold down the CTRL key and click on another line number to add this controller to your selection. Use the SHIFT key to select controllers from line x to line y. To select all the controllers, use the *Select All* icon  on the tool bar.

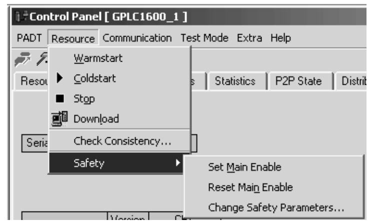
The following commands can be carried out using the Multi Control Panel buttons in the button bar:

Table 7.1 Multi Control Panel Buttons

Button	Command
	Connect Connects the programming software to the selected controller(s) after loss of communication or manual disconnect. After manual disconnect, a new login with password is required.
	Disconnect Disconnects the programming software from the selected controller(s).
	Coldstart Performs a coldstart on the selected controller(s).
	Stop Stops the selected controller(s).
	Download Loads the routine(s) into the selected controller(s). Prior to download, the code generator must have successfully generated program code and the selected controller(s) must be in STOP mode. NOTE: You cannot download a routine into a controller other than the one for which the logic was created.
	Control Panel Starts the control panel for the selected controller(s). This command can be carried out for a single controller by selecting *Online → Control Panel*.
	Diagnostics Starts the diagnostics display for the selected controller(s). This command can be carried out for a single controller by selecting *Online → Diagnostics*.
	Force Editor Starts the force editor for the selected controller(s). This command can be carried out for a single controller by selecting *Online → Force Editor*.
	Select All Selects all controllers in the list.
	Deselect Deselects marked controllers.
	Remove Controller Removes the selected controller(s) from the list. Removing a controller from the Multi Control Panel also disconnects the communication.

Control Panel Resource Menu

Select *Resource* → *Safety* in the Control Panel to modify the safety settings of the controller.



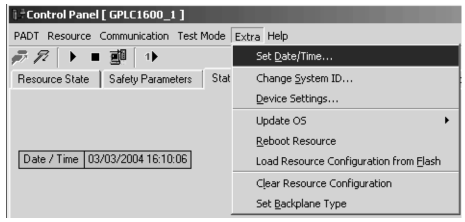
IMPORTANT Any settings you change via the *Resource* menu are directly updated in the controller and are saved in the project.

Menu Item:	Description:
Check Consistency	compares the program running in the controller with the program you are editing in RSLogix Guard PLUS software. If they match, your offline project has previously been downloaded to the GuardPLC.
Set Main Enable	allows safety parameters to be changed. You can only select *Set Main Enable* when the controller is in STOP. For more information, see page 8-7.
Reset Main Enable	keeps safety parameters from being changed. For more information, see page 8-7.
Change Safety Parameters	changes the safety parameters, if *Set Main Enable* is activated. You must have Read/Write or Administrator access to be able to change safety parameters. For more information about these parameters, see page 8-7.

TIP See “Downloading a Routine” on page 6-2 and “Starting a Routine” on page 6-4 for information on *Warmstart, Coldstart, Stop,* and *Download* menu items.

Control Panel Extra Menu

Use the *Extra* menu of the Control Panel to modify communications settings and change controller operation. You must have Administrator access to use most of these menu options as indicated in the table below.



Menu Item:	Description:
Set Date/Time	sets the controller clock, if *Set Main Enable* is activated. Enter the date as mm/dd/yy and the time as hh:mm.
Change System ID (SRS)	changes the system ID (SRS) of the controller. You must have Administrator access to be able to change the system ID (SRS). For more information, see page 4-22.
Device Settings	changes the Ethernet network parameters. You must have Administrator access and the controller must be in STOP mode.
Update OS	allows you to download new COM OS and CPU OS.
Reboot Resource	reboots the controller. See "Recovering From A FAILURE_STOP" on page 8-4.
Load Resource Configuration from Flash	loads a copy of the last executable configuration to the controller
Clear Resource Configuration	deletes the program memory of the controller and resets the configuration of the CPU and COM modules. **GuardPLC 1200 and 2000 only:** This does not affect the battery-buffered memory for long term diagnostics, short term diagnostics, date and time settings, system ID (SRS), or IP address. To reset a controller to default settings, clear the controller and remove the backup battery for at least 20 seconds. Removing the backup battery: • deletes date and time • deletes long term and short term diagnosis • deletes the configuration saved in the battery-buffered memory • deletes all user accounts • does not delete the program memory • does not reset the configuration of the CPU and COM modules Use *Online* → *Communication Settings* and write this value back to the battery-buffered memory. This validates the configuration and you can restart the routine.

Menu Item:	Description:
Set Backplane Type	restores backplane information. The individual modules (CPU, COM, I/O) are linked to each other over the backplane. The controller requires this information to be able to conduct hardware tests. If the EEPROM that stores the backplane information loses its contents, use this menu option to write the backplane type back into the EEPROM. You must have Administrator access to be able to set the backplane type. Follow these steps to set the backplane type: 1. Load a project that is consistent with the connected controller type. **ATTENTION:** If you try to write the backplane type of a controller (such as a GuardPLC 1200 controller) with the backplane type of another controller (such as a GuardPLC 2000 controller), the overwritten controller can no longer be used and must be repaired by the manufacturer. 2. Select *Set Backplane Type*. The backplane type is automatically entered into the dialog window. 3. Change the *Backplane Version* to 0. 4. Click *OK* to confirm the change.

Controller Configuration and Modes of Operation

Using This Chapter

The GuardPLC operating system is stored permanently in the memory of the controller. The operating system is designed to make sure that all tasks of the controller are performed in a safety-related way.

You have access to the controller via the RSLogix Guard PLUS software so that you can define the functionality of the controller.

Controller Modes

The controller can operate in various modes. These modes depend on the results of the tests of the hardware, software, and the system configuration.

After you apply power to the controller or reboot the controller, the controller first performs a system test of the data and address lines and the flash and RAM memories. Then the controller checks the operating system in the flash memory. During this time, the controller is in the INIT mode.

If all these initialization checks are OK, the operating system is started and the controller changes to the STOP mode.

If any hardware and/or software errors are detected, the controller goes to the FAILURE_STOP mode. If the check of the operating system detected errors, the emergency loader starts. The emergency loader loads an operating system from the programming terminal.

If the controller has a valid configuration and a routine downloaded to the controller, the controller goes to the STOP mode.

To have the controller go to the RUN mode:

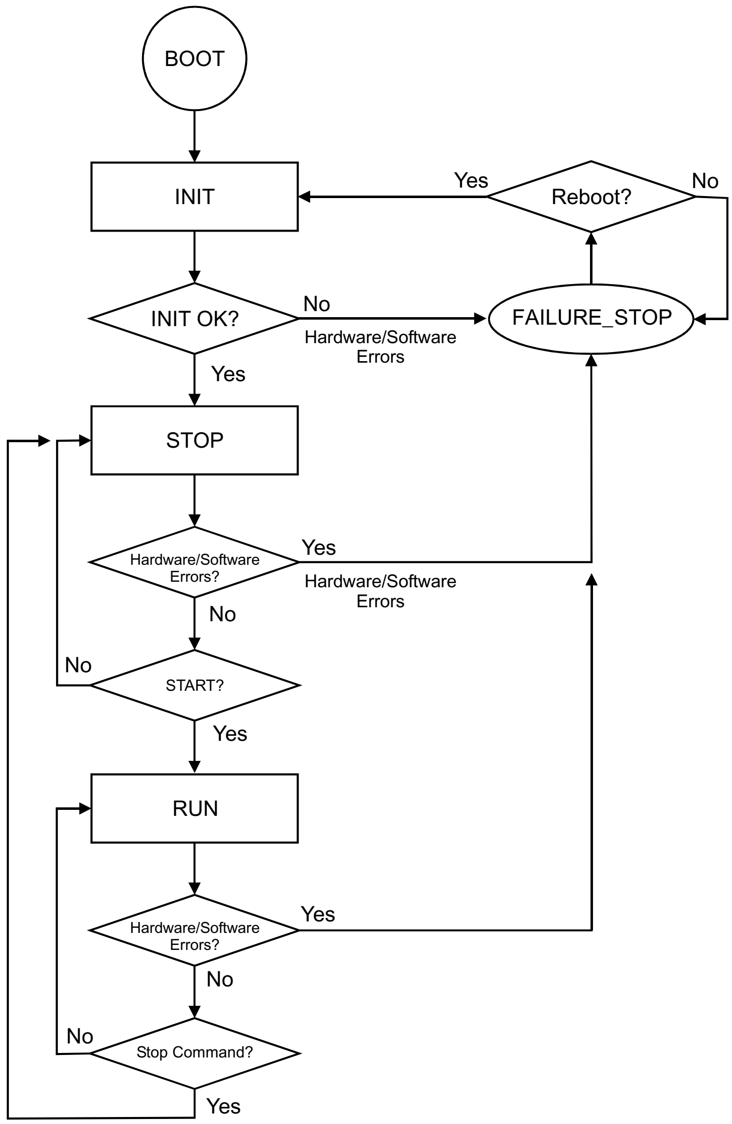
- set the Autostart switch of the both controller and the routine
- manually select RUN mode from the programming software.

If you stop the controller, it transitions from RUN to STOP and interrupts the execution of the routine. The outputs of the routine and the I/O modules are reset to safe values.

You can use the Emergency Stop system variable to put the controller in STOP mode by programming this variable in your logic or forcing it when necessary.

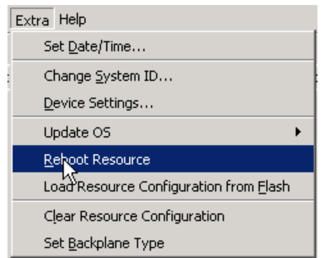
The following table and flowchart summarize the controller modes:

Mode:	**Description:**
INIT	Safe state of the controller during initialization and the hardware tests after booting. • The controller is performing hardware and software tests.
STOP	Safe state of the controller without execution of a routine. • A loaded routine is in the STOP mode. • The outputs of the controller have been reset (LOW). • The controller is performing hardware and software tests.
RUN	The CPU is active. • The routine is being executed. • I/O signals are being processed. • The controller performs non-safety-related communication. • The controller performs software tests, hardware tests, and I/O module tests.
FAILURE_STOP	Safe state of the controller after a system fault. • A loaded routine is in STOP or FAILURE_STOP mode. • The outputs of the controller are being reset (LOW). • The controller is not performing software or hardware tests. • The controller is being held in the safe state. • The hardware watchdog is not triggered. • To recover from FAILURE_STOP, a reboot of the controller is necessary. A reboot can only be initiated via RSLogix Guard PLUS. See “Recovering From A FAILURE_STOP” on page 8-4.


BOOT
INIT
Yes
Reboot?
No
INIT OK?
No
FAILURE_STOP
Hardware/Software
Errors
Yes
STOP
Hardware/Software
Errors?
Yes
Hardware/Software
Errors
No
No
START?
Yes
RUN
Hardware/Software
Errors?
Yes
No
No
Stop Command?
Yes

Recovering From A FAILURE_STOP

If the controller is in FAILURE_STOP, you must reboot the controller by selecting *Extra →Reboot Resource* from the Control Panel as shown below.



TIP A Reboot Resource can only be initiated when the controller is in FAILURE_STOP mode. If you attempt a reboot while the controller is in any other mode, an error message is displayed.

If a routine has already been loaded in the controller when FAILURE_STOP occurs, the controller goes to STOP/VALID_CONFIGURATION after booting. If Autostart Enable is activated, the routine starts up automatically.

If a Routine has not been loaded in the controller when FAILURE_STOP occurs, the controller goes to STOP/INVALID_CONFIGURATION after booting.

TIP If the GuardPLC is in STOP/INVALID_CONFIGURATION after booting, you need to update the SRS. Use the update SRS procedure described on page 6-3.

A brand-new GuardPLC 1200 or 2000 controller, into which a backup battery has not yet been installed, is always in FAILURE_STOP and must be rebooted before you can download a routine.

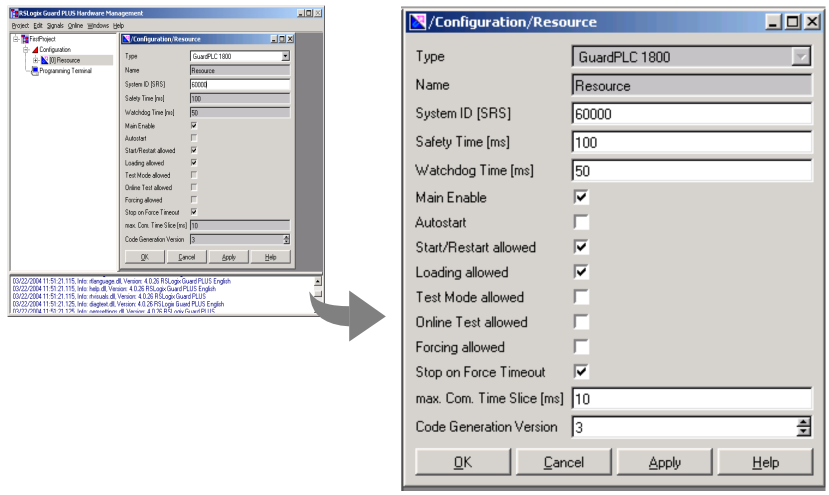
Controller Configuration

To enable the controller to perform its tasks, you have to configure the controller. The parameters you specify are stored in the non-volatile RAM and in the flash file system of the communication section of the controller.

To configure a controller:

1. In the Hardware Management Window, expand the Configuration module.

2. Right-click on *Resource* and select *Properties*.



3. Use the *Type* pull-down menu to select your controller.

4. Set the controller parameters based on the information in the table on page 8-6.

IMPORTANT	The safety time you specify must meet the needs of the controlled process. See the *GuardPLC Controller Systems Safety Reference Manual*, publication 1755-RM001.

<table>
<tr><th>For this parameter:</th><th>Specify:</th></tr>
<tr><td>System ID (SRS)</td><td>the system ID of the controller.

The system ID is a component of the SRS (System, Rack, Slot), and can be in the range of 2 to 65535. The programming terminal uses the system ID to communicate with the controller. The purpose of the SRS is to match a routine to a specific resource and guarantee that only a routine with a matching SRS can be downloaded to a resource.

The system ID of the controller should not be set at 1 because 1 is the default system ID for the programming terminal.

IMPORTANT: The SRS (System, Rack, Slot) set in the configuration is compiled in the routine.EXE file and must match the SRS of the GuardPLC controller in order for a routine to be correctly downloaded to the GuardPLC. A different system ID results in an INVALID_CONFIGURATION error during download.

IMPORTANT: The default SRS of a new controller is 60000. You must use this to establish communications with the controller the first time. Once you establish communications, you can change the SRS.</td></tr>
<tr><td>Safety Time (ms)</td><td>the safety time (in milliseconds) for the controller.

The safety time is the time:
• the controller must react to an input signal with an output signal
• within which the controller must react to an error

The default safety time is 2 times the default watchdog time. You can specify any time from 20 to 50000ms.</td></tr>
<tr><td>Watchdog Time (ms)</td><td>the maximum amount of time (in milliseconds) that the controller can take to execute one cycle.

The watchdog time must be:
• ≥ 10 ms
• ≤0.5 x Safety Time (Worst case, two cycles must occur within the Safety Time. Therefore, Safety Time ÷ 2 is the maximum watchdog time.)
• no more than 5000 ms.

The default watchdog time is:
• 500 ms for GuardPLC 1200 and GuardPLC 2000
• 50 ms for GuardPLC 1600 and GuardPLC 1800
• 10 ms for 1753-IB16, 1753-IB20XOB8, 1753-OB16

If the controller exceeds the watchdog time, the controller goes into FAILURE_STOP.</td></tr>
</table>

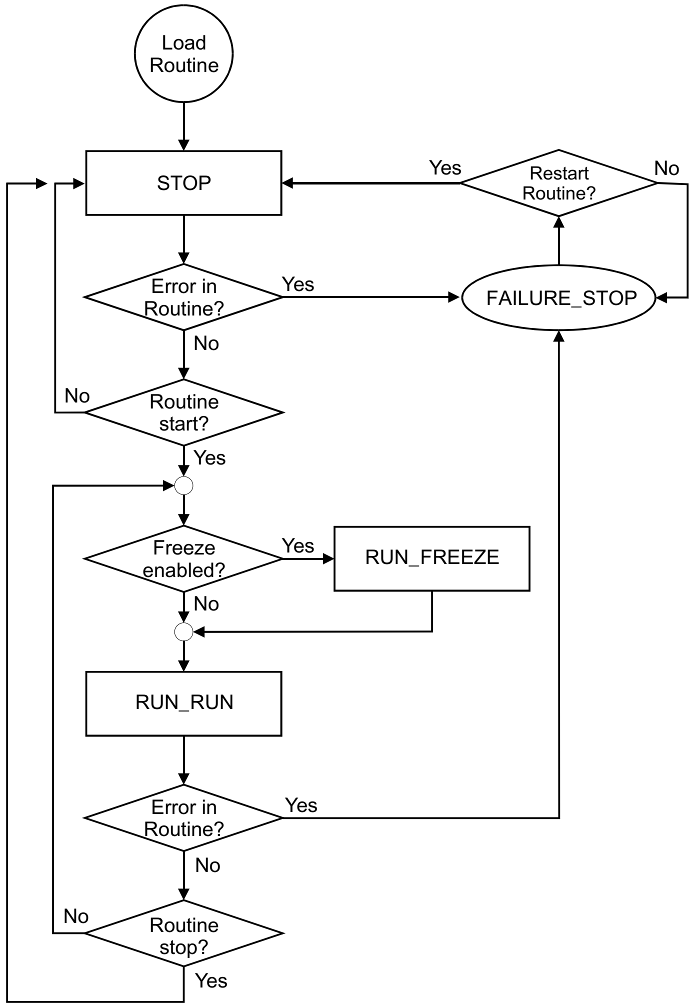
You can set these switches:

This switch:	Specifies:	Default
Main Enable	whether CPU switches can be changed while the controller is executing. If Main Enable is disabled, you cannot change the settings of the other 7 switches (described below) while the controller is in operation (routine in RUN).	On/Enabled
Autostart	whether the controller automatically starts up after rebooting the controller or applying power to the controller. If Autostart Enable is enabled, the routine automatically starts up after a reboot or applying power to the controller.	Off/Disabled
Start/Restart allowed	whether you can start a routine manually. If Start/Restart allowed is enabled, you can start a routine manually via the Routine menu of the Control Panel. Select either *Coldstart* or *Warmstart*. Coldstart is the recommended setting. If Start/Restart allowed is disabled, you cannot start a routine manually. You can only start a routine by rebooting the controller or applying power to the controller.	On/Enabled
Loading allowed	whether you can load new configuration information to the controller. If Loading allowed is disabled, no (new) configuration can be loaded into the controller. This prevents a user from overwriting the current routine.	On/Enabled
Test Mode allowed	whether you can freeze the routine. If Test Mode allowed is enabled, the routine currently running on the controller can be frozen. This allows the Test Mode with Single Cycle function. You are not allowed to freeze a routine in standard operation (this would be non-safe operation).	Off/Disabled
Online Test allowed	whether you can monitor the Function Block code online.	Off/Disabled
Forcing allowed	whether you can force signals. If Forcing allowed is enabled, you can force the signals in the controller. If Forcing allowed is disabled, you can still display the force editor, but the forcing functions are locked.	Off/Disabled
Stop on Force Timeout	whether to stop forcing when the force time expires. If Stop on Force Timeout is enabled, the controller terminates execution of the routine after the user-set force time expires. All outputs go to LOW. If Stop on Force Timeout is disabled, the controller continues executing the routine with the process values when the force time expires.	On/Enabled
Max. Communication Time Slice	the time in milliseconds reserved for a controller to carry out and complete all communication tasks in one CPU cycle. This setting is required for Peer-to-Peer networking.	10 ms

Routine Modes

The controller runs only one routine. The following table and figure summarize the routine modes:

Mode:	Description:
RUN_RUN	The controller is in the RUN mode. • The routine is executed cyclically by the controller. • Input data are processed in the routine. • Output data of the routine are operated.
RUN_FREEZE	The controller is in the RUN mode. • The routine is not executed. • No input data are processed. • No output data of the routine are operated.
STOP	The controller is in the STOP mode. • The routine is no longer being executed. • All outputs have been reset.
FAILURE_STOP	The controller is in the STOP mode. • The routine was stopped due an error. • All outputs are reset. • The hardware watchdog is not triggered. • To recover from FAILURE_STOP, a reboot of the controller is necessary. A reboot can only be initiated via RSLogix Guard PLUS. See “Recovering From A FAILURE_STOP” on page 8-4.
TEST MODE (single step)	The controller is in RUN mode. • The routine is triggered manually. • I/O data are processed. **IMPORTANT:** Test Mode is not permitted for safe operation!



Load a Configuration and Routine (in STOP mode only)

You can load a controller configuration and routine when:

- the controller is in STOP mode, and
- the controller *Loading allowed* switch is set.

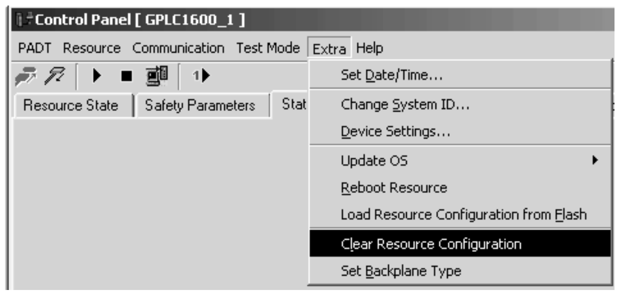
The controller STOP mode is subdivided into these categories:

STOP Mode Category:	Description:
STOP_VALID_CONFIG	The configuration is correctly loaded. The controller can be set to RUN via a command from the programming software. This initiates a loaded user routine.
STOP_INVALID_CONFIG	No configuration loaded or the loaded configuration is faulty. The controller cannot go to RUN.
STOP_LOAD_CONFIG	loading configuration in process

The configuration and the routine are loaded together into the controller. Loading a new configuration and a new routine automatically deletes all previously loaded objects, even if the new objects are faulty.

IMPORTANT	Configuration changes only take effect if you re-generate code before downloading to the controller.

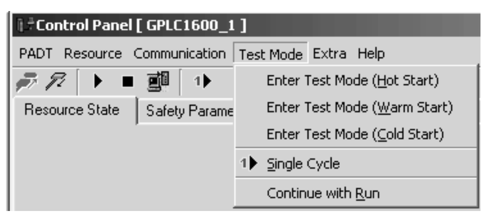
If the controller is in STOP mode, the controller configuration and routine can also be deleted using the programming software's *Clear resource configuration* command. The controller goes into the STOP_INVALID CONFIGURATION mode.



Test Mode of the Routine

In order to execute a single-step operation (cycle step), the controller must be in RUN mode. The *Test Mode allowed* switch must be set to on.

To enter Test mode, select the *Test Mode* menu from the control panel. Then select from *Hot Start, Warm Start,* or *Cold Start*.



The controller state changes to Freeze, and you can now single cycle the routine using the *Single Cycle* option on the *Test Mode* menu. To return to normal operation, select *Continue with Run*.

See "Testing a Routine" on page 6-5 for more information.

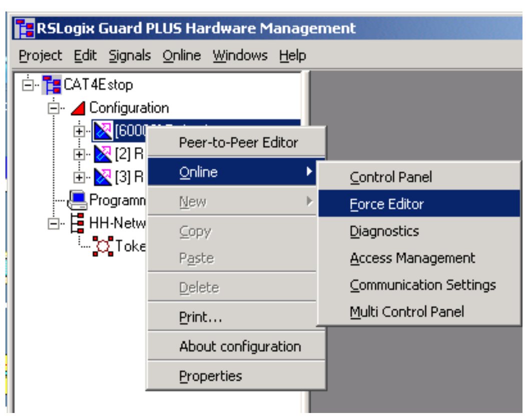
Monitoring and Forcing Signals

Using This Chapter

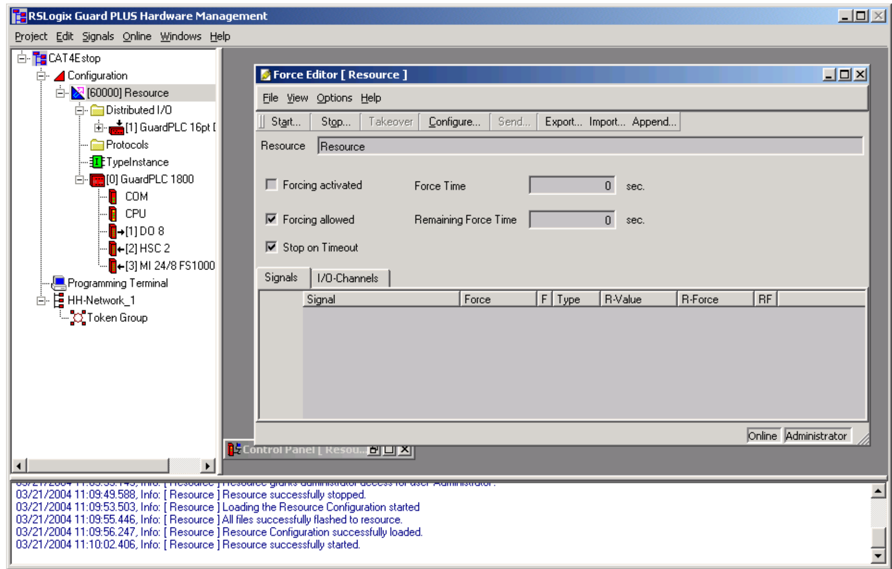
Monitoring Signals

The Force Editor provides a window that lets you select signals to monitor, whether they are forced or not.

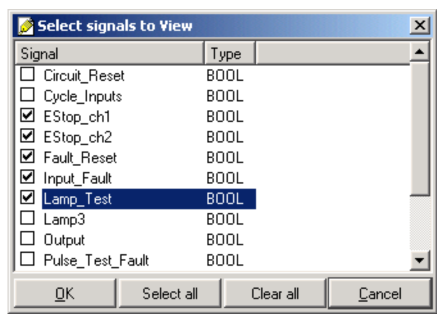
1. Right-click on the resource. Select *Online* → *Force Editor*. If the Control Panel is already open, you do not have to login. Otherwise, the software asks you to log in.



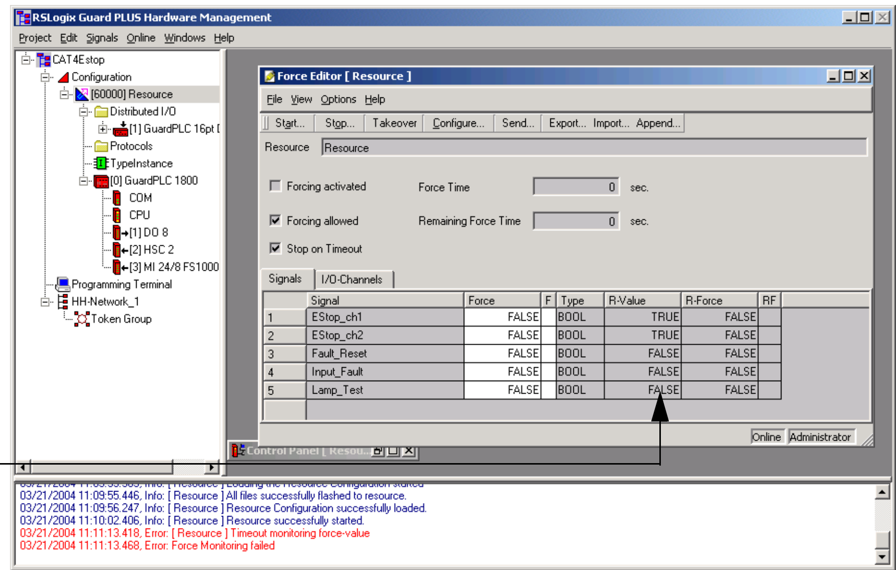
2. After you successfully log in, the software displays the Force Editor.



3. In the Force Editor, select *Configure*. The software displays a list of force signals you can select whether to view or not.



4. If you are not already connected, in the Force Editor, select *File → Connect*. The software displays the values of the signals you selected.



The R-Value (Resource Value) column displays the current values of the signals.

You have the ability to force any of the signals that have been configured in the Force Editor.

Forcing

Forcing describes the intervention of the user in the logic of the user program loaded into the controller. When data is forced, the controller uses the forced values rather than its process values. This changes the value of one or more signals and affects the safety of the controller.

Only signals used in the controller can be forced. The user program and the inputs and outputs are only affected when the controller is in RUN mode.

ATTENTION	When using forcing on a controller with safety tasks, always obey the restrictions listed in the *GuardPLC Controller Systems Safety Reference Manual*, publication 1755-RM001.

Enabling Forces

To enable forcing, both the *Forcing allowed* and *Main Enable* switches must be set. The *Forcing allowed* switch can be set via the programming software, but only if the controller is in RUN or STOP mode.

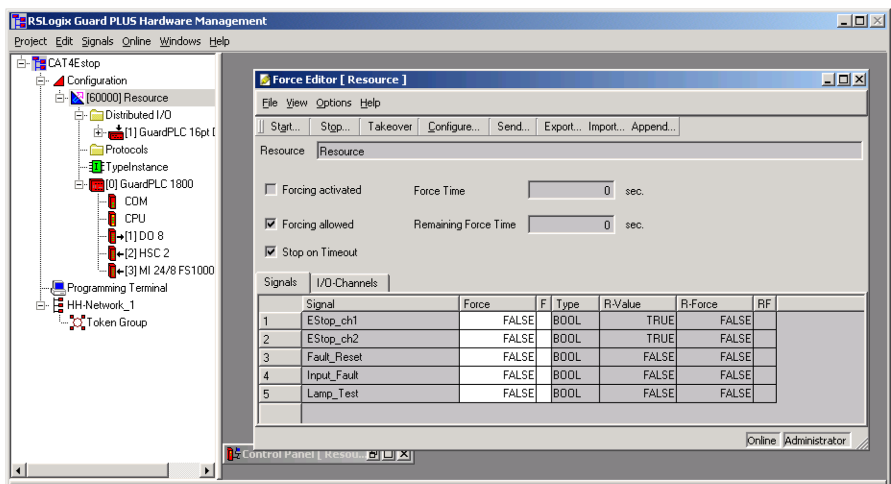
A forced value remains saved in the controller until:

- the user program is stopped,
- the force value is replaced by another value, or
- the controller is switched off.

TIP If a new configuration is loaded, all of the force switches and associated force values are reset.

Starting the Force Editor

Any user can start the Force Editor, regardless of access privilege. However, you can only force signals if *Forcing allowed* is enabled for the controller. Forcing is always disabled for users with Read access.



Before you start the Force Editor, make sure the program running in the controller is the same program that you are editing in RSLogix Guard PLUS software. To verify whether these programs are the same, start the Control Panel and select *Resource* → *Check Consistency*. If the offline/online programs are not identical then the Force Editor will come up offline.

Specifying Force Values and Force Marks

To set a signal with a force value, you:

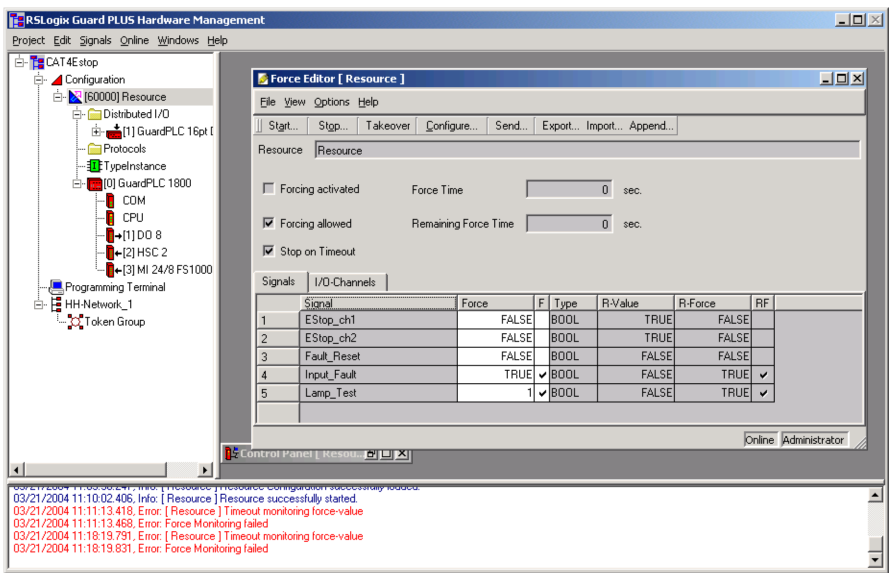
1. Enter the force value for the signal in the *Force* column.

TIP For Boolean signals, “True” or “False” and “1” or “0” are acceptable values.

2. Double-click in the *F* column to mark that you want the controller to use the force value rather than the process value.

3. Send the force value(s) to the controller. The Force Editor displays the force value(s) in the *R-Force* column. A mark in the *RF (resource force)* column indicates that the controller will use the corresponding force value instead of the process value when forcing is enabled.

Multiple force values can be written into the controller at the same time. The force values remain saved in the controller until the routine is reloaded. If the routine is stopped, the resource force marks are also reset.



Field:	Description:
Signal	the name of the signal you want to force.
Force	the value you want to force the signal to have. The value you enter must match the type displayed in the *Type* field.
F (force mark)	a check in this field identifies that the force value you entered is sent to and saved in the controller and will become active when forcing is active. Double-click in this column to mark that you want the controller to use this force value rather than the controller's process value.
Type	displays the type of the signal, as defined in the signal Editor.
R-Value (resource value)	displays the controller value, resulting from the current process and program logic.
R-Force (resource force value)	displays the value of the signal while forcing is active.
RF (resource force mark)	a check in this field identifies that the controller is using the force value rather than the process value as soon as forcing is active.

Force Time

The force time is monitored by the controller. To enter the force time in seconds, the controller must be in RUN or STOP mode with *Forcing allowed* set. For unlimited forcing activity, enter -1.

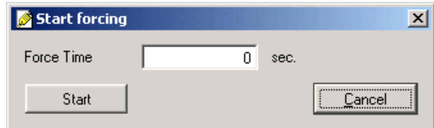
The force time begins when the force process starts. The time is reset to 0 if a new configuration is loaded or if the operating voltage is disconnected.

After the specified time, forcing activity ends. If the controller switch *Stop on Force Timeout* is enabled, the routine returns to STOP mode when forcing ends. If *Stop on Force Timeout* is disabled, the routine continues with the current process values once forcing ends.

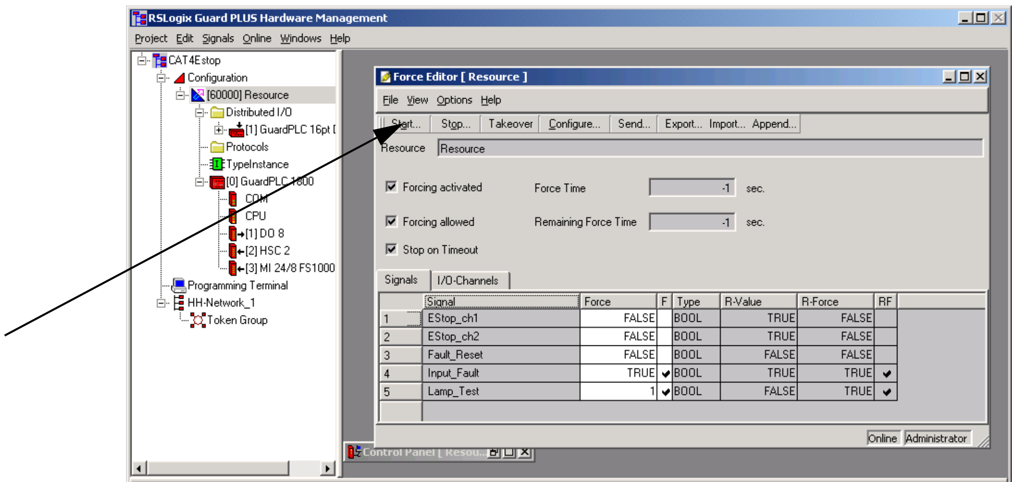
Starting Forces

1. To start forces, select the *Start…* tab or select *File → Start* (see the arrow below).

2. The Force Time dialog box appears. Enter the Force Time.



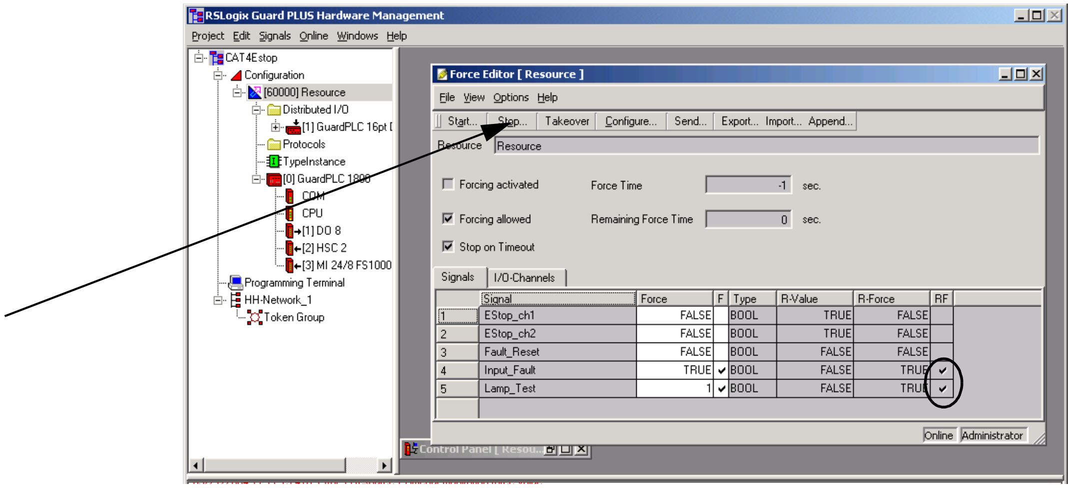
3. The Force Editor opens.



Once forcing starts, the *Forcing activated* box is checked and R-Force values take precedence over R-Values.

Stopping Forces

To stop forcing, select the *Stop…* tab or select *File* → *Stop*.



Once forcing is stopped, the Forcing activated check box is cleared. However, the Resource Force Mark *(RF)* field is still checked, indicating that force values remain in the resource, but are inactive.

Access Management

Using This Chapter

How the Controller Uses Access Levels

An Administrator can set up access privileges for a maximum of ten users per controller. The controller stores the access privileges in its non-volatile memory. The access privileges are not saved with the program, and are not downloaded to the controller with the program. If the controller is changed, access privileges must be re-entered manually.

Every controller has the same default user account, which applies when the controller is:

- new, out of the box
- after disconnecting the operating voltage with the backup battery removed (GuardPLC 1200/2000 only)
- utilizing the Reset button (GuardPLC 1600/1800 only) See “Reset Pushbutton”on page 2-13.

The default account is:

Username: Administrator

Password: <blank>

Access Type: Administrator

The following access levels are available:

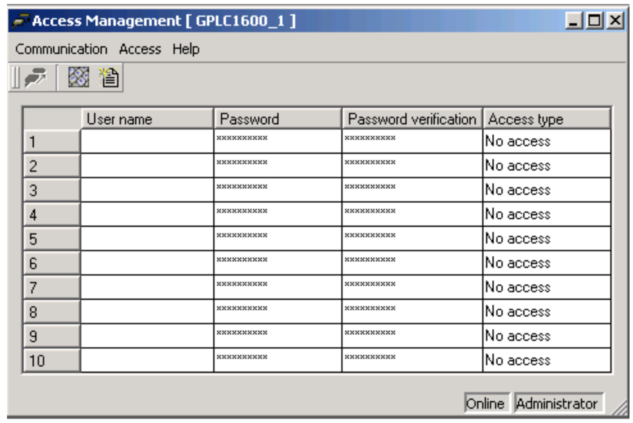
This access level:	Allows:
Administrator	• highest privileges • manage usernames and passwords • read data from controller • write routines and data into controller • force tags • stop, start, freeze, and force a routine • download an operating system • reboot the controller • change IP address and system ID • can also login under read/write and read levels
Read/Write	• read data from controller • write routines and data into controller • force tags • start, stop, freeze, and force a routine • can also login under read level
Read	• lowest privileges • only read data from controller

Creating User Access

To create a user access level:

1. Select *Online →Access Management*. If the Control Panel is open, you do not have to login. Otherwise, the software asks you to log in.

2. After you successfully log in with Administrator access, the software displays the Access Management window.



Field:	**Description:**
Username	name of the user.
Password	password of the user. The password is case sensitive.
Password Verification	verify the password specified above.
Access Type	the access level of the user. Specify Administrator, Read/Write, Read, or No Access.

The username and password are case sensitive and can contain as many as 31 characters. You can use letters, numbers and underscore (_) characters.

At least one of the users must have Administrator privileges.

If you make changes to the user list, use the Set Accounts button to save the changes in the controller.

The Administrator can delete access privileges of all users with the default account access and reset the Administrator account to the default setting of “Administrator” and no password (blank).

IMPORTANT	Changes to access privileges can only be executed when the controller is in the state STOP.

If battery and external power to the controller are simultaneously off, the controller loses all account information and reverts to the default account.

Diagnostics

Using This Chapter

Viewing Controller Diagnostics

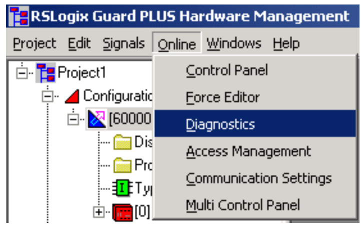
The controller stores short term and long term diagnostics data. The number of entries the controller can save depends on the controller, as shown below:

Type of Data:	GuardPLC 1200		GuardPLC 1600 and 1800		GuardPLC 2000	
	CPU:	COM:	CPU:	COM:	CPU:	COM:
number of short term entries	60	700	60	700	300	700
number of long term entries	250	200	250	200	1000	200

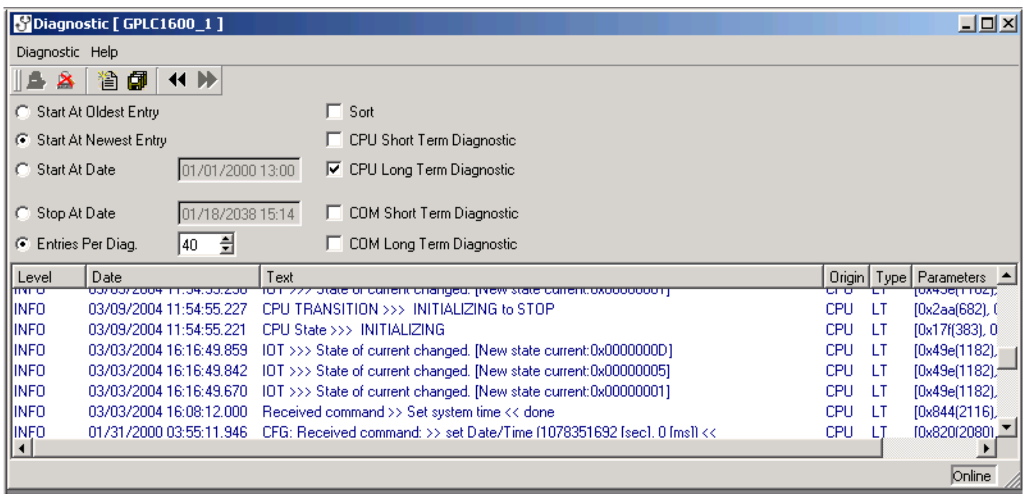
If the memory for short term entries is full and the controller needs to log another entry, the controller deletes the oldest entry.

If the memory for the long term entries is full and the controller needs to add a new entry, the controller deletes the oldest entry only if that entry is more than 7 days old. Otherwise, the new entry is rejected and a message is displayed in the diagnostics window.

1. To display the diagnostics window, left-click on the Resource and select *Online →Diagnostics*. If the Control Panel is already open, you do not have to login. Otherwise, the software asks you to log in.



2. After you successfully log in, the software displays the controller diagnostics.



This field:	Displays:
Level	whether the entry is INFO, WARNING, or ERROR.
Date	the date and time the entry was recorded.
Text	a description of the cause leading to the entry.
Origin	whether the cause of entry originated from the CPU or COM.
Type	whether the entry is short term (ST) or long term (LT).
Parameter	information direct from the CPU or COM. This data is only for error analysis by Rockwell Automation representatives.

TIP You can export diagnostic data to a text file for storage by selecting *Export* from the *Diagnostic* menu.

Selecting Online or Offline Diagnostics

When you start the diagnostics window, Diag. Online is automatically activated. This signals that you want all diagnostics data transferred from the controller to the diagnostics buffer in RSLogix Guard PLUS software. As long as Diag. Online is active, new diagnostic data is transferred to this buffer as it becomes available and if the filter you selected applies.

Diag. Offline disconnects communication with the controller. This ends the transmission of diagnostic data from the controller to the diagnostics buffer in RSLogix Guard PLUS software.

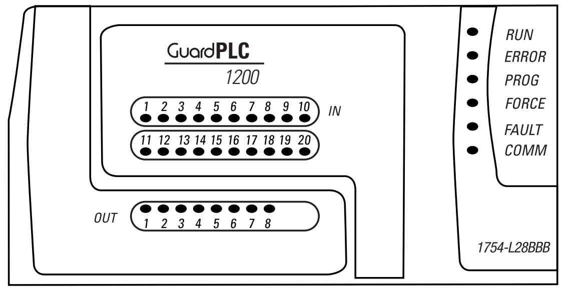
Filtering Diagnostic Data

Select from these filters to determine what diagnostic data to display:

Filter:	Description:
Start At Oldest Entry	Displays all the data from the RSLogix Guard PLUS buffer starting with the oldest entry. The number of lines shown in the table depends on the Entries Per Diag. Enable Sorting defaults to disabled so that the data appears in chronological order from oldest to newest.
Start At Newest Entry	Displays all the data from the RSLogix Guard PLUS buffer starting with the newest entry. The number of lines shown in the table depends on the Entries Per Diag. Enable Sorting defaults to disabled so that the data appears in chronological order from oldest to newest.
Start At Date	Displays entries in chronological order starting at this date and time. The number of lines shown in the table depends on the Entries Per Diag. Enter the date as mm/dd/yy and the time as hh:mm.
Stop At Date	Displays entries in chronological order ending at this date and time. The number of lines shown in the table depends on the Entries Per Diag. Enter the date as mm/dd/yy and the time as hh:mm.
Entries Per Diag.	Determines the maximum number of entries to load into the buffer for the CPU and COM diagnostics. For example, if you enable short term and long term diagnostics for CPU and COM and you set Entries Per Diag. = 10, the diagnostic window contains a maximum of 40 entries (10 entries per diagnostic type). RSLogix Guard PLUS can buffer as many as 5000 entries per type of diagnostic.
Sort	If Sort is disabled, the diagnostic window displays entries in the order they were saved in the controller. If Sort is enabled, the diagnostic window automatically displays entries according to date.
CPU Short Term Diagnostic CPU Long Term Diagnostic COM Short Term Diagnostic COM Long Term Diagnostic	Enables or disables whether to display the diagnostic data for each type.

GuardPLC 1200 LEDs

The GuardPLC1200 controller has these LED indicators:

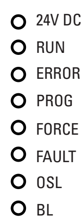


Indicator	State	Condition
INput	On	Digital input channels are high (10 to 30V dc).
	Off	Digital input channels are off.
OUTput	On	Digital output channels are high.
	Off	Digital output channels are off.
RUN	On	This is the normal status of the controller. A routine, which has been loaded into the controller, is executed. The controller processes input and output signals, carries out communication, and performs hardware and software tests.
	Blink	The controller is in STOP mode and is not executing a routine. All system outputs are reset. STOP mode can be triggered by setting the system variable “AB-CPU/Emergency Stop” to TRUE in the routine, or by direct command from the programming terminal.
	Off	The controller is in FAILURE_STOP (see ERROR).
ERROR	On	• A hardware error has been detected by the controller. In this case the controller goes to FAILURE_STOP and the execution of the routine is halted. Hardware errors are errors in the controller, in one or more of the digital input and output modules, or in the counters. • A software error in the operating system has been detected by the controller. • The watchdog has reported an error because of exceeded cycle time. All system outputs will be reset and the controller ceases all hardware and software tests. The controller can only be restarted by a command from the programming terminal.
	Blink	If all the LEDs are on and ERROR blinks, the boot loader detected a corrupted operating system and is waiting for an operating system download.
	Off	No errors are detected.
PROGress	On	The upload of a new controller configuration is in progress.
	Blink	The upload of a new operating system into the Flash ROM is in progress.
	Off	No upload of controller configuration or operating system is in progress.
FORCE	On	The controller is executing a routine (RUN) and FORCE mode is activated by the user.
	Blink	The controller is in STOP, but forcing has been saved and will be activated when the controller is started.
	Off	Forcing is off.

Indicator	State	Condition
FAULT	On	The routine logic has caused an error. The controller configuration is faulty. The upload of a new operating system was not successful and the operating system is corrupted.
	Blink	An error has occurred during a Flash ROM write cycle. One or more I/O errors have occurred.
	Off	None of the above errors have been detected.
COMMunication	On	The programming terminal, with Administrator or Read/Write access, is communicating with the controller via an Ethernet link.
	Off	No communication or read-only communication on an Ethernet link.

GuardPLC 1600 and GuardPLC 1800 Controllers and GuardPLC Distributed I/O

System LEDs



Indicator	State	Condition
24V dc	On	24V dc operating voltage present.
	Off	No operating voltage.
RUN	On	This is the normal status of the controller. A routine, which has been loaded into the controller, is executed. The controller processes input and output signals, carries out communication and performs hardware and software tests.
	Flashing	The controller is in STOP mode and is not executing a routine. All system outputs are reset. STOP mode can be triggered by setting the *Emergency stop* system variable to TRUE in the routine, or by direct command from the programming software.
	Off	The controller is in FAILURE_STOP (see ERROR).
ERROR	On	A hardware error has been detected by the controller. The controller goes to FAILURE_STOP and the execution of the routine is halted. Hardware errors are errors in the controller, errors in one or more of the digital input and output modules, or errors in the counters. A software error in the operating system has been detected by the controller. The watchdog has reported an error due to exceeded cycle time. All system outputs will be reset and the controller ceases all hardware and software tests. The controller can only be restarted by a command from the programming software.
	Off	No errors are detected.

Indicator	State	Condition
PROGress	On	The upload of a new controller configuration is in progress.
	Flashing	The upload of a new operating system into the Flash ROM is in progress.
	Off	No upload of controller configuration or operating system in progress.
FORCE	On	The controller is executing a routine (RUN) and FORCE mode is activated by the user.
	Flashing	The controller is in STOP, but Forcing has been initiated and will be activated when the controller is started.
	Off	Forcing is OFF.
FAULT	On	The routine (logic) has caused an error. The controller configuration is faulty. The upload of a new operating system was not successful and the operating system is corrupted.
	Flashing	An error has occurred during a Flash ROM write cycle. One or more I/O errors have occurred.
	Off	None of the above errors has occurred.
OSL	Flashing	Emergency **O**perating **S**ystem **L**oader is active.
BL	Flashing	**B**oot **L**oader unable to load operating system or unable to start COMM operating system loader.

Communication LEDs

Safety-Related GuardPLC Ethernet

Communication via GuardPLC Ethernet is indicated via two small LEDs integrated into each RJ-45 connector socket.

Indicator	State	Condition
Green	On	Full duplex operation
	Flashing	Collision
	Off	Half duplex operation, no collision
Yellow	On	Connection established
	Flashing	Interface activity

Non-Safety-Related Communication

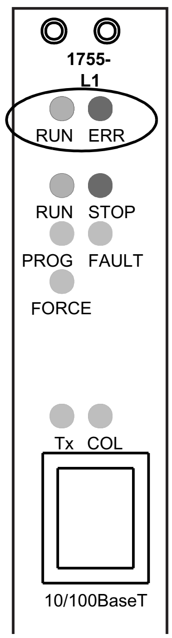
Active communication via the serial ports, COMM1 and COMM3, is indicated by an LED located above the port.

GuardPLC 2000 LEDs

The GuardPLC2000 controller has LED indicators for:

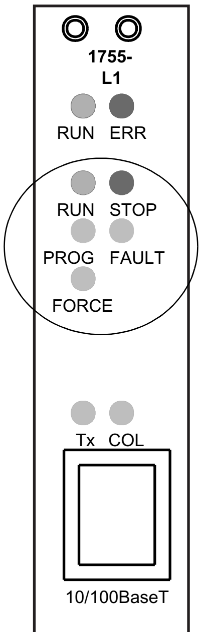
- module, both the program and the communication
- controller and the system hardware
- routine
- Ethernet communication to the programming terminal

Controller Indicators



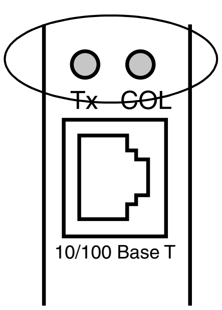
LED	Status	Explanation
RUN	ON	This is the normal status of the controller (RUN or STOP mode). The controller carries out communication and performs software tests.
	BLINK	Downloading an Operating System
	OFF	The controller is in FAILURE_STOP (see LED ERR below), or there is no power supply.
ERR	ON	The controller is in the FAILURE_STOP state and the execution of the routine is halted. All system outputs will be reset and the controller ceases all hardware and software tests. The operating system loader has found a flash error (FAULT is blinking).
	BLINK	The boot loader has found an error in the operating system in the flash (if all other LEDs are ON); the download of a new operating system is awaited.
	OFF	No errors are detected.

Routine Indicators

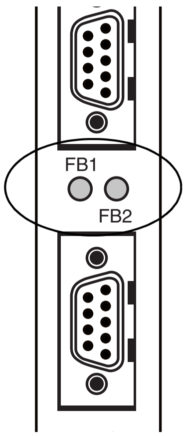


LED	Status	Explanation
RUN	ON	The routine is in RUN or FREEZE.
	OFF	The routine is in FAILURE_STOP.
STOP	ON	The routine is in STOP or FAILURE_STOP.
PROG	ON	The download of a new controller configuration is in progress.
	BLINK	The download of a new operating system into the flash ROM is in progress.
	OFF	No download of controller configuration or operating system is in progress.
FAULT	ON	The routine (user program) has caused an error. The controller configuration is faulty. The download of a new operating system was not successful and the operating system is corrupted.
	BLINK	An error has occurred during a flash ROM write cycle of the operating system. At least one I/O module error is present.
	OFF	No errors have been detected.
FORCE	ON	The controller is executing a routine (RUN) and one or more inputs and/or outputs may be forced by the user.
	BLINK	The controller is in STOP, but one or more inputs and/or outputs have been prepared for forcing and will be activated as soon as the controller is started.
	OFF	No inputs and/or outputs are forced or are prepared to be forced.

Ethernet Communication Indicators



LED	Status	Explanation
Tx	On	Data is transmitting via Ethernet by the communication processor.
COL	On	A collision on Ethernet is detected.

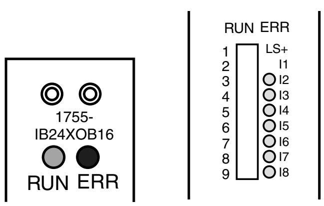


Serial Communication Indicators

LED	Status	Explanation
FB1	On	Field bus no. 1 is active
FB2	On	Field bus no. 2 is active (serial interface module)

IMPORTANT Only the bottom serial port on the GuardPLC 2000 controller is active, as indicated by the FB2 LED.

1755-IB24XOB16 LEDs



The 1755-IB24XOB16 digital combination input and output module (AB-DIO) has LED indicators for:

- power supply
- module status
- I/O status

Power Supply and Module Status

LED	Status	Explanation
RUN	ON (green)	The module has the correct operating voltage (24V dc).
	OFF	The module has no power.
ERR	ON (red)	If the system is in STOP mode, one or more of the inputs or outputs is faulty, or the module is faulty. Use the RSLogix Guard PLUS software to verify the location of the fault. If the module is faulty, replace the module immediately, or the safety-related operation of the GuardPLC 2000 controller is not maintained.
	OFF	The module is operational.

I/O Status

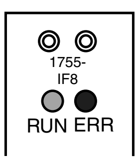
Status	Explanation
ON (yellow)	• Input is high • Output is energized
OFF	• Input is low • Output is de-energized

While the system is in RUN mode, ERR is indicated continuously for both a module and a channel error. Depending on the type of error, the module switches off only a faulty output channel, but the operation of the other outputs continues, or all the output channels are switched off. The inputs are always in operation. A faulty input channel transmits Low-signal to the logic. If the entire module is switched off, all input and output channels are switched off.

1755-IF8 Analog Input Module LEDs

The 1755-IF8 analog input module (AB-AI) has LED indicators for:

- power supply
- module status



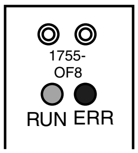
LED	Status	Explanation
RUN	ON (green)	The module has the correct operating voltage (24V dc).
	OFF	The module has no power.
ERR	ON (red)	If the system is in STOP mode, one or more of the inputs or outputs is faulty, or the module is faulty. Use the RSLogix Guard PLUS software to verify the location of the fault. If the module is faulty, replace the module immediately, or the safety-related operation of the GuardPLC 2000 controller is not maintained.
	OFF	The module is operational.

While the system is in RUN mode, ERR is indicated continuously for both a module and a input channel error. Depending on the type of error, the module may switch off only one input channel (i.e., a faulty channel transmits the value 0 to the logic, but the module continues operation with the remaining channels). If the entire module is switched off, all input channels transmit the value 0 to the logic.

1755-OF8 Analog Output Module LEDs

The 1755-OF8 analog output module (AB-AO) has LED indicators for:

- power supply
- module status



LED	Status	Explanation
RUN	ON (green)	The module has the correct operating voltage (24V dc).
	OFF	The module has no power.
ERR	ON (red)	If the system is in STOP mode, one or more of the inputs or outputs is faulty or the module is faulty. Use the RSLogix Guard PLUS software to verify the location of the fault. If the module is faulty, replace the module immediately or the safety-related operation of the GuardPLC 2000 controller is not maintained.
	OFF	The module is operational.

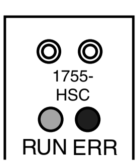
While the system is in RUN mode, ERR is indicated continuously for both a module and an output channel error. Depending on the type of error, the module may switch only one pair of output channels (1+2, …, 7+8) to the de-energized state (i.e. the value 0V or 0 mA), but the module continues operation with the remaining channels. If the entire module is switched off, all output channels are switched to the de-energized state.

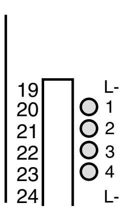
1755-HSC Combination High-Speed Counter and Output Module LEDs

The 1755-HSC combination high-speed counter and output module (AB-CO) has LED indicators for:

- power supply
- module status
- I/O status

Power Supply and Module Status





LED	Status	Explanation
RUN	ON (green)	The module has the correct operating voltage (24V dc).
	OFF	The module has no power.
ERR	ON (red)	If the system is in STOP mode, one or more of the inputs or outputs is faulty or the module is faulty. Use the RSLogix Guard PLUS software to verify the location of the fault. If the module is faulty, replace the module immediately or the safety-related operation of the GuardPLC 2000 controller is not maintained.
	OFF	The module is operational.

I/O Status

LED	Status	Explanation
1, 2, 3, 4	ON (green)	The corresponding output is energized.
	OFF	The corresponding output is de-energized.

While the system is in RUN mode, ERR is indicated continuously for both a module and a counter channel error. Depending on the type of error, the module may switch off only one counter channel (i.e., the counter transmits the value 0 to the logic, the output has no signal, but the module continues operation with the remaining counter channel). If the entire module is switched off, all counter channels are switched off.

Peer-to-Peer Communication Overview

Using This Chapter

Peer-to-Peer Communication Basics

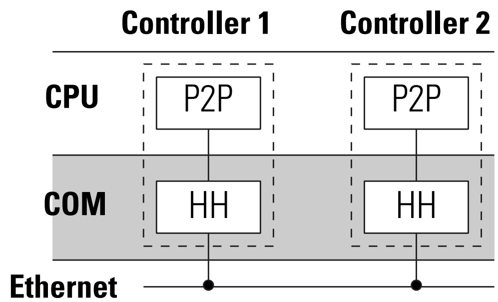
Peer-to-Peer communication is used for data exchange between two or more controllers and distributed I/O on a GuardPLC safe Ethernet network. GuardPLC Ethernet is certified for use in SIL3 and CAT 4 applications and is designed to carry safety-related data. The controllers are usually connected via Ethernet, but other means of communication, such as telephone lines or two-way radios are also possible, using gateways from Ethernet to the respective technology.

The Peer-to-Peer protocol is primarily responsible for:

- the communication between controller CPUs, including automatic connection setup
- extended diagnostics
- all safety-relevant features for correct data transfer

Each controller is equipped with one or more 10/100 Base T Ethernet ports. The High-Level High-Speed (HH) protocol is implemented in the operating system of the GuardPLC 1200/1600/1800 and GuardPLC 2000 communication module (COM) and interacts with the Ethernet port. The HH protocol is based on UDP/IP and IEEE 802.3 standards and is responsible for the collision-free data exchange via standard Ethernet in various network topologies.

As seen in the figure below, both the HH and the Peer-to-Peer protocols are vital for safe Ethernet Communication. HH protocol can be considered the wire or transport media through which messages are passed. Peer-to-Peer (P2P) is the protocol that runs on the wire, making sure that the messages are transmitted over the HH connection within the watchdog time. P2P is the mechanism that qualifies GuardPLC Ethernet as a safety network.



TIP The Peer-to-Peer protocol is designated as a safe protocol according to DIN V 19250(AK6), IEC61508 (SIL 3) and EN 954-1 (CAT 4) respectively.

Networking Limitations

A peer-to-peer link is defined as communication from one GuardPLC to another GuardPLC, or from a GuardPLC to a distributed I/O block. A device on an Ethernet network must make a connection to another device on the Ethernet network in order for the two of them to communicate. Connections only need to be established between devices that wish to communicate with each other.

A single GuardPLC controller may have up to 64 connections to other devices on the GuardPLC Ethernet network (GuardPLC controllers, GuardPLC Distributed I/O Blocks, OPC servers, or programming terminals). Each connection can transfer up to 900 Bytes of data in each direction (read and write). The data size is determined by the number of signals transferred between the devices.

In contrast, a GuardPLC Distributed I/O block can only have one connection, the connection to the controller that 'owns' it. The amount of data shared between a DIO block and the controller is fixed and defined by the type of I/O block.

The total number of controllers, DIO blocks, OPC servers, and programming terminals on a network is only limited by the number of available IP addresses and the network bandwidth (max 100 Mbits per second) of a segment of the network. However, large amounts of data flowing on the network will affect the network response time, and therefore the safety time of the system.

Network Configuration

Communication between GuardPLC controllers can be established via different kinds of Ethernet topologies. Both the HH protocol and the Peer-to-Peer protocol can be adapted to the network in use, to allow smooth and efficient data transfer.

You configure the HH protocol and the Peer-to-Peer protocol by setting parameters, either manually or with the help of network profiles. Network profiles are preset combinations of parameters you can select to make configuration simpler.

To optimize data transfer and customize the configuration, you must have an extensive knowledge of the network in use and the operation of the parameters. The following sections summarize the most important HH and Peer-to-Peer protocol parameters.

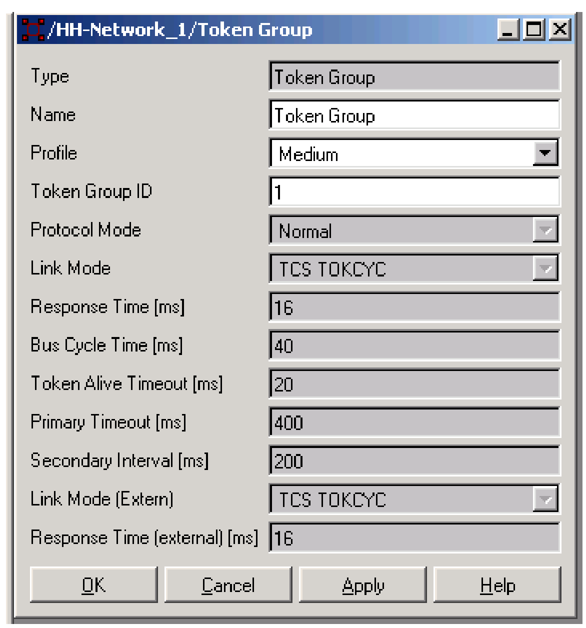
HH Protocol Parameters

The HH protocol parameters are displayed in the HH Network/Token Group window. They can be preset by selecting one of two profiles:

- Fast
- Medium

The profiles are explained in HH Network Profiles on page 12-10.

TIP While manual changes to the parameters are possible by selecting the "None" profile, keep in mind that ill-considered changes can disable communication completely.



Token Group ID

The Token Group ID is the numerical identifier for a Token Group. Each Token Group must have its unique ID.

Protocol Mode

Choose either Normal or RAW protocol mode.

Normal

In Normal mode, software token passing is ON, meaning that access to the Ethernet network is controlled via token passing. Only the controller that holds the token is allowed to access the network.

This mode is recommended for networks with slow hubs to avoid message collisions.

RAW

In RAW mode, software token passing is OFF. No token is created. Ethernet access is coordinated by hardware only. The affiliated Link Mode is “TCS direct”.

Data transfer is faster than in “Normal Mode” and message collisions are prevented by the switching and full-duplex mode ports.

This mode is recommended for networks, where full-duplex (recommended) LAN-switches are used exclusively, or the switches integrated into the GuardPLC 1600 and 1800 can be used.

Link Mode

Select either TCS Direct or TCS TOKCYC.

TCS Direct

In TCS Direct mode, safety-related data are sent as soon as they are prepared for transmission. Network media access is coordinated by hardware.

TCS TOKCYC

This Link Mode corresponds to Protocol Mode “Normal”.
Safety-related data are sent when the controller receives the token.
Network media access is coordinated by software.

Response Time

Response Time is the controller’s maximum permissible Response Time for a network message. PES_1 (Programmable Electronic $System_1$) sends a message to PES_2 and expects the answer within the Response Timeout.

The actual values of the ResponseTime can be read in the HH Status of the Control Panel.

	Controller	LinkId	Status	RspT last [ms]	RspT avg [ms]	RspT max [ms]	RspT min [ms]
1	GPLC1200_2	2	connected	2	2	2	2
2	GPLC2000_1	3	connected	3	3	3	3

Token Cycle Time

This is the maximum permissible time for one token cycle. In other words, the time within which a controller expects the token.

The Token Cycle Time depends on the number of controllers in a Token Group and can be read in the HH Status of the Control Panel.

		last	avg	max	min
1	Token Cycle Time	11	8	36	5

Token Alive Timeout

The current holder of the token must send a token alive message to the Primary(1) controller within this time period or the Primary assumes the token is bad. If the token alive message is missing, a new token is created by the Primary.

(1) The Primary is the controller that generates and supervises the token.

Primary Timeout

Time, within which the Primary expects a check for liveliness from the Secondary[1] controller. If the liveliness check fails to appear, the Primary assumes that the present Secondary is disconnected. In this case, the Primary selects a new Secondary.

Secondary Interval

Time, after which the Secondary checks the Primary for liveliness. The Secondary Interval is less than the Primary Timeout.

Link Mode (Extern)

Same as Link Mode above, except for the connection is to a controller in another Token Group.

Response Time (Extern)

Same as Response Timeout above, except for the connection is to a controller in another Token Group.

Peer-to-Peer Protocol Parameters

All Peer-to-Peer protocol parameters are displayed in the Peer-to-Peer Editor. With the exception of the ResponseTime and the ReceiveTMO, which have to be configured by the user, all other Peer-to-Peer protocol parameters are automatically preset with the selection of a Peer-to-Peer profile. See "Configure Peer-to-Peer Communication" on page 13-11 for detailed instructions on how to configure the Peer-to-Peer protocol.

(1) The Secondary is a controller in the same Token Group as the Primary. The Secondary supervises the Primary.

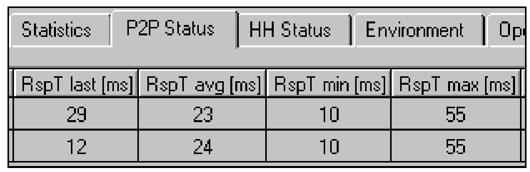
Message Response Time (ReponseTime)

ResponseTime is the user-configurable time it takes to receive an acknowledgement of a sent message from the recipient.

The ResponseTime is not a freely configurable parameter, but results from the physical conditions of the communication path and the configuration of the network protocol.

Because the ResponseTime influences the speed of message exchange, a test run is recommended to investigate network timing.

Use the *P2P Status* tab, in the Control Panel to display the minimum, maximum, and average ResponseTime.



The ResponseTime is the sum of the following variables, described in the table below:

$$\text{ResponseTime} = T_{GR1} + T_1 + T_{GR2} + T_3 + T_2$$

Table 12.1 Response Time Variables

Variable:	Definition:
T_{GR1}	Message delay between two PES: $\mathbf{CPU_1} \rightarrow COM_1 \rightarrow$network$\rightarrow COM_2 \rightarrow \mathbf{CPU_2}$
T_1	Time on CPU_2 to process all protocol stacks: T_1 = CycleTime(CPU_2) x n_2 where n_2 is the number of cycles needed on CPU_2 to process all protocol stacks. Set the Communication Time Slice (see below) large enough to allow all protocol stacks to be processed in one cycle.
T_2	Delay of the acknowledgement on CPU_2: T_2 = AckTMO + n_2 x [0 … CycleTime(CPU_2)] If AckTMO = 0 or ProdRate = 0, then $T_2 = 0$
T_{GR2}	Message delay between two PES: $\mathbf{CPU_2} \rightarrow COM_2 \rightarrow$network$\rightarrow COM_1 \rightarrow \mathbf{CPU_1}$ (usually identical with T_{GR1})
T_3	Time on CPU_1 to process all protocol stacks: T_3 = CycleTime(CPU_1) x n_1 where n_1 is the number of cycles needed on CPU_1 to process all protocol stacks. Set Communication Time Slice (see page 13-2) large enough to allow all protocol stacks to be processed in one cycle.

Receive Timeout (ReceiveTMO)

ReceiveTMO is the safety-related, user-configurable monitoring time, within which PES_1 must receive a correct response from PES_2.

TIP ReceiveTMO is also valid for the return path from PES_2 to PES_1.

If ReceiveTMO elapses, safety-related communication closes down and all imported (via communication) safety-related tags reset to their user-configurable initial values.

If the ReceiveTMO ≥ 2 x ResponseTime(minimum), the loss of at least one message can be handled without losing the Peer-to-Peer connection.

If the ReceiveTMO is not ≥ 2 x ResponseTime(minimum), the availability of the Peer-to-Peer connection is only guaranteed in a collision- and noise-free network. However, this does not result in a safety problem for the CPU!

TIP The maximum permissible value for ReceiveTMO depends upon the application and is set in the Peer-to-Peer Editor along with the expected maximum ResponseTime and the profile.

Profile	Response Time	Receive TMO
Fast & Cleanroom	40	80
Fast & Cleanroom	100	100

Resend Timeout (ResendTMO)

Resend Timeout is the safety-related monitoring time of PES_1. If the receipt of a data transmission is not confirmed by PES_2 within this time period (ResendTMO), PES_1 repeats the data transmission.

Acknowledge Timeout (AckTMO)

Reception of data must be confirmed by the CPU with an ACK (acknowledge) message to the sender of the data. If the CPU is busy, ACK is delayed. Acknowledge Timeout is the maximum delay an ACK message may have.

The AckTMO cannot be entered manually, but is set in conjunction with a profile in the Peer-to-Peer Editor. For fast networks, AckTMO is zero.

Queue Length (QueueLen)

QueueLen describes the number of messages which may be transmitted without having to wait for an acknowledgement. It corresponds to the network bandwidth and delay.

QueueLen cannot be entered manually, but is set along with a profile in the Peer-to-Peer Editor.

Production Rate (ProdRate)

ProdRate is the minimum time interval between two data messages. The purpose of ProdRate is to limit the amount of data to a magnitude which can be transported to the recipient without overloading a (slow) communication channel. This results in an even load on the communication channel and avoids the reception of outdated data.

TIP A production rate of 0 means that a data message can be transmitted with each cycle of the user program.

Watchdog Time (WDZ)

Watchdog Time is the maximum permissible duration of a RUN cycle on a PES. The RUN cycle depends upon the complexity of the user program and the number of Peer-to-Peer connections.

Worst-Case Reaction Time (T_R)

Worst-Case Reaction Time is a safety-relevant application parameter. It is the time between the occurrence of a physical input signal change at PES_1 and the corresponding physical output signal change at PES_2:

$$\text{Worst-Case Reaction Time } (T_R) \leq t_1 + t_2 + t_3 + t_4$$

where:

Table 12.2 Worst-Case Reaction Time Variables

Variable:	Definition:
t_1	The worst-case time for the user program on PES_1 to process the input signal and prepare the data for transmission. 2 x WDZ (PES_1)
t_2	The additional transmission delay on PES_1. Equals 0 ms, if the ProdRate is 0. Otherwise: equals ReceiveTMO + WDZ (PES_1)
t_3	ReceiveTMO The maximum age of a message when received at PES_2.
t_4	The maximum time for the received data message to be processed by the user program on PES_2 and the output signal to be set.

The Worst Case ReactionTime T_R is process-dependent and has to be coordinated with the approving board. In the Peer-to-Peer Editor, the Worst Case ReactionTime can be read in the "Worst Case" column.

	Resource	Worst Case	Profile
1	GPLC1200_2	1180	Fast
2	GPLC2000_1	2100	Fast

HH Network Profiles

Two HH network profiles are used to configure the appropriate set of parameters for the network in use. These profiles, described below, can be selected in the Properties of the HH Network Token Group.

- Profile I: Fast
- Profile II: Medium

A third profile option, None, allows you to set parameters manually. See The "None" Profile on page 12-16 for more information.

Profile I: Fast

This is the recommended profile. It provides the fastest data throughput, and covers approximately 95% of all application cases.

Use Fast for:

- applications which require fast data update rates within a Token Group(1).
- fast communication between two or more Token Groups(1), where the other Token Groups must run Fast as well.
- applications which require the shortest feasible Worst-Case Reaction Time.

TIP Because Token Passing is switched off in the Fast profile, it is possible to generate a Token Group with only one controller. No second controller is needed to exchange the token. The single controller can communicate with other Token Groups containing more controllers.

The minimum network requirements are outlined in the table below.

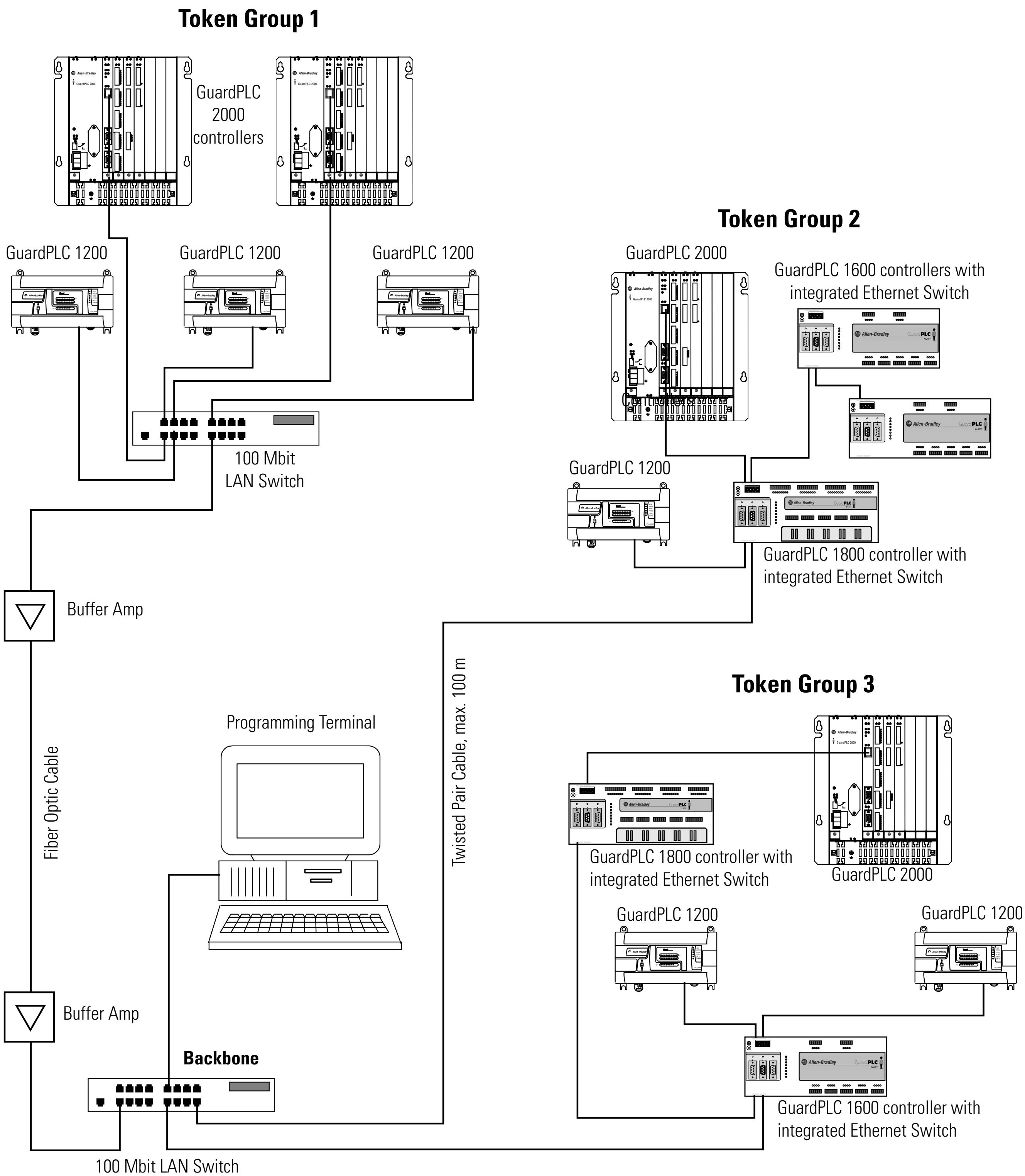
Table 12.3 Minimum Ethernet Network Requirements for Profile I

Requirement	Definition
Fast	100 Mbit technology (100 Base TX)
Switched	Fast Ethernet (full-duplex recommended) LAN switches or integrated switches (GuardPLC 1600/1800) required.
Cleanroom	No loss of data due to traffic overload, harsh environmental conditions, or network defects.

TIP The network can be shared with other applications, if sufficient bandwidth is provided.

(1) A Token Group consists of at least two controllers, which share the same token. Each controller must be a member of exactly one Token Group. A Token Group can work stand-alone or can exchange data with other Token Groups.

Example of HH Network Profile I Topology



Profile II: Medium

This profile provides medium-speed data throughput and covers approximately 4% of all application cases. It is appropriate for applications where timing is not a critical factor. With the Medium profile, network media access within a Token Group and communication with external Token Groups is controlled by Token Passing. These external Token Groups must also run Medium profiles.

IMPORTANT In the Medium profile, a Token Group must be comprised of at least two controllers to carry out Token Passing, otherwise the controller configuration is erroneous. (“STOP/INVALID CONFIGURATION”).

Table 12.4 Minimum Ethernet Network Requirements for Medium & Cleanroom

Requirement:	Definition:
Medium:	10 Mbit technology (10 Base T) Hubs are used within the Token Groups and LAN switches connect one Token Group to another.
Clean:	No loss of data due to traffic overload, harsh environmental conditions, or network defects.

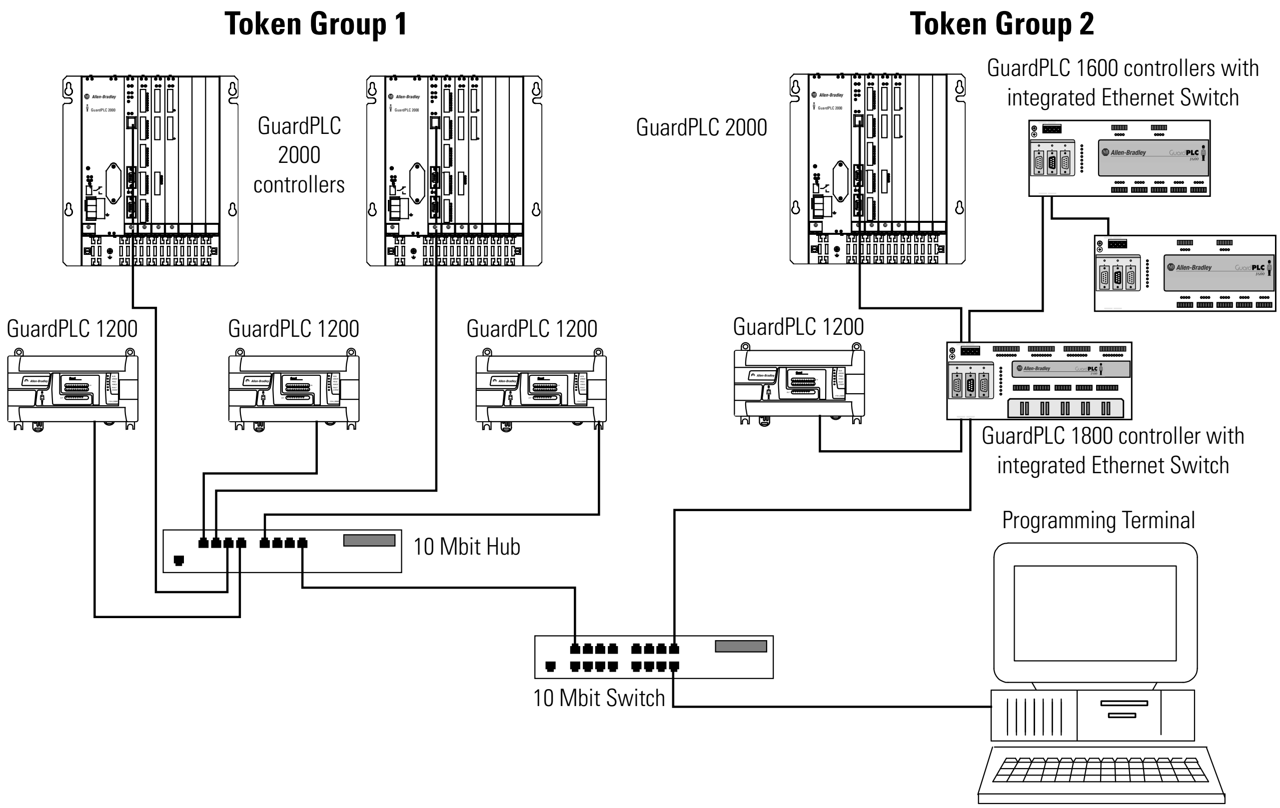
IMPORTANT The network must not be shared with other applications. Do not use more than one Programming Terminal (recommended). Programming Terminals increase network traffic, but do not participate in Token Passing!

Using LAN Switches and Hubs

When using a hub instead of a LAN switch to interconnect two or more controllers of the same Token Group, network access within the Token Group is no longer conducted by the hardware, but must be managed by Token Passing.

Each Token Group handles its Token Passing individually, depending on user settings, CPU cycle times, network topology, etc. This means that for two (or more) Token Groups, which are exchanging data, Token Passing is not synchronized, resulting in a loss of messages between the Token Groups.

IMPORTANT To minimize loss of messages, only one controller in a Token Group is allowed to exchange data with exactly one controller in a second Token Group. Furthermore, the overall number of links between Token Groups is limited to eight.



The illustration above shows an application, consisting of two Token Groups. The Token Groups equipped with hubs require Token Passing to coordinate network access within the Token Groups. The Token Groups are interconnected via a LAN switch.

In this network topology, only one controller in Token Group 1 is allowed to exchange data with one controller in Token Group 2. If Token Group 2 needs data from different controllers in Token Group 1, the “talking” controller in Token Group 1 must collect the data.

In the HH Network Profile II Configuration Topology example on page 12-16, only the following links between Token Groups are allowed:

- A1 ↔ A2
- B1 ↔ B2
- C1 ↔ C2

To configure this scenario, the controllers are placed in their respective token groups:

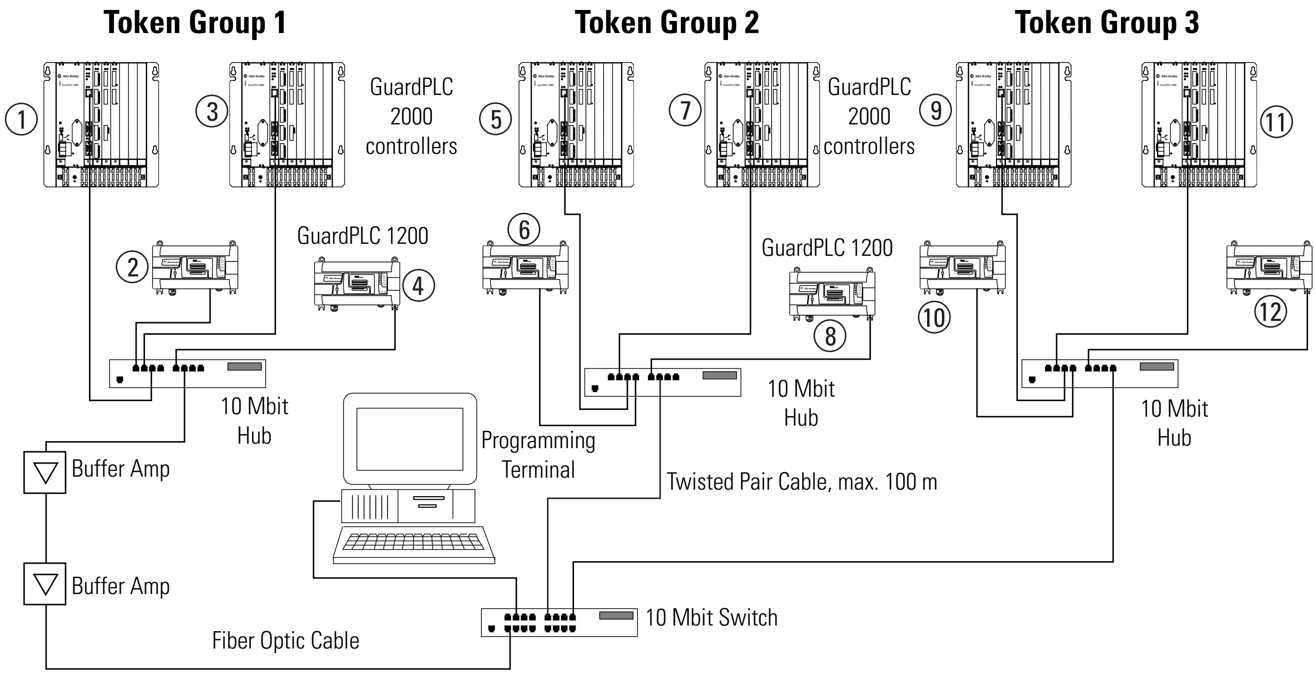
Token Group 1	Token Group 2	Token Group 3
Controller 1	Controller 5	Controller 9
Controller 2	Controller 6	Controller 10
Controller 3	Controller 7	Controller 11
Controller 4	Controller 8	Controller 12

In the Peer-to-Peer Editor, you create connections between controllers. For example, all controllers in Token Group 1 can communicate to each other, but Controller 1 can also communicate to Controller 5 in Token Group 2:

Token Group 1 Connections

Controller 1	Controller 2	Controller 3	Controller 4
Controller 2	Controller 1	Controller 1	Controller 1
Controller 3	Controller 3	Controller 2	Controller 2
Controller 4	Controller 4	Controller 4	Controller 3
Controller 5	—	—	—

HH Network Profile II Configuration Topology



The “None” Profile

The None profile is different from the profiles described previously because it has no pre-defined parameters. You must set all the parameters manually.

To set the parameters, select either Fast or Medium from the HH Network/Token Group window, and press the Apply button. This presets the parameters according to the profile.

To enable manual changes and activate the entry fields, select None and press *Apply* again. The former parameter settings will be overridden and can then be changed.

Because the profiles Fast and Medium cover nearly all conceivable network topologies, None is only recommended for evaluation purposes. An extensive knowledge of the functions of the parameters, their value ranges, and their impact on the availability of the network is required for proper manual parameterization.

IMPORTANT	The None profile should not be used in regular applications.

Peer-to-Peer Network Profiles

Due to the variety of parameters, manual network configuration is very complex and requires extensive knowledge of the parameters and how they influence one another.

To simplify the setup, RSLogix Guard PLUS provides six Peer-to-Peer profiles, which can be selected by the user, depending upon application requirements and the capabilities of the network.

Profiles are combinations of matched parameters which are automatically set when the user chooses a certain profile. The intention of all profiles is to optimize the data throughput on the network, which minimizes the ReceiveTMO and results in a low Worst Case ReactionTime. (For the definitions of the Peer-to-Peer network parameters, see page 12-6).

The six profiles are described in the following sections:

- Fast & Cleanroom,
- Fast & Noisy,
- Medium & Cleanroom,
- Medium & Noisy,
- Slow & Cleanroom, and
- Slow & Noisy

Peer-to-Peer Profile I: Fast & Cleanroom

This profile provides the fastest data throughput for applications which require fast data update rates. It is also best for applications which require the shortest feasible Worst-Case ReactionTime.

Fast & Cleanrooom Characteristics

<table>
<tr><td rowspan="3">Minimum Ethernet network requirements(1)</td><td>Fast</td><td>100 Mbit technology (100 Base TX)</td></tr>
<tr><td>Switched</td><td>Fast Ethernet (full-duplex recommended) LAN switches or integrated switches (GuardPLC 1600/1800) required.</td></tr>
<tr><td>Cleanroom</td><td>No loss of data due to traffic overload, harsh environmental conditions or network defects.</td></tr>
<tr><td>Characteristics of the communication path</td><td colspan="2">Minimum delays
ResponseTime $\leq$ReceiveTMO $\div$ 2 (otherwise ERROR)</td></tr>
<tr><td rowspan="3">Variables</td><td>ResponseTime</td><td>manually set in the Peer-to-Peer Editor</td></tr>
<tr><td>ReceiveTMO</td><td>manually set in the Peer-to-Peer Editor</td></tr>
<tr><td>WDZ (Watchdog Time)</td><td>manually set in the controller properties</td></tr>
<tr><td>Suitable HH network profile</td><td colspan="2">Fast</td></tr>
<tr><td>Peer-to-Peer parameter presets</td><td colspan="2">• QueueLen = 2
• Communication Time Slice large enough to process and send all data defined for transmission in one CPU cycle.
• ResendTMO
– if ReceiveTMO $\geq$ 2 x WDZ, then ResendTMO = ReceiveTMO $\div$ 2, or ResendTMO = ResponseTime, whichever is greater
– if ReceiveTMO < 2 x WDZ, then ResendTMO = ReceiveTMO
• AckTMO = 0
• ProdRate = 0</td></tr>
</table>

(1) The network can be shared with other applications, if sufficient bandwidth is provided.

Peer-to-Peer Profile II: Fast & Noisy

This profile provides fast data throughput for applications which require fast data update rates. It is good for applications which require the shortest feasible Worst-Case Reaction Time where minor loss of messages can be corrected.

Fast & Noisy Characteristics

<table>
<tr><td rowspan="3">Minimum Ethernet network requirements</td><td>Fast</td><td>100 Mbit technology (100 Base TX), if HH network profile Fast & Cleanroom is selected.
10 Mbit technology (10 Base T), if HH network profile Medium & Cleanroom is selected.</td></tr>
<tr><td>Switched</td><td>Fast Ethernet (full duplex recommended) LAN switches, if HH network profile Fast & Cleanroom is selected.
10 MBit hubs, if HH network profile Medium & Cleanroom is selected.
Or use switches integrated into the GuardPLC 1600/1800 controllers.</td></tr>
<tr><td>Noisy</td><td>Low probability for loss of messages.
Time for ≥ 1 repetitions.</td></tr>
<tr><td>Characteristics of the communication path</td><td colspan="2">Minimum delays
ResponseTime ≤ReceiveTMO ÷ 2 (otherwise ERROR)</td></tr>
<tr><td rowspan="3">Variables</td><td>ResponseTime</td><td>manually set in the Peer-to-Peer Editor</td></tr>
<tr><td>ReceiveTMO</td><td>manually set in the Peer-to-Peer Editor</td></tr>
<tr><td>WDZ</td><td>manually set in the controller properties</td></tr>
<tr><td>Suitable HH network profile</td><td colspan="2">Fast
Medium (≤10 controllers in a Token Group)</td></tr>
<tr><td>Peer-to-Peer parameter presets</td><td colspan="2">• QueueLen = 2
• Communication Time Slice large enough to process and send all data defined for transmission in one CPU cycle.
• ResendTMO
– if ReceiveTMO ≥ 2 x WDZ, then ResendTMO = ResponseTime (≥ 1 Resend possible)
– if ReceiveTMO < 2 x WDZ, then ERROR
• AckTMO = 0
• ProdRate = 0</td></tr>
</table>

Peer-to-Peer Profile III: Medium & Cleanroom

This profile provides medium data throughput for applications where only a moderate data update rate is required and where the Worst Case Reaction Time is not a critical factor. It is well-suited for virtual private networks (VPN), where data exchange is slow due to safety devices (firewalls, encoding/decoding), but error-free.

TIP Normally use the profile Medium & Noisy (see page 12-21).

Medium & Cleanroom Characteristics

Minimum Ethernet network requirements	Medium or Fast	10 MBit (10 Base T) or 100 Mbit technology (100 Base TX) or network with both 10 MBit and 100 MBit components. LAN switches required.
	Clean	No loss of data due to traffic overload, harsh environmental conditions or network defects. Time for ≥ 0 repetitions.
Characteristics of the communication path	Moderate delays ResponseTime ≤ReceiveTMO (otherwise ERROR)	
Variables	ResponseTime	manually set in the Peer-to-Peer Editor
	ReceiveTMO	manually set in the Peer-to-Peer Editor
	WDZ	manually set in the controller properties
Suitable HH network profile	Fast Medium (≤10 controllers in a Token Group)	
Peer-to-Peer parameter presets	• QueueLen = 3 • Communication Time Slice large enough to process and send all data defined for transmission in one CPU cycle. • ResentTMO – if ReceiveTMO ≥ 2 x WDZ, then ResendTMO = ResponseTime (≥ 0 Resends possible) – if ReceiveTMO < 2 x WDZ, then ResendTMO = ReceiveTMO • AckTMO = ReceiveTMO or AckTMO = AckTMOMax, whichever is smaller • ProdRate = ResponseTime ÷ 4	

Peer-to-Peer Profile IV: Medium & Noisy

The Medium and Noisy profile provides medium data throughput for applications where only a moderate data update rate is required. It is good for applications where the Worst Case ReactionTime is not a critical factor. Minor loss of messages can be corrected.

Medium & Noisy Characteristics

Minimum Ethernet network requirements	Medium or Fast	10 MBit (10 Base T) or 100 Mbit technology (100 Base TX) or network with both 10 MBit and 100 MBit components. Usage of hubs possible.
	Noisy	Low probability for loss of messages. Time for ≥ 1 repetitions.
Characteristics of the communication path	Moderate delays ResponseTime ≤ReceiveTMO ÷ 2	
Variables	ResponseTime	manually set in the Peer-to-Peer Editor
	ReceiveTMO	manually set in the Peer-to-Peer Editor
	WDZ	manually set in the controller properties
Suitable HH network profile	Medium or Fast	
Peer-to-Peer parameter presets	• QueueLen = 3 • Communication Time Slice large enough to process and send all data defined for transmission in one CPU cycle. • ResendTMO – if ReceiveTMO ≥ 2 x WDZ, then ResendTMO = ResponseTime (≥ 1 Resend possible) – if ReceiveTMO < 2 x WDZ, then ERROR • AckTMO = ReceiveTMO or AckTMO = AckTMOMax, whichever is smaller • ProdRate = ResponseTime ÷ 4	

Peer-to-Peer Profile V: Slow & Cleanroom

This profile provides low data throughput for applications where only a low data update rate is required from remote controllers, via communication paths, whose conditions cannot be predicted by the user.

TIP Normally use the profile Slow & Noisy (see page 12-23).

Slow & Cleanroom Characteristics

Minimum Ethernet network requirements	Slow	Primarily for data exchange via ISDN, leased line or slow line-of-sight radio link.
	Clean	No loss of data due to traffic overload, harsh environmental conditions or network defects. Time for ≥ 0 repetitions.
Characteristics of the communication path	Moderate to long delays ResponseTime ≤ReceiveTMO, otherwise ERROR	
Variables	ResponseTime	manually set in the Peer-to-Peer Editor
	ReceiveTMO	manually set in the Peer-to-Peer Editor
	WDZ	manually set in the controller properties
	N	number of link partners a controller can talk to defined in the Peer-to-Peer Editor
Suitable HH network profile	Medium or Fast	
Peer-to-Peer parameter presets	• QueueLen = 4 • Communication Time Slice large enough to process and send all data defined for transmission in one CPU cycle. • ResendTMO – if ReceiveTMO ≥ 2 x WDZ, then ResendTMO = ResponseTime (≥ 0 Resends possible) – if ReceiveTMO < 2 x WDZ, then ResendTMO = ReceiveTMO • AckTMO = ReceiveTMO or AckTMO = AckTMOMax, whichever is smaller • ProdRate = ResponseTime ÷ 4	

Peer-to-Peer Profile IV: Slow & Noisy

This profile provides low data throughput for applications where only low data update rates are required. It is primarily for data exchange via poor quality telephone lines or distorted radio links.

Slow & Noisy Characteristics

<table>
<tr><td rowspan="2">Minimum Ethernet network requirements</td><td>Slow</td><td>Data transfer via telephone, satellite link, radio etc.</td></tr>
<tr><td>Noisy</td><td>Low loss of data due to distortions on the communication path or network defects.
Time for ≥ 1 repetitions.</td></tr>
<tr><td>Characteristics of the communication path</td><td colspan="2">Moderate to long delays
ResponseTime ≤ReceiveTMO ÷ 2, otherwise ERROR</td></tr>
<tr><td rowspan="2">Variables</td><td>ResponseTime</td><td>manually set in the Peer-to-Peer Editor</td></tr>
<tr><td>ReceiveTMO</td><td>manually set in the Peer-to-Peer Editor</td></tr>
<tr><td>Suitable HH network profile</td><td colspan="2">Medium or
Fast</td></tr>
<tr><td>Peer-to-Peer parameter presets</td><td colspan="2">• QueueLen = 4
• Communication Time Slice large enough to process and send all data defined for transmission in one CPU cycle.
• ResendTMO
– if ReceiveTMO ≥ 2 x WDZ, then ResendTMO = ResponseTime (≥ 1 Resend possible)
– if ReceiveTMO < 2 x WDZ, then ERROR
• AckTMO = ReceiveTMO or AckTMO = AckTMOMax, whichever is smaller
• ProdRate = ResponseTime ÷ 4</td></tr>
</table>

Configuring Peer-to-Peer Communication

Using This Chapter

Using Peer-to-Peer communication, you can exchange signals between controllers by dragging signals onto pages that create controller-to-controller connections. For example: Controller 1 could send three signals (out1, out2, and out3) to Controller 2. Controller 2 can then use these signals as inputs within its function block code.

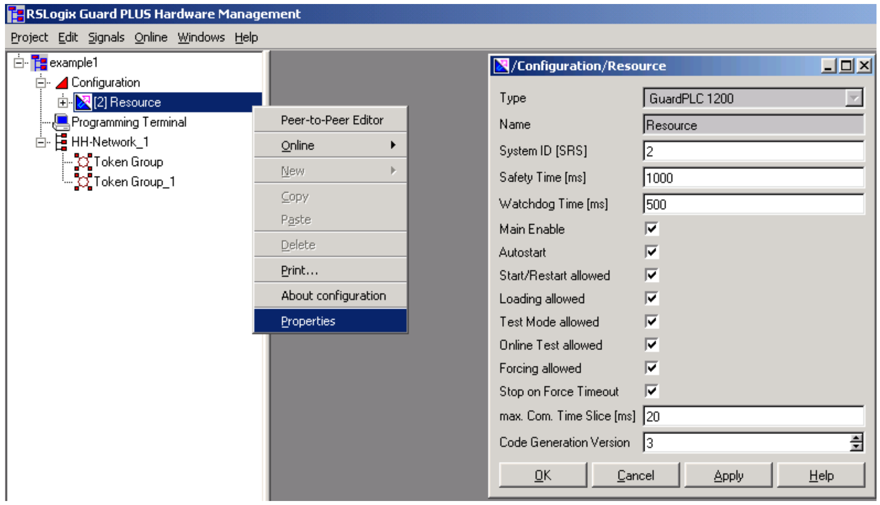
Considerations for Using Peer-to-Peer

Before you start a project that exchanges data between several controllers, you should become familiar with the requirements of your application. Questions about the network design, which should be answered prior to developing the project, are:

- Is timing a critical factor of the application? This is the most important question!
- How many controllers will be involved?
- Is it necessary to establish an Ethernet network exclusively for the application, or can an existing network be shared?
- How far away from each other are the controllers?
- Are transportation media, other than Ethernet, needed (such as telephone lines, radios, fiber optics, etc.)?
- Is it necessary for each controller to communicate with all other controllers?
- Can some functions of the application be grouped and executed separately by an isolated group of controllers (Token Group)?

Setting Peer-to-Peer Controller Properties

Right-click on *Resource* and select *Properties*. Set the timing parameters and switches according to the requirements of your application.



The “Communication Time Slice” and “Code Generation Version” settings are needed for Peer-to-Peer network parameterization.

Communication Time Slice

The Communication Time Slice is the time in milliseconds reserved for a controller to carry out and complete all communication tasks in one CPU cycle.

The minimum Communication Time Slice depends on the number of communication connections (n) a controller has.

The minimum Communication Time Slice (CTS_{min}) is calculated as follows:

For $n \leq 13$:	CTS_{min} $(n \leq 13) = n \times 1\ ms + 4\ ms$
For $n > 13$:	CTS_{min} $(n > 13) = n \times 1.3\ ms$

IMPORTANT Do not set the Communication Time Slice below the calculated value. If the Communication Time Slice is too small, it takes more than one CPU cycle to carry out the pending communication tasks. Therefore, more time is needed to complete the communication tasks, which degrades performance and could result in a communication shutdown due to a communication timeout (ReceiveTMO).

The time actually needed for communication adds to the CPU cycle time. A short Communication Time Slice limits the communication time to a low value. This prevents the CPU cycle time from being noticeably influenced by network occurrences. Although a Communication Time Slice well above the minimum value may result in cycle time on the local machine slowing down a bit if network traffic is heavy, it is not necessarily negative.

If you are transferring safety I/O over the network, you need a Communicatin Time Slice high enough to guarantee that the communications are completed every cycle. If it takes more than one cycle to read/write safety I/O, your safety time will need to increase to compensate.

If you are only transferring status data over the network, then a lower Communication Time Slice is permissible, because it leaves more time in the cycle for your program to run. It’s likely to be acceptable even if it takes more than one cycle to read the status.

Check the CPU short-term diagnostics for any “Time Slice expired” entries and increase the Communication Time Slice if necessary, before the application goes into regular operation. In the “Statistics” of the Control Panel, “Number of Time Slices” higher than 1 also indicate a Communication Time Slice that is too short. “Number of Time Slices” indicates the number of cycles it took for communications to complete.

The maximum Communication Time Slice depends on the application and is calculated as follows:

WDZ ≥ Communication Time Slice (max) + Application Execution Time

In other words, the Communication Time Slice plus Application Execution Time must not exceed the Watchdog Time.

EXAMPLE

If the controller on page 13-2 has 10 connections, the minimum Communication Time Slice is:

CTS_{min} = 10 x 1 ms + 4 ms = 14 ms.

CTS_{min} is increased by 6 milliseconds to provide a safety margin.

CTS_{min} = 20 ms

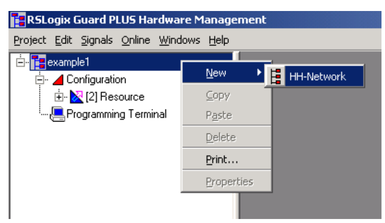
With a Watchdog Time of 500 ms, this leaves 480 ms for the application to be executed.

Code Generator Version

To compile the logic correctly for your type of controller, set Code Generator Version to "3" for RSLogix Guard PLUS software. Set to version 2 for RSLogix Guard software.

Create a Peer-to-Peer Network

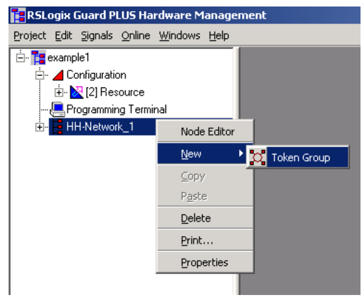
To create a Peer-to-Peer network, right-click on the project in the Hardware Management window and select *New →HH-Network*.



You can right-click on *HH-Network* and *Rename* the entry, if desired.

Create Token Group(s)

A single Token Group is automatically created with the HH network. If you need more, create Token Groups by right-clicking on *HH-Network* and selecting *New →Token Group*.



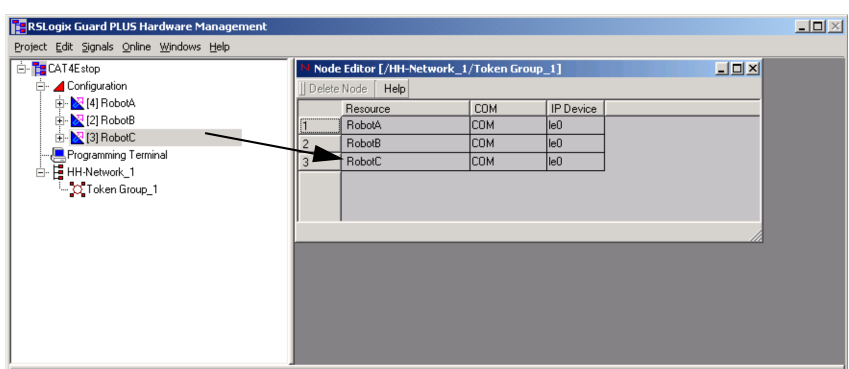
Expand the *HH-Network*, right-click on the Token Group(s) and *Rename* the Token Group(s), if desired.

Add Controllers to Token Group(s)

A controller must be a member of only one Token Group. To add a controller to a Token Group:

1. Expand the *HH-Network*, right-click on a Token Group and select *Node Editor*. The Node Editor is empty when you open it for the first time.

2. Click on a controller in the tree view and drag and drop it in the Node Editor.



Configure Token Group(s)

1. Right-click on the Token Group and select *Properties*. In the "HH-Network/Token Group" window select a profile.

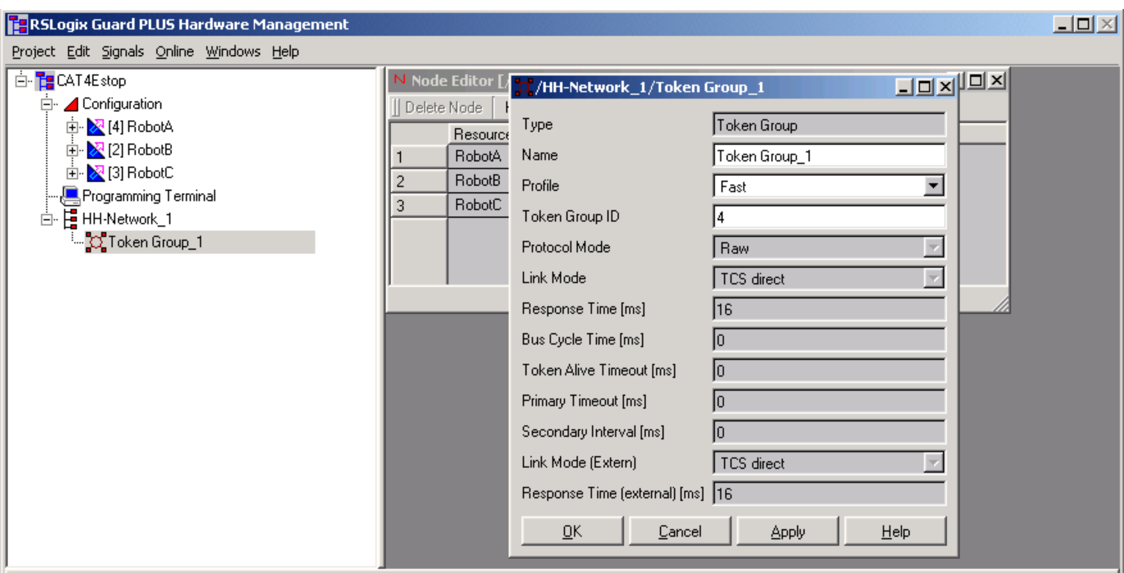
   For a description of the HH-Network profiles, see page 12-10. In general, "Fast" works with most network topologies.

2. Enter a Token Group ID.

   The Token Group ID must be greater than 0. If you create more than one Token Group, each Token Group must have a unique ID.

   Do not make changes to the other settings in this window.

   See page 12-3 for the description of the HH protocol parameters.



IMPORTANT You must select identical profiles for Token Groups that you want to interconnect. If Link Mode (External) does not match, communication between Token Groups is impossible.

Design the Logic

Create Peer-to-Peer Signals

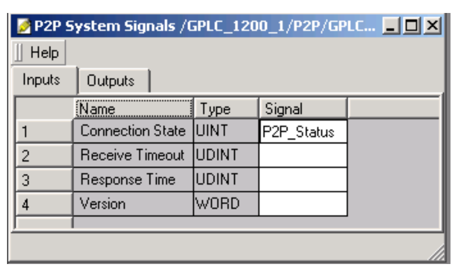
Signals are transferred among controllers over the Peer-to-Peer network. Consider the following when creating signals:

- You can create as many signals as you need in the logic for all controllers.
- You can add signals anytime.
- Signals with the same name can be used on more than one controller without influencing each other (LOCAL variable), as long as they are not exchanged via network.
- Signals which are intended for network exchange, must have the same name on the participating controllers. Whether a signal is written to or read from the network is defined in the Peer-to-Peer Editor as explained in "Configure Peer-to-Peer Communication" on page 13-11.

Use Peer-to-Peer System Signals

The status of the Peer-to-Peer communication as well as some timing parameters can be evaluated in the user program by means of system signals. Furthermore, the user program can control how a Peer-to-Peer connection is set up.

Input System Signals



The following system signals can be used as inputs for the application:

- **Connection State**. Using the Connection State system signal of the Peer-to-Peer Editor, the user program can evaluate the status of the communication between two controllers. The following table shows the possible values for the Connection State system signal and the corresponding status.

Value	Status	Explanation
0	CLOSED	Communication path is closed. No attempt to connect.
1	TRY_OPEN	Communication path is closed. Attempt to connect.
2	CONNECTED	Communication path is open. No attempt to connect.

- **Receive Timeout**, in milliseconds, is set by the user. For more information see Receive Timeout (ReceiveTMO) on page 12-8 and "Define Peer-to-Peer Parameters" on page 13-13.
- **Response Time**, in milliseconds, is the actual value of the last answer message and is identical to RspT last in the P2P status of the Control Panel. For more information, see "Reconfigure ResponseTime" on page 13-22.
- **Version** indicates the CRC for the Peer-to-Peer configuration between two controllers. The CRC must be identical in order to establish communication.

Output System Signal

Using the output system Connection Control signal, the user program can control how the Peer-to-Peer connection is set up.

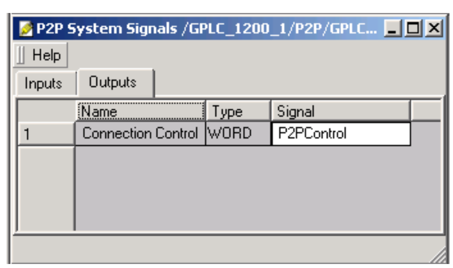


Table 13.1 Connection Control Values

Value	Setting	Explanation
0x0000	AUTOCONNECT	After loss of Peer-to-Peer communication, the controller tries to re-establish communication in the next CPU cycle. This is the standard mode of operation.
0x0100	TOGGLE_MODE 0	These modes allow automatic connect with DISABLE after loss of communication. If TOGGLE_MODE is 0 and communication is lost (Connection State = CLOSED), a connect is performed only after TOGGLE MODE is set to 1 by the user program. If TOGGLE_MODE is 1 and communication is lost, a connect is performed only after TOGGLE_MODE is set to 0 by the user program.
0x0101	TOGGLE_MODE 1	
0x8000	DISABLED	Peer-to-Peer communication is disabled. No attempt to connect.

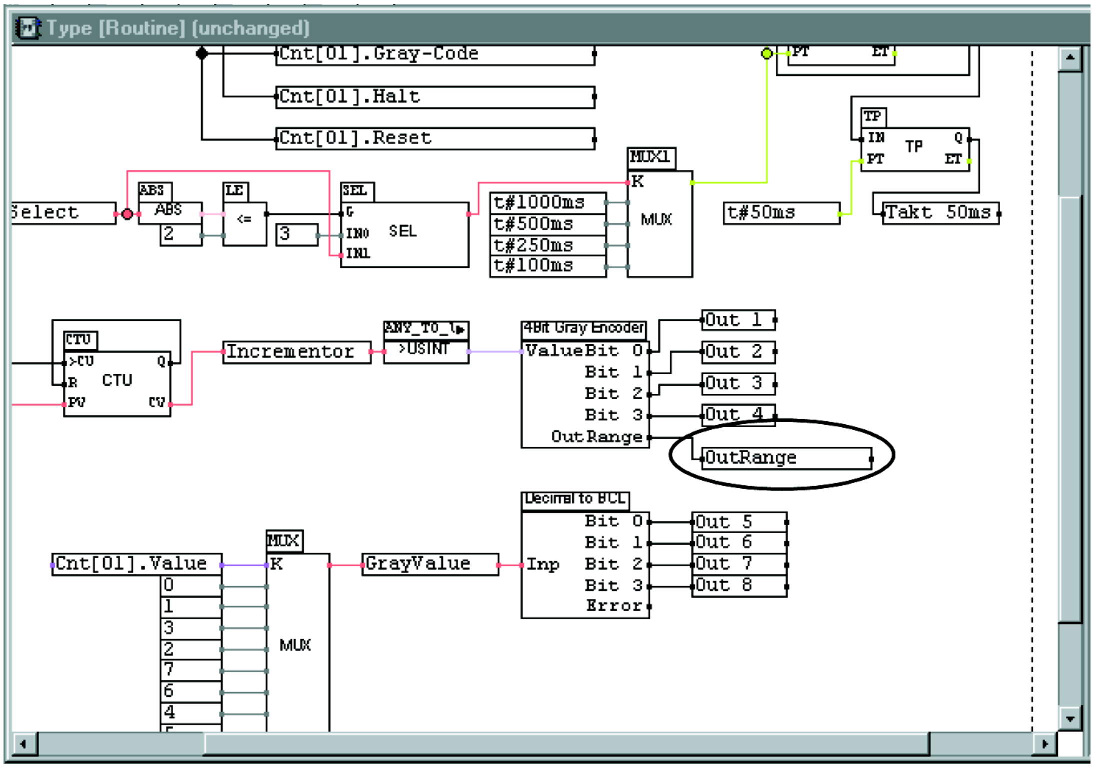
IMPORTANT If the P2PControl signal, in the illustration on page 13-8, is set to 32768, Peer-to-Peer communication is disabled. If Connection Control is not set by the application, the default is 0 and Autoconnect is enabled.

Design the Logic for all Controllers

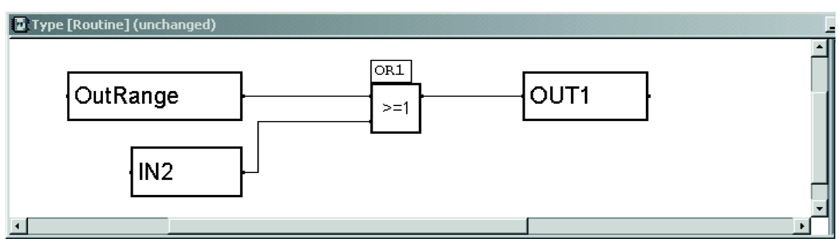
Design the logic for the controllers, considering the variables intended for network exchange.

The following examples show part of the routines for controllers Robot A and Robot B, respectively. To evaluate the state of the “OutRange” signal in Robot B, use the same signal name (OutRange) as an input for the logic of Robot B. OutRange is sent over Ethernet, via Peer-to-Peer, from Robot A to Robot B, which uses it as an input.

Design Logic for Robot A



Design Logic for Robot B



Configure Peer-to-Peer Communication

As discussed in the following sections, you configure Peer-to-Peer Communication by:

- Defining Controller Connections
- Assigning the HH-Network
- Selecting a Peer-to-Peer Profile
- Defining Peer-to-Peer Parameters, and
- Defining Process Signals for Exchange

Define Controller Connections

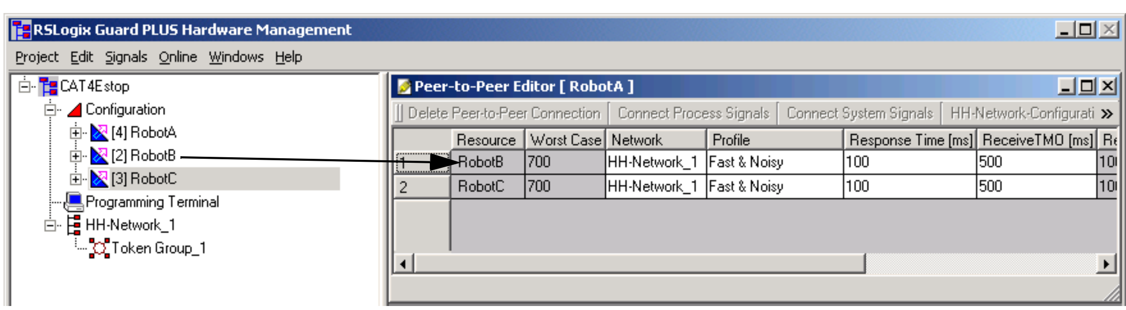
To define all of the controllers each controller can communicate with:

1. Right-click on the resource you want to define controller connections for and select *Peer-to-Peer Editor*.

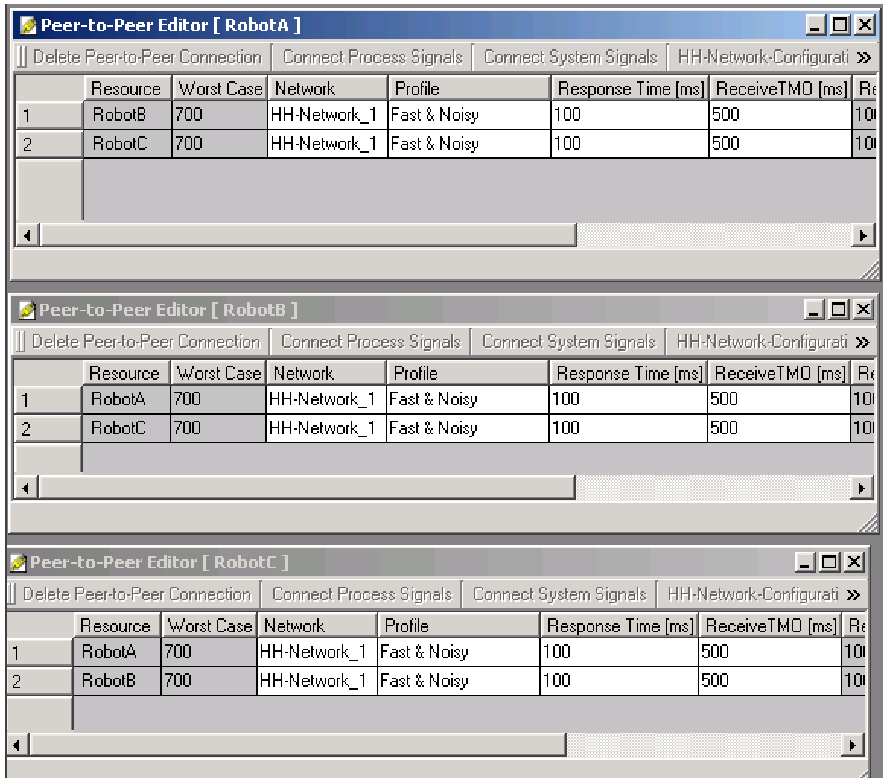
   The title bar of the Peer-to-Peer Editor shows the name of the selected controller. When the Peer-to-Peer Editor is opened for the first time, it does not contain any entries.

2. In the project tree, click on a resource and drag and drop it in the Peer-to-Peer Editor. Repeat this step to add more controller connections.

   In the example below, RobotA (title bar) has a connection to RobotB and RobotC. Because the return path is automatically added, you do not need to drag RobotA onto the Peer-to-Peer editors of RobotB or RobotC.



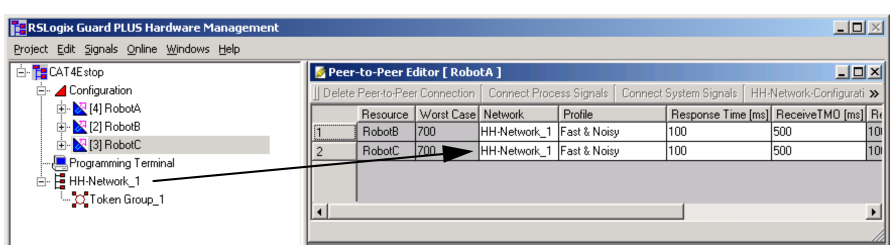
The following example shows how the three Peer-to-Peer editors would appear if connections existed between all three controllers.



Assign HH-Network

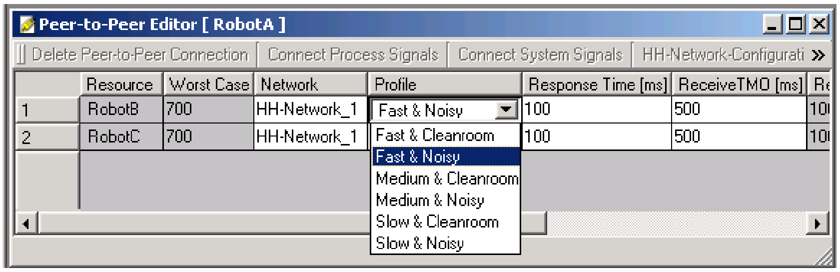
Peer-to-Peer communication requires the HH-Network, which must be entered in the Peer-to-Peer Editor.

To assign the HH-Network, click on the HH-Network in the tree view and drag and drop it in the *Network* column of the Peer-to-Peer Editor. The return path is automatically updated with the HH-Network.



Select Peer-to-Peer Profile

1. Click in the *Profile* column and select one of the profiles. Make sure that the profile is suitable for your network topology and matches the HH profile. See page 12-10 for a detailed description of all the profiles.



2. Click outside the table or press the Return key to activate the selection. The profile of the return path is automatically updated with the new profile.

Define Peer-to-Peer Parameters

The most important timing parameter of a safety related installation is the Safety Time. Safety Time is the time a process can run with incorrect controller outputs without affecting the safety of the process (see the *GuardPLC Controller Systems Safety Reference Manual*, publication number 1755-RM001 for more details).

The Worst Case Reaction Time (T_R) is the time within which two linked controllers must detect the occurrence of a physical input signal at PES_1 and put out the resulting physical output signal at PES_2.

To guarantee the integrity of the application, the requirement below must always be fulfilled:

T_R < Safety Time

When you select a Peer-to-Peer profile, most parameters are automatically preset. Because ReceiveTMO (safety-relevant) is part of the Worst Case ReactionTime T_R (see Peer-to-Peer Protocol Parameters on page 12-6), ReceiveTMO must be calculated and set manually by overwriting the default value in the Peer-to-Peer Editor.

For profiles where ProdRate = 0 (Fast & Cleanroom, Fast & Noisy), ReceiveTMO is:

$$\text{ReceiveTMO} = T_R - 2 \times \text{WDZ}(\text{PES}_1) - 2 \times \text{WDZ}(\text{PES}_2)$$

For profiles where ProdRate ≠ 0, ReceiveTMO is:

$$\text{ReceiveTMO} = [T_R - 3 \times \text{WDZ}(\text{PES}_1) - 2 \times \text{WDZ}(\text{PES}_2)] \div 2$$

Calculate the ReceiveTMO with the suitable formula and overwrite the default value in the Peer-to-Peer Editor.

Profile	Response Time [ms]	ReceiveTMO [ms]
Fast & Cleanroom	100	500

In first approximation, the ResponseTime can be calculated as:

$$\text{ResponseTime} = \text{ReceiveTMO} \div 2$$

Overwrite the default value of the ResponseTime with the calculated value.

TIP Setting the ResponseTime this way allows the controller to resend a message, in case of unexpected message loss. For best network performance, the ReceiveTMO and the ResponseTime are optimized after the project has been compiled, loaded and started on the controllers. At that time, the actual ResponseTimes and the actual cycle times can be read in the Control Panel.

Define The Signals to Exchange Between Each Controller Connection

1. Right-click on a resource in the project tree, select *Peer-to-Peer Editor*. The Peer-to-Peer Editor opens.

2. Click on a line number (left-most column) in the Peer-to-Peer Editor table. This selects a controller with which the controller, named in the headline of the Peer-to-Peer Editor, exchanges data.

3. Open the Signal Editor (select *Signals →Editor*).

4. Click on the *Connect Process Signals* button in the Peer-to-Peer Editor.

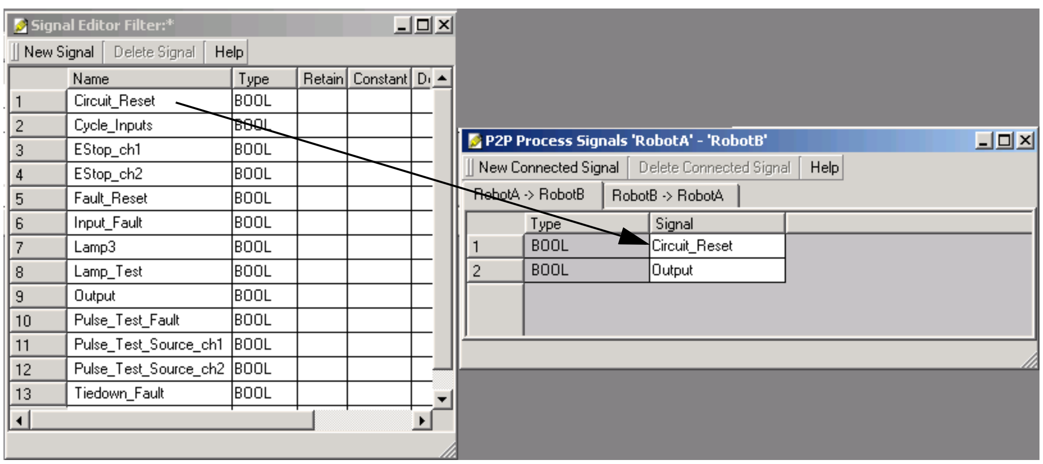
5. Arrange the Signal Editor and the Peer-to-Peer (P2P) Process Signals window side by side. When you open it for the first time, the P2P Process Signals window is empty.

6. Using the tabs below the button bar of the P2P Process Signals, select the direction of data exchange.

   In the example below, the direction of data exchange is from RobotA to RobotB.

7. In the Signal Editor, click on a signal name and drag & drop it in the P2P Process Signals.

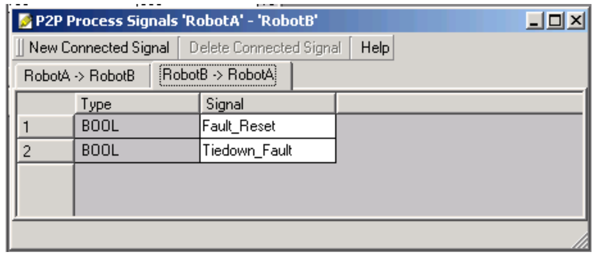
   You can also add signals using *New Connected Signals* button. this creates a new line in the list in which you must enter the case-sensitive signal name exactly as defined in the Signal Editor.



TIP Sending a signal from one controller to another ($PES_1 \rightarrow PES_2$) makes the value of this signal available in PES_2. To process this value in the logic of PES_2, identical signal names must be used in the logic of both PES_1 and PES_2.

8. Change the direction of data exchange with the tab and define the return signals.

The illustration below shows the signals which RobotB sends to RobotA.

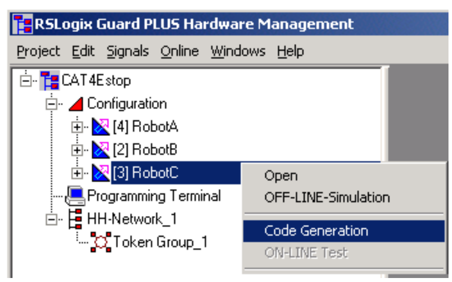


Compile and Download

Compile Logic

If changes, such as adding or deleting a tag, are made to a connection between two controllers, the code must be recompiled for both controllers.

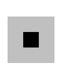
To compile logic, right-click on a resource (controller) in the RSLogix Guard PLUS Project Management window, and select *Code Generation*.



If code generation is not successful, carefully check the Error-state viewer in the Hardware Management Window for error messages and correct the errors.

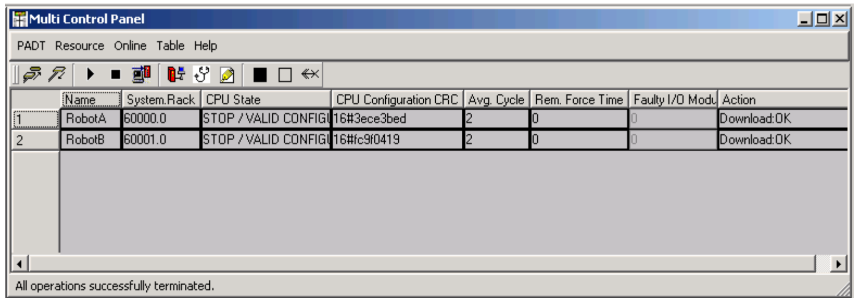
Start Download

1. Using the Multi-Control Panel, click the *Select all* button  to select all controllers.

2. Click *Stop*  to make sure that all controllers are in STOP mode.

3. Click *Download* to start the simultaneous download for all selected controllers. The “Action” column shows the command which is currently executed or a short status message.

   In the example below, the downloads have completed successfully.



4. After successful download, the CPU Status is STOP/VALID CONFIGURATION. Select all controllers again if necessary and click *Coldstart* to start the application.

Network Optimizing

With the initial network settings made in the HH protocol and Peer-to-Peer protocol, communication is likely to work, but the settings can be optimized for homogenous network load and faster message exchange.

IMPORTANT

If there is no real need to reduce Worst Case ReactionTime, do not make changes to the WDZ and the ReceiveTMO!

Only optimize the ResponseTime.

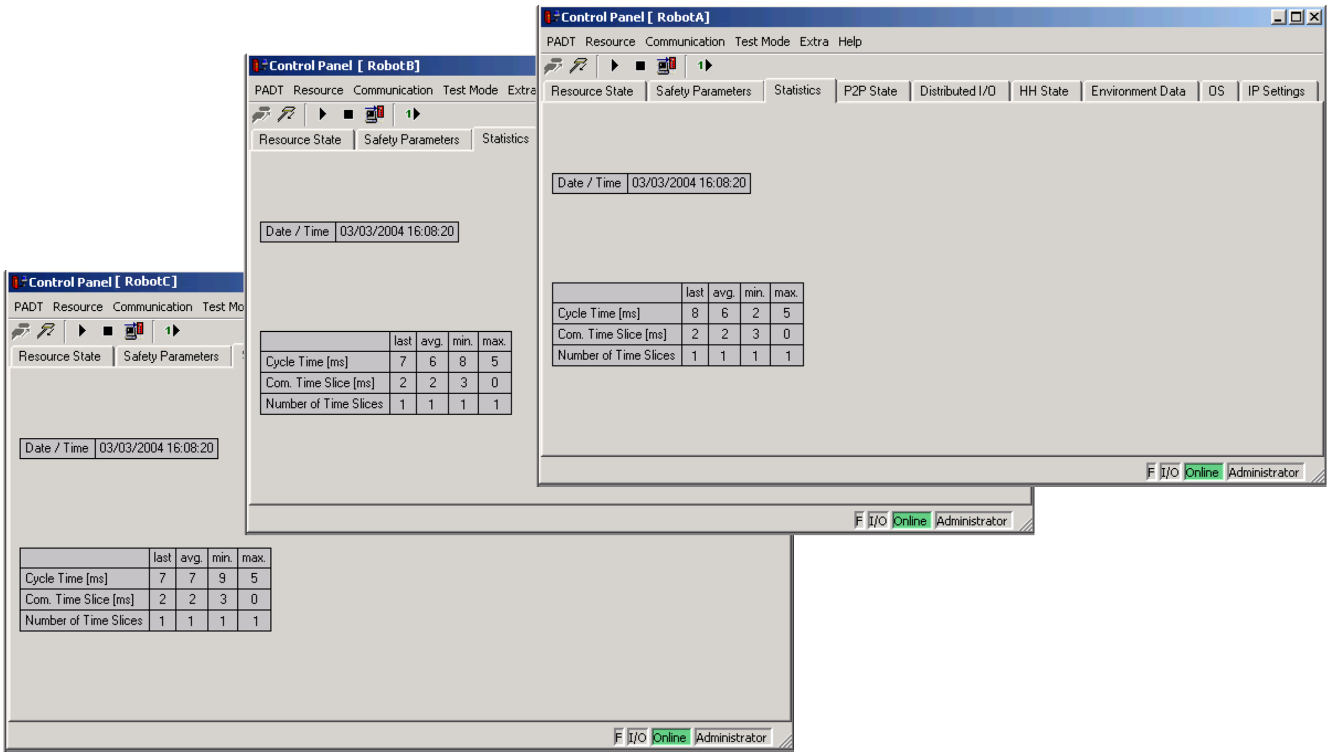
A high WDZ or ReceiveTMO does not degrade performance, but an optimized ResponseTime increases availability.

Before starting the optimization steps, let the project run for several hours. Test as many operating conditions as possible to address timing factors that may prevent a project from running after optimization.

Check Routine Timing

1. In the Multi Control Panel, select all controllers and click the *Control Panel* button.

2. In the Control Panels of each controller, select the *Statistics* tab.



3. Write down the maximum "Cycle Time" for each controller.

4. Write down the maximum "Com. Time Slice" for each controller.

IMPORTANT

Before you continue to optimize settings, make sure that *Number of Time Slices* (see above) is not greater than 1. If *Number of Time Slices max.* is greater than 1, more than one CPU cycle is needed to carry out all communication tasks.

In this case, you need to determine if it is permissible for communications to take multiple cycles to complete. This depends on how many cycles can be completed within the safety time.

If you need to increase the *Com. Time Slice*, start the code generator again, and download and start the new routine on the controller.

Reconfigure Watchdog Time

To optimize the Watchdog Time to the lowest possible value, you must know the maximum CPU cycle time. Cycle Time max., as displayed in the Statistics of the Control Panel (see page 13-19), is the value that occurred so far, but is not necessarily the maximum value that can occur depending on network and process conditions.

If the maximum Cycle Time cannot be estimated, run the project for several hours and under as many conditions as possible.

To reconfigure Watchdog Time:

1. In the project tree, right-click on the first resource and select *Properties*.

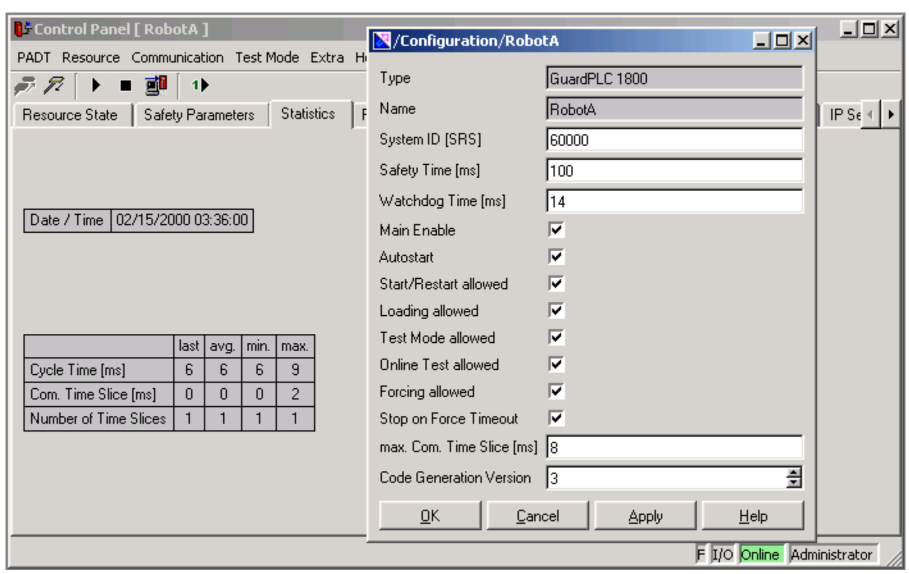
2. Calculate a Margin of Safety, MoS:

   MoS = 0.1 x (Cycle Time max.)
   MoS should be at least 6 ms. If MoS < 6 ms, then
   MoS = 6 ms

3. Readjust the Watchdog Time:

   Watchdog Time = (Cycle Time max.) + (MoS)

   In the example on the following page, the new Watchdog Time is: 8 ms + 6 ms = 14 ms.



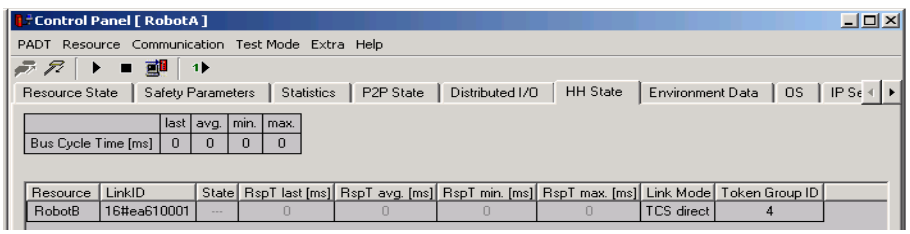
4. For all controllers in your project, re-adjust the Watchdog Times to their individual optimum values.

TIP	After these modifications, you must re-compile the project with the Code Generator and download the routines in the controllers again.

5. Start the project and let it run for a while. If you encounter controller errors due to a Watchdog Time that is too short, increase the Watchdog Time. Otherwise, continue with the network optimization.

Check HH Status

In the Control Panel, click on the *HH Status* tab.



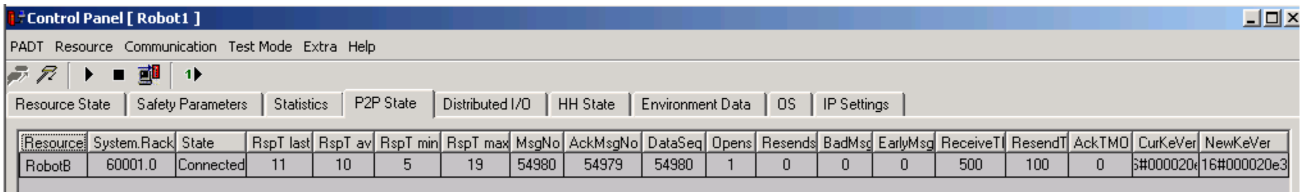
The HH Status displays the following information:

Parameter	**Explanation**
Bus Cycle Time	Time in milliseconds for a Token cycle. The value is 0, if Token Passing is off (any Cleanroom profile).
Resource	Name of the controller
LinkId	Controller network ID
State	Status of the communication
RspT	• If Link Mode is “TCS direct” (Token Passing OFF), RspT is the ResponseTime of the HH profile for a message from $PES_1 \rightarrow PES_2 \rightarrow PES_1$, based on the network hardware and topology. This parameter cannot be changed by the user. • If Link Mode is “TCS TOKCYC” (Token Passing ON), RspT is part of the Bus Cycle Time.
Link Mode	• “TCS direct” when Token Passing is OFF. • “TCS TOKCYC” when Token Passing is ON.
Token Group ID	ID of the Token Group

Read the RspT min parameter. This is the minimum time needed for the communication modules (COM) of two controllers to talk to each other. Refresh RspT values with *Communication →Update HH State*, if Token Passing is OFF.

Check Peer-to-Peer Status

In the Control Panel, click on the *P2P Status* tab.



The P2P Status displays the following information:

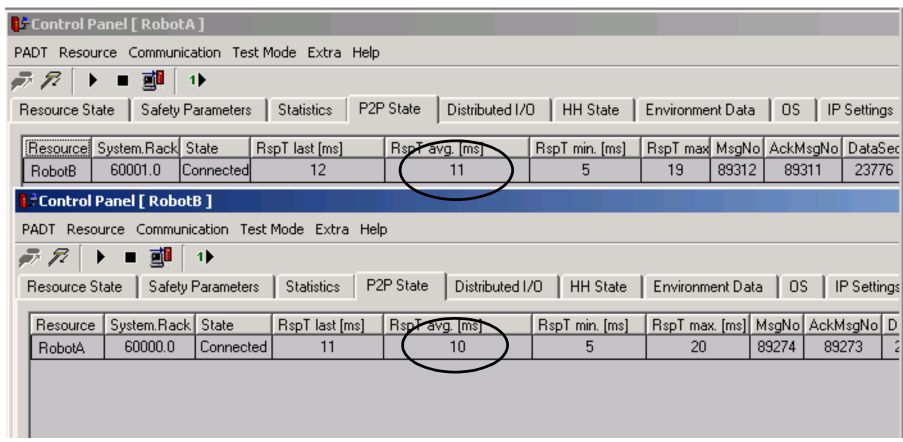
Parameter	Explanation
Resource	name of the controller
System.Rack	network ID of the controller
State	Status of the communication
RspT (last, avg, min, max)	Measured ResponseTime for a message from $PES_1 \rightarrow PES_2 \rightarrow PES_1$, based on the network hardware, CPU cycle time, and Peer-to-Peer profile. This parameter will be optimized later.
MsgNr	Counter (32-bit resolution) for all messages sent to a controller. *In the illustration above, Robot A has sent message no. 54980 to Robot B.*
AckMsgNr	The number of the received message that the controller has to acknowledge. *In the illustration above, Robot A has* acknowledged *message no. 54979 from Robot B.*
DataSeq	Counter (16-bit resolution) for sent messages, which contain process data. *In the illustration above, Robot A has sent data message no. 54980 to Robot B.*
Opens	Number of successful connects to a controller. A figure higher than 1 indicates that a controller dropped out and has been reconnected.
Resends	Counter (32-bit resolution) for messages that have been resent due to an elapsed ResendTMO.
BadMsgs	Counter (32-bit resolution) for received messages that are corrupted, or are not expected at that instant. A corrupt message, for example, is a message with a wrong sender or with a faulty CRC. An unexpected message, for example, is an "Open" command, when the controllers are already connected.
EarlyMsgs	Counter (32-bit resolution) for received messages which are not in the correct sequence. If a message drops out and is lost at the addressee, there is a gap in the received messages, and the next message comes early.
Receive Tmo	Receive Timeout as entered by the user (see Define Peer-to-Peer Parameters on page 13-13).
ResendTMO	Resend Timeout as set by the profile.
AckTmo	Acknowledge Timeout as set by the profile.
CurKeVer	CRC for the Peer-to-Peer configuration. Identical to the Peer-to-Peer system signal "Version" (see page 13-8).
NewKeVer	Reserved for future use.

Reconfigure ResponseTime

The ResponseTime initially configured in Define Peer-to-Peer Parameters on page 13-13 was derived from theoretical considerations and was chosen conservatively, to start the network running. The ResponseTime actually needed is usually much smaller than the theoretical value and can be optimized to improve network performance.

To optimize the ResponseTime:

1. Open the Control Panels for all controllers in the project and select *P2P State*. Position the horizontal slider so that you can read the ResponseTime.



2. Compare the RspT avg of two linked controllers for the forward and return path. Values for RspT avg may jump a bit.

   Watch both readings for a couple of seconds and pick the largest value. Your reading need not be accurate to the millisecond.

   Note down the larger of both values.

   The example on page 13-22 shows RespT avg for Robot A →Robot B (11 ms) and Robot B →Robot A (10 ms).

3. Compare the RspT max of two linked controllers for the forward and return paths.

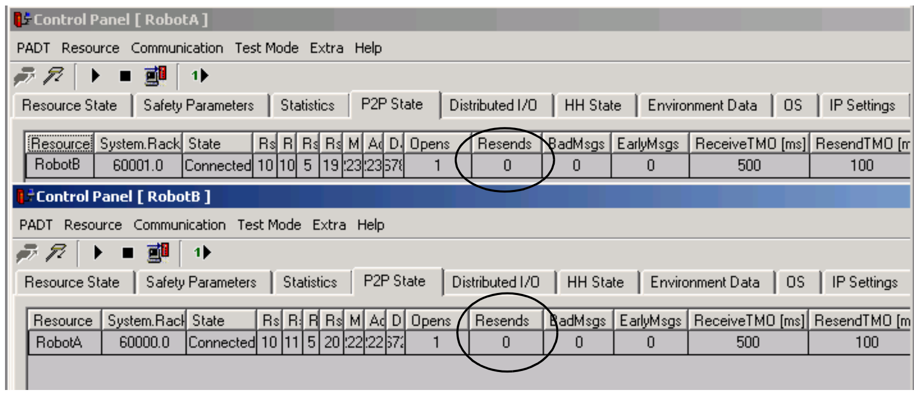
   Note down the larger of both values.

   The example on page 13-22 shows RspT max for Robot A →Robot B (19 ms) and Robot B →Robot A (20 ms).

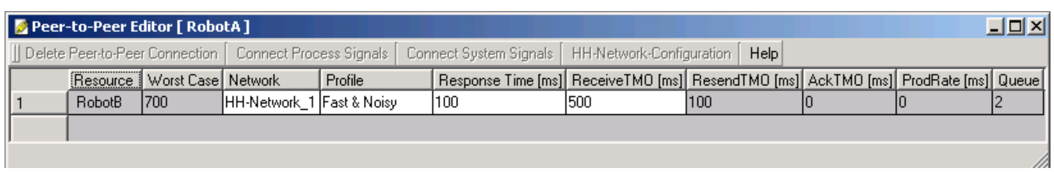
4. In the *P2P State* tab, check the entries for Resends and EarlyMsgs.

   If the entries for both Resends and EarlyMsgs are 0, no messages have been repeated. In this case, delete the noted RspT avg.

   If one or more entries for Resends or EarlyMsgs is not 0, messages have been repeated. In this case, delete the noted RspT max.



5. Enter the remaining noted value for RspT, either avg or max, in the ResponseTime of the Peer-to-Peer Editor.



Reconfigure Receive Timeout

1. Set the new ReceiveTMO to:

   ReceiveTMO = 2 x ResponseTime

2. The Worst Case Reaction Time is optimized and displayed in the Peer-to-Peer Editor (see above).

3. Compile the project and download the routines in the controllers again. Start and test your application.

Communicating with ASCII Devices

Using This Chapter

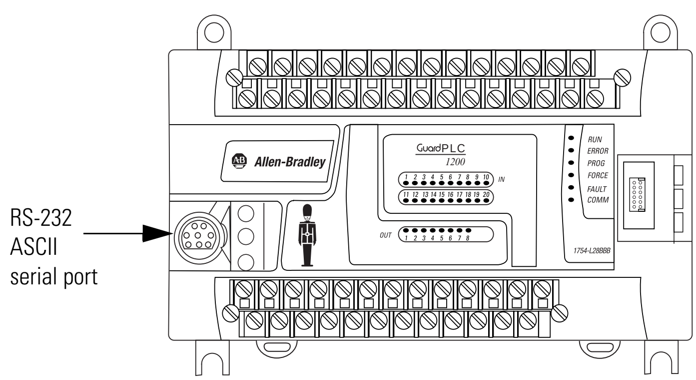
For information about:	See page
connecting the controller to an ASCII device	14-1
configuring the ASCII port	14-4
connecting signals	14-5
ASCII protocol	14-6

Connecting the Controller to an ASCII Device

For the sole purpose of sending the status of the signals from the GuardPLC to an external device, you can connect an intelligent ASCII device to the GuardPLC serial port. This ASCII connection is one-way from the GuardPLC (slave) to the master device. You cannot program the GuardPLC or change the values in the GuardPLC using this port.

To use the ASCII function, signals that you wish to send out the serial port must be connected to placeholders in the ASCII-protocol Connect Signals window. These signals are then capable of being sent out the serial port if a command string is properly received from the master. The command string includes a starting address and number of signals to be sent. The GuardPLC replies to this command string by sending the values of these signals out the serial port in an ASCII string.

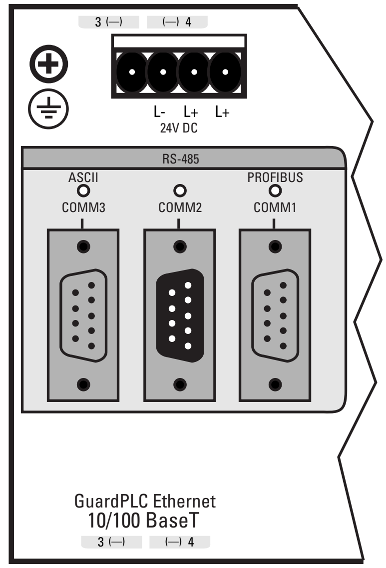
Connecting to a GuardPLC 1200 Controller



Use a 1761-CBL-PM02 series C cable to connect to the serial port. The mini-DIN connector attaches to the controller. The other end is a 9-pin D-shell connector. This mini-DIN connector is not commercially available, so you cannot make this cable.

Connecting to a GuardPLC 1600 or 1800 Controller

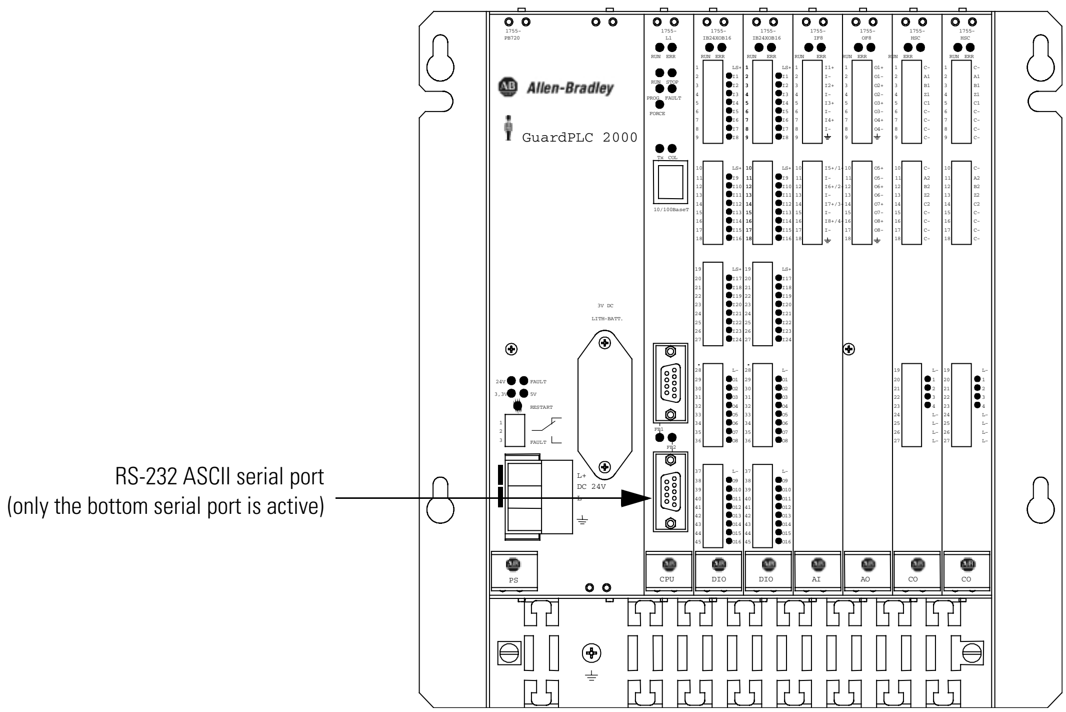
The ASCII COMM3 port location and connector pin assignment are shown below.



Connection	Signal	Function
1	---	---
2	RP	5V, decoupled with diodes
3	RxD/TxD-A	Receive/Transmit data A
4	CNTR-A	Control Signal A
5	DGND	Data reference potential
6	VP	5V, positive pole of supply voltage
7	---	---
8	RxD/TxD-B	Receive/Transmit data B
9	CNTR-B	Control Signal B

IMPORTANT The ASCII port is RS-485. You must use and electrical interface device to connect the controller to an RS-232 device.

Connecting to a GuardPLC 2000 Controller



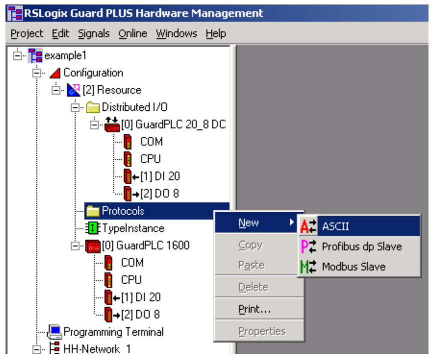
The serial port requires a 9-pin D-shell connector.

Pin:	Function:
1	none
2	send data
3	receive data
4	none
5	ground
6	none
7	RTS
8	CTS
9	none

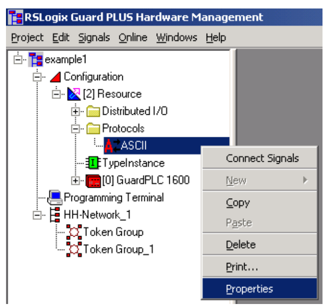
Configuring the ASCII Serial Port

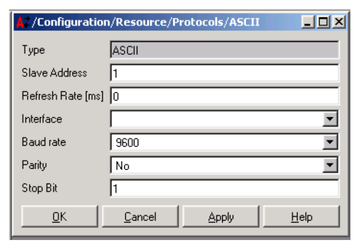
You must either create a new project or open an existing project before you can configure ASCII communications. Once the software opens a project, it automatically displays the Hardware Management window, from which you configure the ASCII port.

1. Right-click on *Protocols* and select *New →ASCII*.



2. Right-click on the *ASCII* icon and select *Properties*.





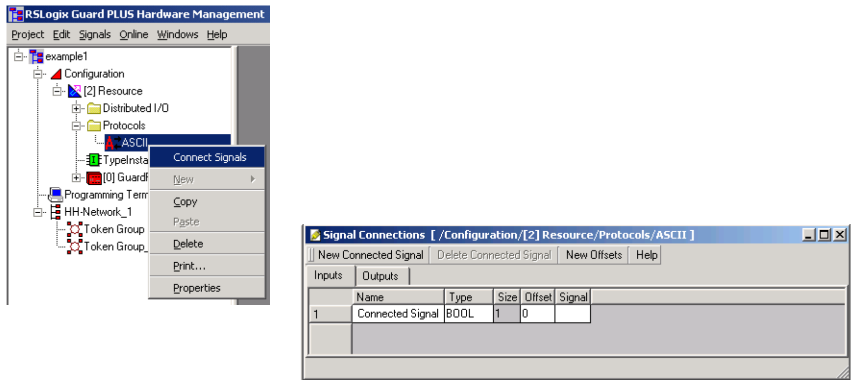
For this field:	Specify:
Slave Address	the slave address (1 to 65535) of the controller. The ASCII protocol of the controller supports only a direct point-to-point connection between the master and slave. The controller is always configured as slave. It only transfers process values via the serial interface to the master when it receives the corresponding request from the master.
Refresh Rate	the refresh rate in milliseconds for non-safe communication between the COM and CPU. The default is 0, the fastest refresh rate.
Interface	select the field bus interface to be used by the ASCII protocol (comm1, comm2, comm3). Select comm3 for GuardPLC 1600 or 1800 controllers.
Baud Rate(1)	the data transfer speed in bits/s. Select from a dropdown menu of predefined values between 300 and 115,200 bps. The default baud rate is 9600 bps.
Parity	the parity for the recognition of transfer errors. Select *No*, *Odd*, or *Even*. The default is *No* parity.
Stop Bit	either 1 or 2 stop bits for the serial data transfer. The default is 1 stop bit.

(1) Even if the baud rate is changed from 9600, the power up string is always sent out at 9600 baud.

Connecting Signals

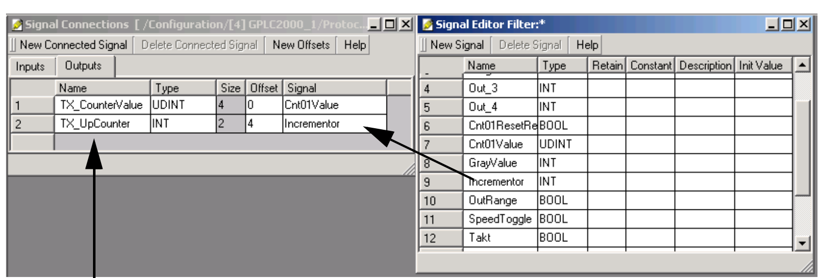
Only ASCII output signals are sent from the controller. You connect signals to the ASCII outputs to determine which signal values you want to send from the controller to the connected ASCII device.

1. Expand *Protocols*. Right-click on the *ASCII* icon and select *Connect Signals*.



If you want to:	Select this tab:
create a new signal	New Connected Signal
renumber offsets sequentially for all signals	New Offsets
delete the selected signal	Delete Connected Signal

2. Edit the output signals you want to send to the ASCII device.
   - Use the *Outputs* tab to define output values to send to the ASCII device.
   - Associate each output with a signal from the signal editor by dragging the signal from the Signal Editor to the *Signal* field on the *Outputs* tab in the ASCII Signal Connections Window.
   - See Chapter 5 for more information about defining signals.



NOTE: The signal name is only used in printouts.

TIP

The offset in the ASCII output section is numbered based on bytes. In the example, the first signal uses bytes 0, 1, 2, and 3. The second signal uses bytes 4 and 5. However, when you request these signals in the command string (see ASCII master - request below), the first signal is always 0, the second signal is always 1, the third signal is always 2, etc.

The output section automatically sorts the name field based on alphanumerical order. This does not automatically change the offsets, but if you renumber after sorting, the offsets will change and there is no undo feature. This changes the order in which the signals are sent out the serial port.

Since names are only used in printouts, you may want to enter these names in alphanumeric order to begin with. (For example signal 101, signal 102, signal 103, signal 104, etc.)

ASCII Protocol

The controller is a slave ASCII device and expects this protocol from the master device.

ASCII master - request

If the ASCII master sends a request, the slave can send a response. The master request has this format (each character is one byte):

Start Sign	Destination	Source	Function Code	Start Address	Number of Variables	End Sign
1 char	2 char	2 char	1 char	5 char	3 char	1 char

Component:	Description:
Start Sign	identifies the start of a message ^ character
Destination	unique slave address (GuardPLC controller) 01 to 99
Source	unique master address (requester) 01 to 99
Function Code	read data R character
Start Address	data start address for characters to read (offset) 00000 to 65535
Number of Variables	number of variables to read and send back to master 000 to 999
End Sign	identifies the end of a message & character

For example, this string requests the first two variables from the slave:

Start Sign	Destination	Source	Function Code	Start Address	Number of Variables	End Sign
^	15	01	R	00000	002	&

ASCII slave - controller response

If the controller receives a request from an ASCII master, it responds in this format (each character is one byte):

Start Sign	Destination	Source	Function Code	Start Address	Number of Variables	Number of Characters	Data	End Sign
1 char	2 char	2 char	1 char	5 char	3 char	4 char	maximum 10000 char	1 char

Component:	Description:
Start Sign	identifies the start of a message ^ character
Destination	unique master address (requester) 01 to 99
Source	unique slave address (GuardPLC controller) 01 to 99
Function Code	r character identifies data sent by slave E identifies error with master request
Start Address	data start address for characters to read (offset) 00000 to 65535
Number of Variables	number of variables to read and send back to master 000 to 999
Number of Characters	number of characters in the data string (This includes the "/" delimiter but not the "&" termination character.) 0000 to 9999
Data	data characters
End Sign	needed to recognize the end of a message & character

For example, this string replies to the master request for the first two variables from the slave:

Start Sign	Destination	Source	Function Code	Start Address	Number of Variables	Number of Characters	Data	End Sign
^	01	15	r	00000	002	0005	4/123	&

Every data field in the message is separated with a slash (/). The slash also counts as a character when counting the total number of characters in the data string.

TIP The reply string will have a variable number of characters if non-BOOL are used. For example, 99 is 2 characters, 100 is 3 characters. There is no leading zero.

If the master request was not received properly at the GuardPLC, the slave response is the following:

Start Sign	Destination	Source	Function Code	Start Address	Number of Variables	Number of Characters	End Sign
^	01	15	E	00000	000	0000	&

This error response is typically sent when more signals are requested than exist in the ASCII protocol output tab. For example, 10 signals were dragged to the ASCII output section, but 20 signals were requested in the command string.

Data type formats

Follow these formats for sending different data types:

Data Type:	Format:		Example:
BOOL	Description: Size: Range:	boolean 1 character 1 = true; 0 = false	0 1
SINT	Description: Size: Range:	short integer 1 to 4 characters -128 to 127	-101 5 127 -128
INT	Description: Size: Range:	integer 1 to 6 characters -32768 to 32767	-25724 232 -6 248
DINT	Description: Size: Range:	double integer 1 to 11 characters -2147483648 to 2147483647	-1357679042 257 6200471
USINT	Description: Size: Range:	unsigned short integer 1 to 3 characters 0 to 255	123 35 6 255
UINT	Description: Size: Range:	unsigned integer 1 to 5 characters 0 to 65535	65535 7 333 597
UDINT	Description: Size: Range:	unsigned double integer 1 to 10 characters 0-4294967295	4294967295 256 334510

Communicating with Modbus and Profibus Devices

Using This Chapter

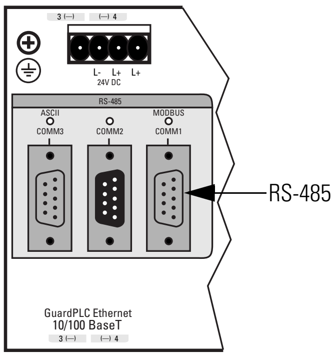
Modbus RTU Slave

Modbus is only available on GuardPLC 1600 or 1800 controllers. You can connect a Modbus master to the controller's COMM1 port. This Modbus connection is two-way non-safety-related communication between the controller (slave) and the master device. You cannot program the controller using this port.

The controller is a Modbus RTU slave device and only responds to reads and writes from the master.

To use the Modbus function, signals that you wish to send out/receive into the COMM1 port must be connected to placeholders in the Modbus-protocol Connect Signals window.

Connecting the Controller to a Modbus Device



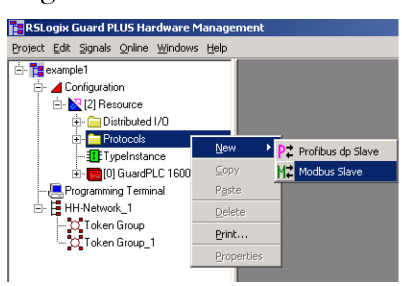
Connection	Signal	Function
1	---	---
2	RP	5V, decoupled with diodes
3	RxD/TxD-A	Receive/Transmit data A
4	CNTR-A	Control Signal A
5	DGND	Data reference potential
6	VP	5V, positive pole of supply voltage
7	---	---
8	RxD/TxD-B	Receive/Transmit data B
9	CNTR-B	Control Signal B

IMPORTANT The Modbus port is RS-485. You must use an electrical interface device to connect the controller to an RS-232 device.

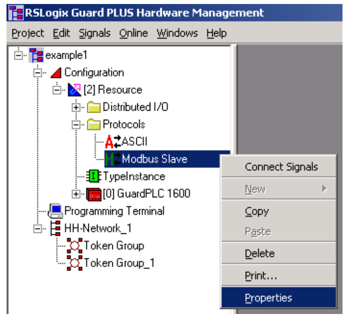
Configuring the Modbus Serial Port

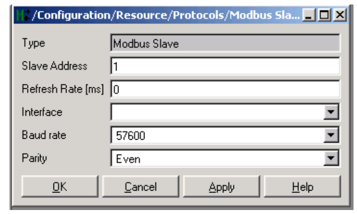
You must either create a new project or open an existing project before you can configure Modbus communications. Once the software opens a project, it automatically displays the Hardware Management window, from which you configure the Modbus port.

1. Right-click on *Protocols* and select *New →Modbus Slave*.



2. Expand *Protocols*. Right-click on the *Modbus Slave* icon and select *Properties*.





For this field:	Specify:
Slave Address	the slave address (1 to 247) of the controller. The Modbus protocol of the controller supports only a direct point-to-point connection between the master and slave. The controller is always configured as slave. It only transfers process values via the serial interface to the master when it receives the corresponding request from the master.
Interface	select the field bus interface to be used by the Modbus Slave protocol (comm1, comm2, comm3). Select *comm1* for GuardPLC 1600 or 1800 controllers.
Refresh Rate	Refresh rate in ms for non-safe communications. The default is 0, the fastest refresh rate.
Baud Rate	the data transfer speed in bits/s. Select from a dropdown menu of predefined values between 300 and 115,200 bps. The default baud rate is 9600 bps.
Parity	the parity for the recognition of transfer errors. Select *No, Odd,* or *Even*. The default is No parity.
Stop Bit	either 1 or 2 stop bits for the serial data transfer. The default is 1 stop bit.

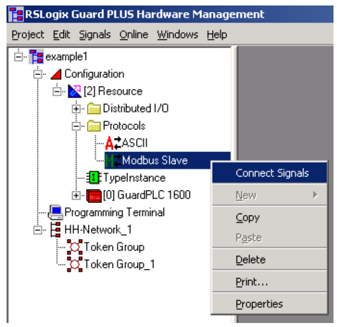
Connecting Signals

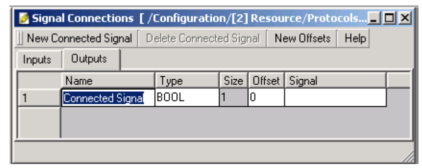
The Modbus RTU Slave protocol allows you to read data from the GuardPLC and write data to the GuardPLC, but none of this data can be used for safety functions.

“Inputs” are signals sent from the Modbus master to the controller (slave). “Outputs” are signals sent from the controller (slave) to the master.

To connect signals:

1. Expand *Protocols*. Right-click on the *Modbus Slave* icon and select *Connect Signals*.





If you want to:	Select this tab:
create a new signal	New Connected Signal
renumber offsets sequentially for all signals	New Offsets
delete the selected signal	Delete Connected Signal

2. Edit the signals you want to receive or send.
   - Use the *Inputs* tab to determine which values to read into the controller.
   - Use the *Outputs* tab to define output values to send to the Modbus master. Signals in the output tab must match the order of signal types requested by the Modbus master.
   - Associate each input or output with a signal from the signal editor. You can drag and drop signals from the signal editor to the signal connections dialog.

The Modbus function calls must match the order in which the signal offsets appear. For example, if you want to read 3 Boolean signals followed by 4 Registers, the first 3 signals must be “BOOL” and the next 4 must be “INT” signals.

TIP

The offset in the Modbus output section is numbered based on bytes. When you request these signals, the first signal is always 0, the second signal is always 1, the third signal is always 2, etc.

The output section automatically sorts the name field based on alphanumerical order. This does not automatically change the offsets, but if you renumber after sorting, the offsets will change and there is no undo feature. This changes the order in which the signals are sent out the serial port.

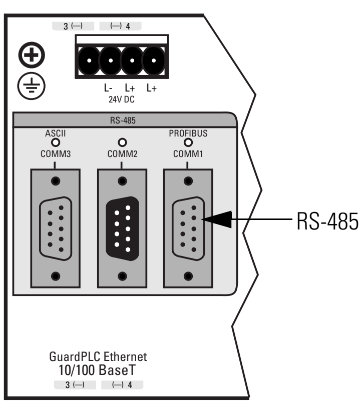
Since names are only used in printouts, you may want to enter these names in alphanumeric order to begin with. (For example signal 101, signal 102, signal 103, signal 104, etc.)

Profibus DP Slave

Profibus DP Slave protocol is only available via the GuardPLC 1600 and 1800 controller's COMM1 port. This connection is two-way non-safety-related communication from the controller (slave) to the master device. You cannot program the controller using this port.

To use the Profibus DP function, signals that you wish to send out the COMM1 port must be connected to placeholders in the Profibus DP-protocol Connect Signals window.

Connecting the Controller to a Profibus DP Device



Connection	Signal	Function
1	---	---
2	RP	5V, decoupled with diodes
3	RxD/TxD-A	Receive/Transmit data A
4	CNTR-A	Control Signal A
5	DGND	Data reference potential
6	VP	5V, positive pole of supply voltage
7	---	---
8	RxD/TxD-B	Receive/Transmit data B
9	CNTR-B	Control Signal B

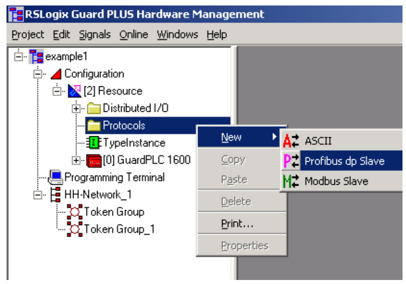
IMPORTANT

The Profibus port is RS-485. You must use an electrical interface device to connect the controller to an RS-232 device.

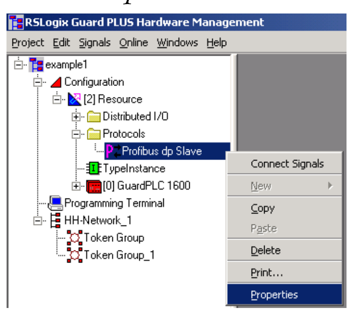
Configuring the Profibus DP Serial Port

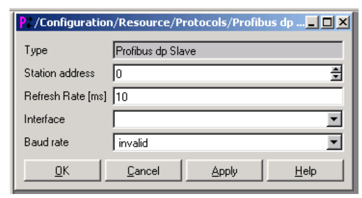
You must either create a new project or open an existing project before you can configure Profibus DP communications. Once the software opens a project, it automatically displays the Hardware Management window, from which you configure the Profibus port.

1. Right-click on *Protocols* and select *New →Profibus dp Slave*.



2. Expand *Protocols*. Right-click on the *Profibus dp Slave* icon and select *Properties*.





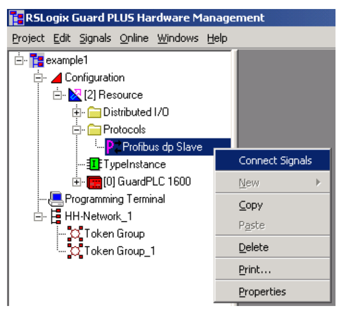
For this field:	Specify:
Station Address	the address which uniquely identifies the Profibus dp slave on the network. The station address must be less than or equal to 126.
Refresh Rate	Refresh rate in ms for non-safe communications. The default is 0, the fastest refresh rate.
Interface	select the field bus interface to be used by the Profibus dp Slave protocol (comm1, comm2, comm3). Select *comm1* for GuardPLC 1600 or 1800 controllers.
Baud Rate	the data transfer speed in bits/s. Select from a dropdown menu of predefined values between 300 and 115,200 bps. The default baud rate is 9600 bps.

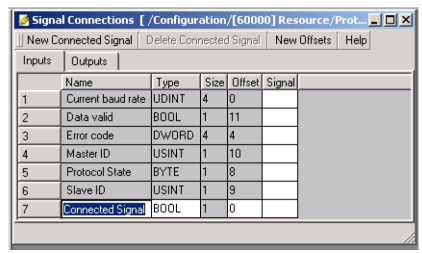
Connecting Signals

The Profibus DP Slave protocol allows you to read data from the GuardPLC and write data to the GuardPLC, but none of this data can be used for safety functions.

"Inputs" are signals sent from the Profibus master to the controller (slave). "Outputs" are signals sent from the controller (slave) to the master.

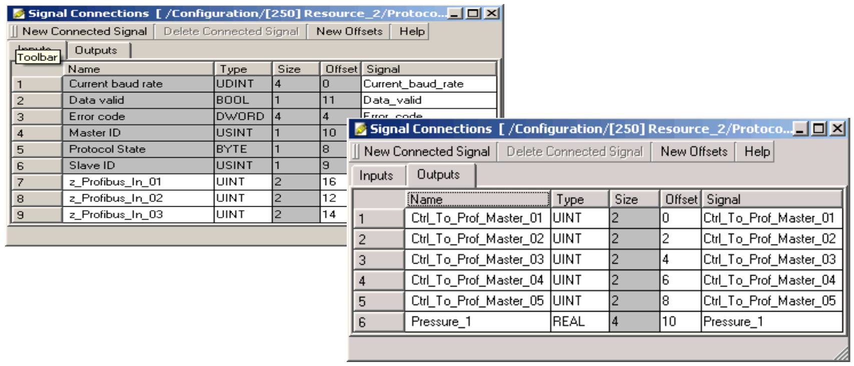
1. Expand *Protocols*. Right-click on the *Profibus-dp Slave* icon and select *Connect Signals*.





If you want to:	Select this tab:
create a new signal	New Connected Signal
renumber offsets sequentially for all signals	New Offsets
delete the selected signal	Delete Connected Signal

2. Edit the signals you want to receive or send:
   - Use the *Inputs* tab to determine which values to read into the controller. The *Inputs* tab contains pre-defined system variables which can be interrogated via the assignment of signals.
   - Use the *Outputs* tab to define output values to send to the Profibus master.
   - Associate each input or output with a signal from the signal editor. You can drag and drop signals from the signal editor to the signal connections dialog.



- Click on *New Offsets* to automatically calculate the offsets for the new signals.

IMPORTANT	Due to the offsets of the system variables, the offset of the first input signal must begin with 12. The offset for the first output signal begins with 0. The Profibus ID for the first input signal is 0.

Configuring the Profibus Master

For both the Profibus output and input signals, the Profibus ID of the first signal to communicate, the number of signals, and the number of bytes must be configured in the Profibus Master.

Configuration is accomplished via parameter data read from a GSD file. The parameter data consist of 32 bytes in hexadecimal format which may be displayed in different ways depending upon the Profibus DP master software.

For more information on using Profibus protocol, consult the online Help.

Pulse Testing

Line control is a short-circuit and line break monitoring system (e.g., E-Stop inputs) which utilizes pulse testing. Line control can be configured for the following GuardPLC controllers and distributed I/O:

- GuardPLC 1600
- GuardPLC 2000/1755-IB24XOB16
- 1753-IB16
- 1753-IB20XOB8

Line control cannot be configured on the GuardPLC 1200, GuardPLC 1800, and the 1753-OB16 output-only module. The GuardPLC 1800 is excluded because it features digital inputs that are actually analog inputs with 1-bit resolution.

TIP

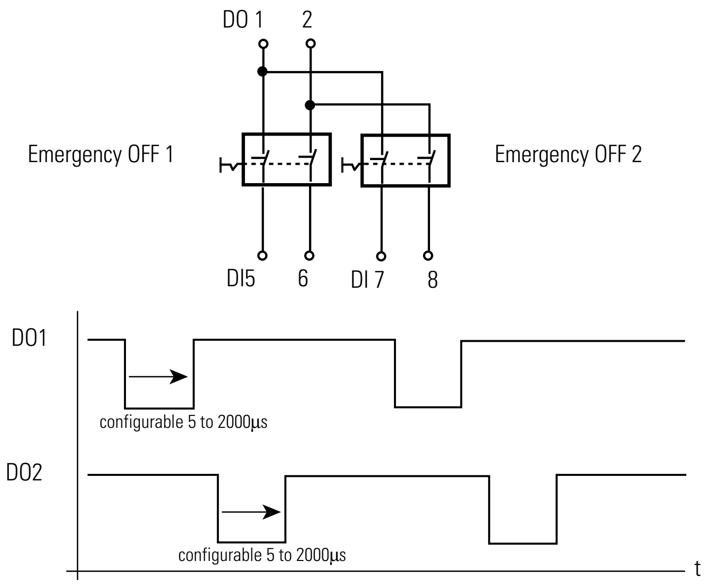
Pulse testing can still be performed on the GuardPLC 1200 and 1800, but software certified function blocks are required. These blocks perform exactly the same function in software as line control does via software configuration.

See the *Certified Function Block Safety Reference Manual*, publication number 1753-RM001, for more information on the Single Pulse Test Output (SPTO) and Redundant Pulst Test Output (RPTO) certified function blocks.

Wiring for Line Control

GuardPLC 1600 and 1753-IB20XOB8

Up to 8 digital outputs (DO1 to DO8) can be configured as pulsed outputs. The example below shows 2 pulse outputs connected to the digital inputs (DI) of the same system. As a result, the connections to the digital inputs (DI) are monitored.



The digital outputs DO1 and DO2 are pulsed (briefly set to low) so that the connections to the digital inputs are monitored. The duration of the test can be configured in the range of 5 to 2000 µs with a default value of 400 µs.

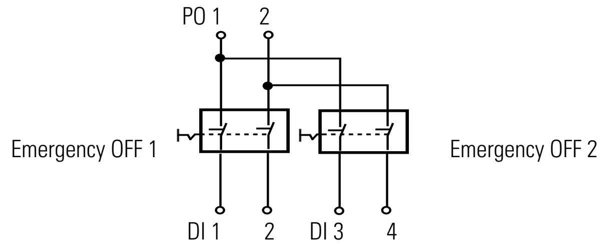
1753-IB16

For Line Control, the 1753-IB16 has four digital pulse test sources (PO) connected to the following terminals:

Terminal Number	Designation	Function
25	L-	Reference pole
26	1	Pulsed source 1
27	2	Pulsed source 2
28	3	Pulsed source 3
29	4	Pulsed source 4
30	L-	Reference pole

The example below shows 2 pulse test sources connected to the digital inputs (DI) of the same system. As a result, the connections to the digital inputs (DI) are monitored.

.



Response to Faults

When the following occurs, inputs are set to 0, a fault code is generated, and the FAULT LED is on.

- short-circuit between two parallel connections
- reversal of two connections
- earth fault on one of the lines (only with earthed reference pole)
- line break or opening of the contacts (e.g., when one of the E-stop off switches is pressed in the example above), the FAULT LED is on and the fault code is generated.

TIP	If multiple errors exist at the same time, the error code is the sum of the individual error codes.

Configuration for Line Control

Required Signals

Set up the following signals using the *Outputs* tab of the digital inputs Signal Connections window in RSLogix Guard PLUS:

Name	Description	Type	Initial Value	Notes
Number of Pulse Channels	Number of pulse outputs being used	USINT	1 to 8	1 to 4 for 1753-IB16 1 to 8 for GuardPLC 1600/2000 1 to 8 for 1753-IB20XOB8
Pulse Slot	Slot occupied by the module with the pulsed outputs	UDINT	1 or 2	2 for GuardPLC 1600 2 for 1753-IB20XOB8 1 for 1753-IB16 1 to 6 for GuardPLC 2000 (wherever 1755-IB24XOB16 is located)
Pulse Delay	Pulse delay is both the low pulse width and pulse test duration.	UINT	400 (default)	Values in µs from 5 to 2000.
Error Code	Error code for each switch	BYTE	N/A	See page B-11 for error code descriptions.
Value	Value for each switch	BOOL		
DI[xx].PulseChannel	Indicates which pulse output is sourcing the input channel	USINT	1 to 8	1 to 4 for 1753-IB16 1 to 8 for GuardPLC 1600/2000 1 to 8 for 1753-IB20XOB8
DO[xx].Value	Initialization value for the pulse outputs	BOOL	TRUE	Each pulse output must be activated.

Pulse Outputs

The pulse outputs must begin at DO[01] and must be sequential. For example, if two pulse outputs are required, they must be DO[01] and DO[02].

Inputs

IMPORTANT To prevent crosstalk, two directly adjacent inputs must not be sourced from the same pulse output.

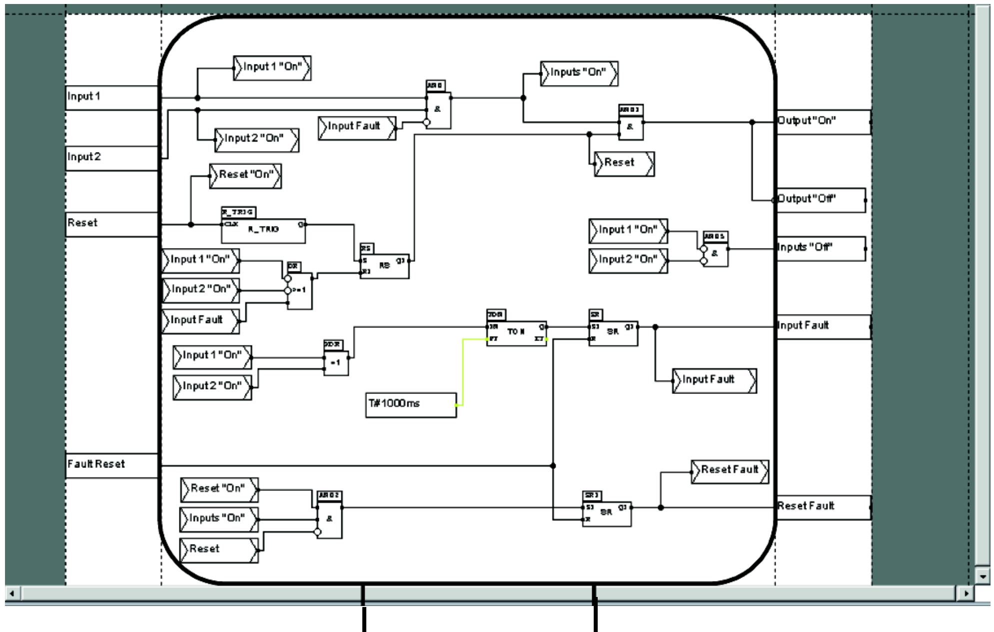
Creating User-Defined Function Blocks

Using This Chapter

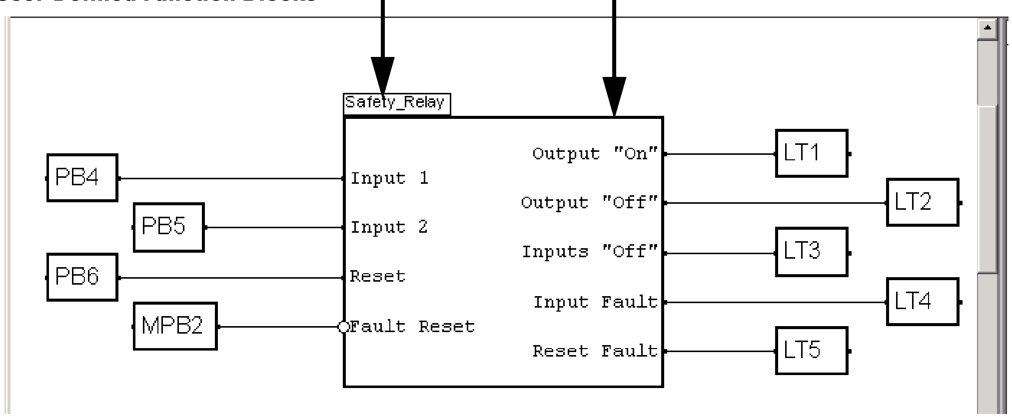
Creating User-Defined Function Blocks

With RSLogix Guard PLUS software, you can create user-defined function blocks that consist of standard function block logic, as shown in the illustrations below.

Existing Function Block

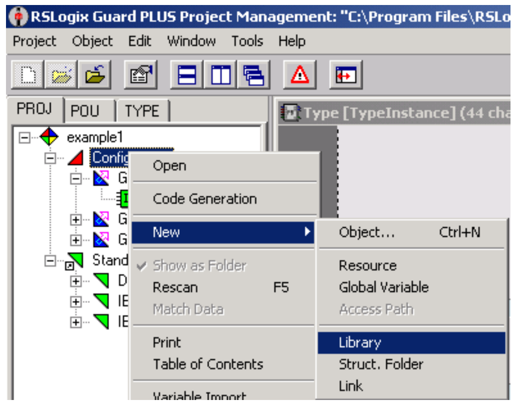


User-Defined Function Blocks

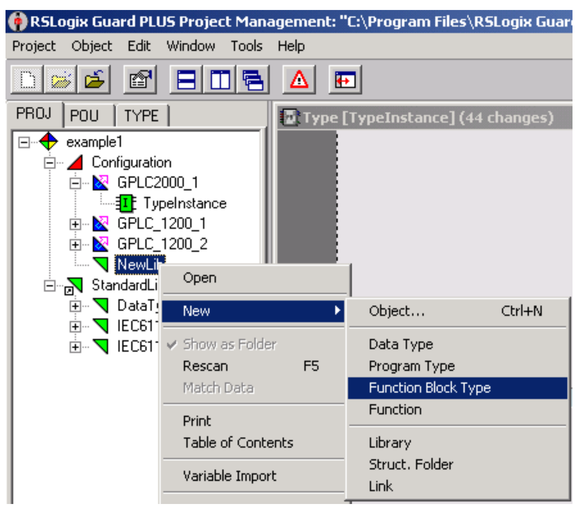


To create a function block:

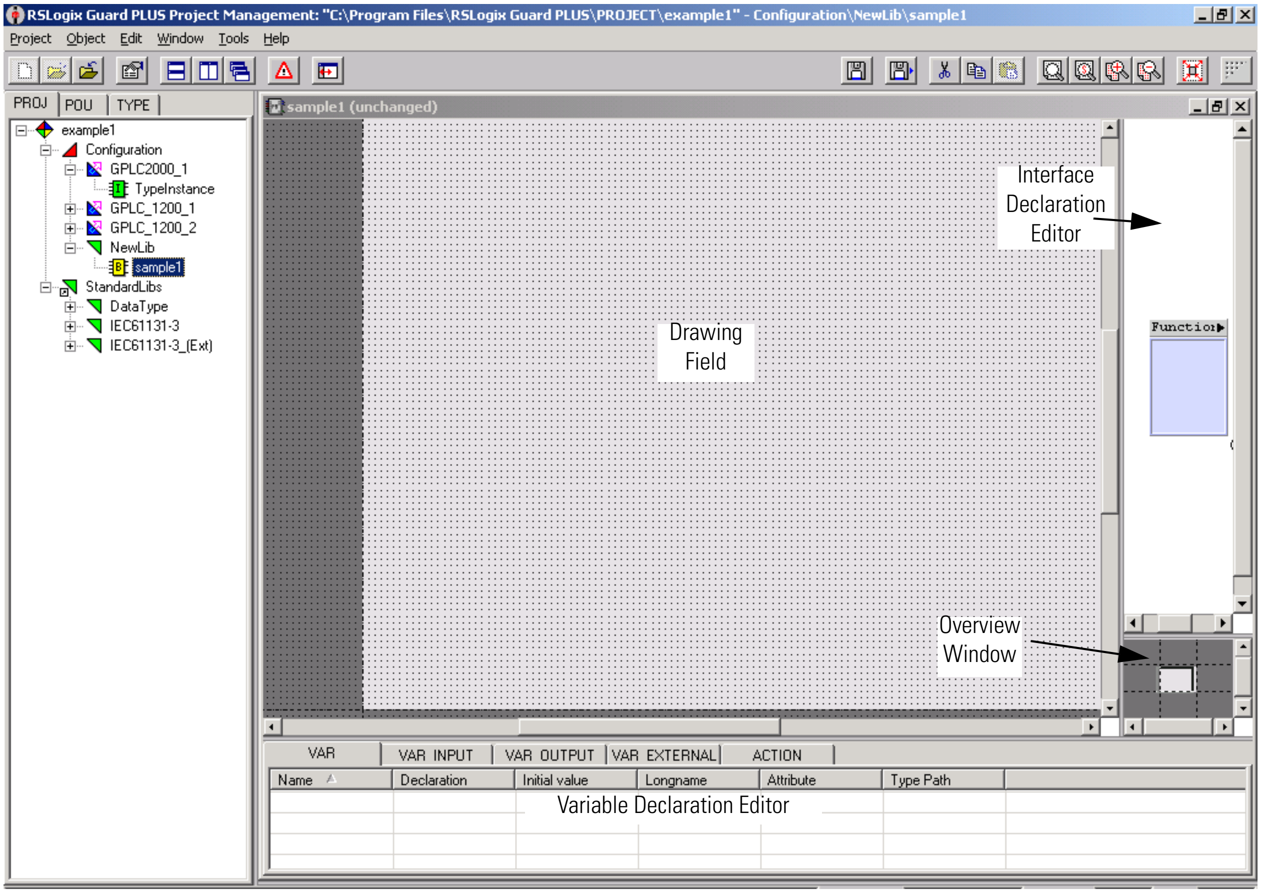
1. In the Program Management Window, right-click on *Configuration* and select *New* →*Library*.



2. Right-click on the new Library and select *New* →*Function Block Type* to create the new function block. You can rename the new function block by right-clicking on it and selecting *Rename*.



3. Double-click the new function block to start the editor.



The FBD editor for user-defined function blocks differs slightly from the FBD editor for routines. The components of the editor are:

Use this component:	To:
Overview Window	displays the function block diagram in reduced scale.
Drawing field	create the logic of the FB-type.
Variable declaration editor (only in FB type editor)	create and define internal variables of a block and initialize them for further use.
Interface declaration editor (only in FB type editor)	define the graphical appearance of a block. The appearance of the block will match the appearance of the user-defined function block in the FBD Editor.

IMPORTANT You cannot place an instance of a user-defined function block within itself.

Declaring variables

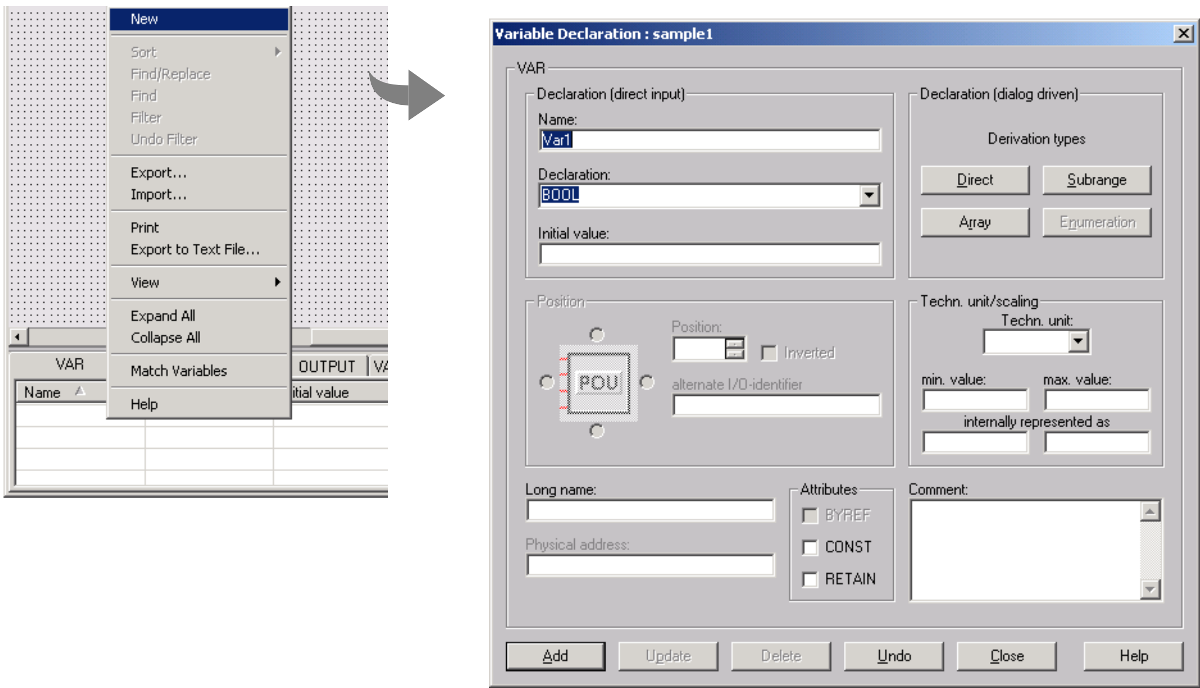
Variable declaration defines the connecting points of the function block. There are tabs for these types of variables.

Use this tab:	To define:
VAR	an internal variable without type limitations. You can also define the attribute: CONST a constant value that cannot be changed by logic
VAR_INPUT	an input variable, which is also displayed on the block.
VAR_OUTPUT	an output variable, which is also displayed on the block.
VAR_EXTERNAL	a global variable that can also be used and edited within function blocks or functions. Value changes are also visible to the outside.
ACTION	an action block. Action blocks describe what action should be performed and which behavior should trigger it.

Valid data types for variables are: BOOL, BYTE, DATE, DATE_AND_TIME, DINT, DWORD, INT, LREAL, REAL, SINT, STRING, TIME, TIME_OF_DAY, UDINT, UINT, USINT, WORD. The default type is BOOL.

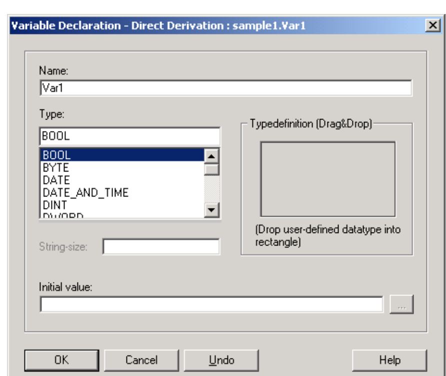
The controller handles REAL values as float values and LREAL values as double values.

To declare a variable, select the tab for the type of variable from the user-defined FBD editor. Right-click in any blank area and select *New Variable*.

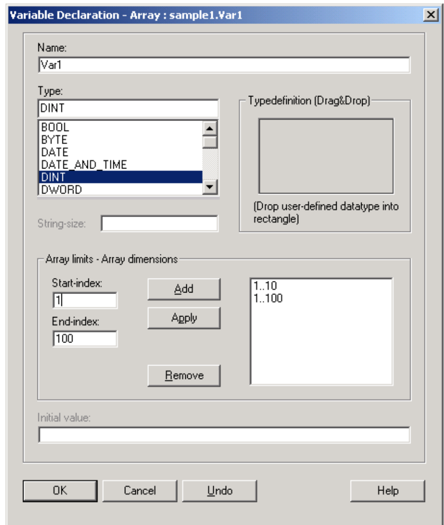


You can either define the variable here or use the Derivation Type buttons (recommended). The Derivation Type buttons activate dialogs to help declare a variable of the selected type. Use these buttons to ensure accurate syntax.

This button:	Defines a variable type:
Direct	derived directly from another variable type



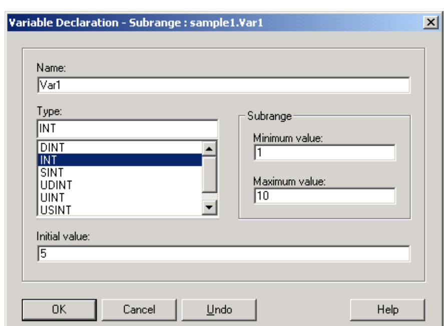
<table>
<tr><th>This button:</th><th>Defines a variable type:</th></tr>
<tr><td>Array</td><td>array of one or more dimensions



For example:

ARRAY:array[1..10] of INT
one-dimensional array of INT values
ARRAY[7] accesses the 7th element

ARRAY:array[1..10,1..10] of REAL
two-dimensional array of REAL values
ARRAY[3,5] accesses the 3rd element of the 5th row</td></tr>
<tr><td>Subrange</td><td>has values that should be within a certain range



For example:

DINT(0,200) is an DINT value where
the minimum allowable value is 0 and
the maximum allowable value is 200</td></tr>
<tr><td>Enumeration</td><td>not yet implemented</td></tr>
</table>

Defining Technical Units and Scaling

You can define technical units and scaling for each variable:

In this field:	Define:
Techn. unit	an available unit from the pulldown menu
min. value max. value internally represented as	reference points to convert a technical unit into an internal value For example: • technical unit from 0 to 24V • internal representation from 0 to 1000 Enter floating point numbers for the scaling.

The available technical units are:

Abbreviation:	Unit:	Definition:
A	Ampere	electrical current
Bq	Bequerel	activity of a radioactive source, disintegration rate
C	Colomb	electrical charge
cd	Candela	light intensity
F	Farad	capacitance
Gy	Gray	absorbed dose
H	Henry	inductance
Hz	Hertz	frequency
J	Joule	energy
K	Kelvin	temperature (in Kelvin)
kg	Kilogram	mass
lm	Lumen	illumination
lx	Lux	illumination density
m	Meter	length
mol	Mol	amount of substance
N	Newton	force
Ohm	Ohm	electrical resistance
Pa	Pascal	pressure
Rad	Radiant	plane angle
s	Siemens	electrical conductance
S	Second	time
sr	sRadiant	solid angle
T	Tesla	magnetic flux density
V	Volt	electrical potential
W	Watt	power
Wb	Weber	magnetic flux

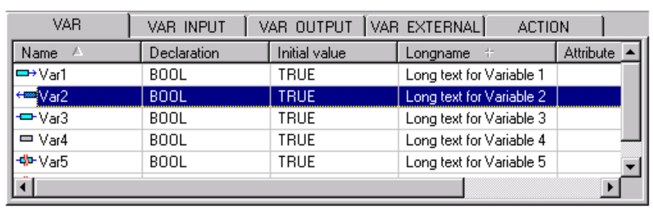
Defining I/O Positions

For input and output variables, you need to define the variables' positions on the function block. The position portion of the variable declaration display is only available for input and output variables. You can define:

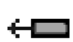
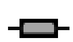
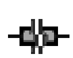
In this field:	Define:
Connection	the side of the block (left, top, right, or bottom) to which the input or output should be connected
Position	the position of the input or output within the block
Inverted	whether to invert I/O of data type BOOL You can only invert BOOL data. Inversions are indicated by a circle around the I/O.
Alternate I/O identifier	an I/O-name This name appears in the block, rather than the generated name.

How the Variables Display

Once you declare your variable, the editor displays the variables.



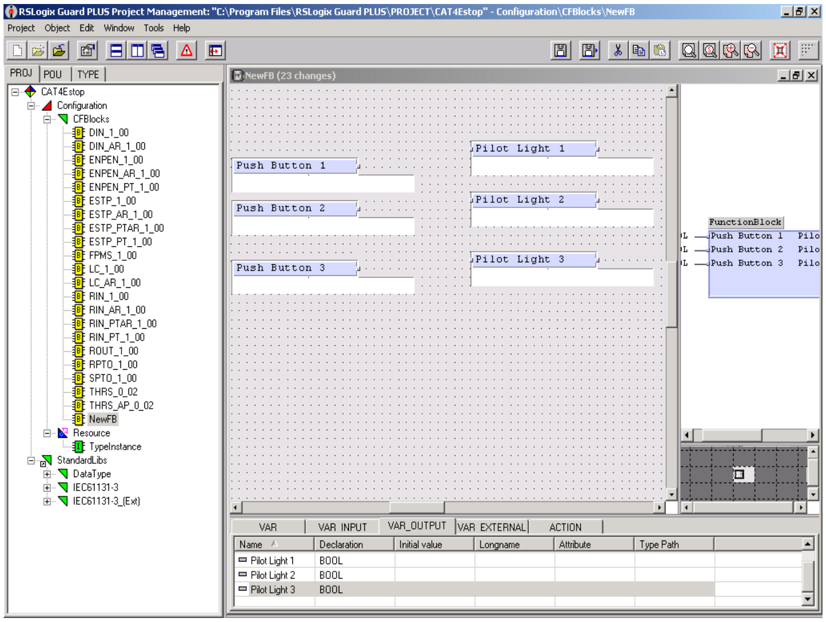
The editor uses these symbols to identify the variables:

Symbol:	Definition:
	used as source The variable is read from.
	used as sink The variable is written to.
	used as source and sink The variable is read from and written to. Variables used in different types of connections are also identified by this symbol.
	not used as source or sink; but the variable is set in the function block diagram.
	not used The variable is declared but not set in the function block diagram.

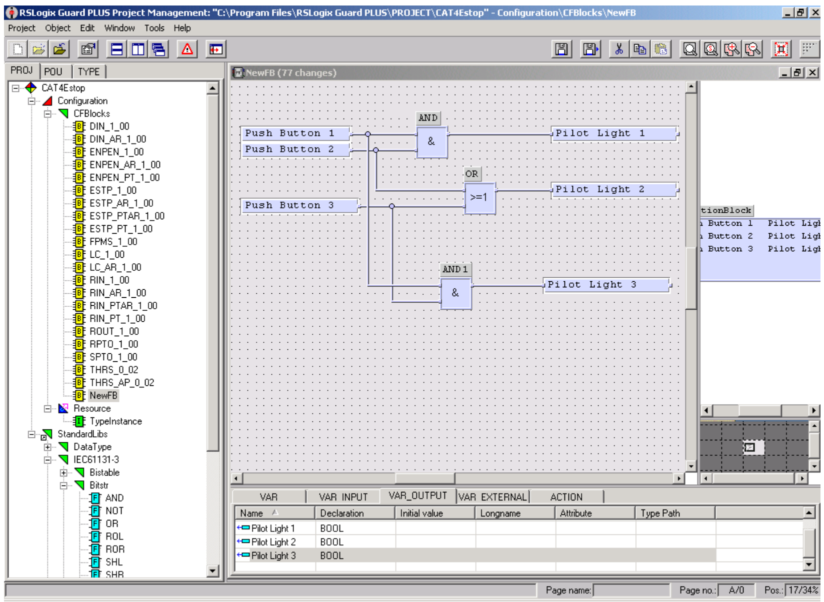
Moving Declared Variables to the User-Defined Function Block Page

In order to use these declared variables, you must:

1. Drag them from the Variable Definition Editor to the user-defined function block page.



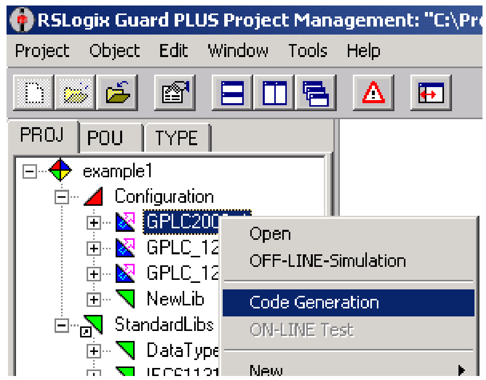
2. Drag down the required function blocks and make all the necessary connections.



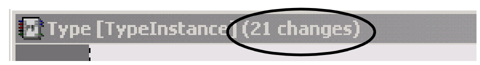
To use the completed user-defined function blocks, you must drag them to the function block page and connect signals to them.

Generating Function Block Code

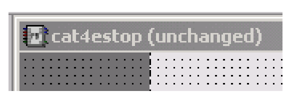
When you complete your function block logic, you must compile that logic into code the controller can execute. In the Program Management Window, expand the project. Right-click the Resource and select *Code Generation*.



IMPORTANT	You should save before every Code Generate. A save is required for any change to the function block page. Any time a change is made, the number of changes displayed on the function block menu bar increments.



When the save is complete, the menu bar displays "unchanged".



The software compiles your function block logic and generates the files that you download to the controller.

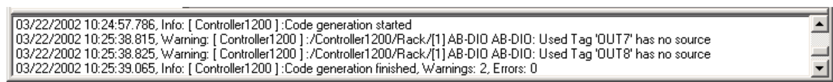
IMPORTANT	When the code generator compiles logic, it also takes into consideration the settings you specified in the Hardware Management Window. If you change these settings and want the changes to take effect, you must compile and download the project again.

Checking for Errors and Warnings

After performing a code generate, check the status bar at the bottom of the Project Management window. The status bar indicates whether or not a .L2P file was successfully generated.

If a .L2P file was not created, check the Hardware Management Window to view the errors and/or warnings compiled during the process of code generation.

The window below shows an example of code generation warnings.



Using High-Speed Counters

This chapter covers using counters in the following systems:

- GuardPLC 1200,
- GuardPLC 1800, or
- GuardPLC 2000 using a 1755-HSC module.

Using This Chapter

Counter/Decoder Modes

The counters can be used in the following operating modes:

- counter mode
- decoder mode

The two counters can be used in different modes at the same time.

Counter Mode

Counter Mode is used for counting pulses at speeds up to 1MHz on the GuardPLC 2000 and 100kHz on the GuardPLC 1200 and 1800.

Tips for Using Counters in a GuardPLC System

1. The 5V signal must be between 4.5V and 5.5V, while the 24V signal must be between 13V and 26.4V.
2. The steepness of the falling edge must be at least 1V per microsecond.
3. The low and high signal times must be at least 5 µs for the GuardPLC 1200 (duty cycle 50% at 100 kHz) and 0.5 µs for the GuardPLC 2000 (duty cycle 50% at 1 MHz).
4. Shield the cable against noise.

Counter Mode Inputs

Pins	Functions
A1, A2	counting input for pulses (high-signals) with falling edge of the pulses
B1, B2	counting direction input, incrementing the counter with low-signal, decrementing the counter with high-signal
Z1, Z2	resets inputs Resets can be made with a short high-signal. A continuous high-signal blocks the counter. Resets can also be made by the controller program.
C1, C2	has no function (GuardPLC 2000 - 1755-HSC only)
C-	GuardPLC 2000 common reference pole, all pins have electrical continuity
L-	GuardPLC 1800 common reference pole, all pins have electrical continuity
I-	GuardPLC 1200 common reference pole, all pins have electrical continuity

Decoder Mode

Decoder Mode is used for safety supervising the inputs by Gray code, but in the application, the bit structure is handled as a normal binary code value. To use this value, it must be converted in the application. The counter inputs can be connected to an incremental encoder with 4-bit binary code to recognize rotation and the direction of rotation.

Decoder Mode Inputs

Pins	Functions
A1, A2	bit 1 (LSB)
B1, B2	bit 2
Z1, Z2	bit 3
C1, C2	bit 4 (GuardPLC 2000 only)

Understanding Counter Module Configuration

Counter Mode/Manual Direction

The simplest mode of operation is pulse counting with Manual Direction. It can be used, for example, in connection with a light barrier where counting events are to be recorded. The direction of counting is determined by the routine.

The count begins at 0 and is incremented or decremented by 1 at each negative transition of the counting pulse. The resolution of the counter is 24 bits. This results in a value range from 0 to 16,777,215.

The counting pulse must be bounce free and must not exceed the maximum frequency of 1 MHz for a GuardPLC 2000 controller or 100 kHz for a GuardPLC 1200 or 1800 controller. The counter input can be set to a voltage of 5V or 24V via the software.

To ensure that the counter functions correctly, the following parameters have to be configured:

Parameter:	Setting:
Cnt[0x].5/24V Mode	true for 24V **or** false for 5V You must configure this parameter with a constant.
Cnt[0x].Auto Advance Sense	(optional according to routine) false to count only up or only down based upon the direction bit
Cnt[0x].Direction	(optional according to routine) true to decrement (counts from 16,777,215 downward) **or** false to increment
Cnt[0x].Gray Code	(optional according to routine) false
Cnt[0x].Reset	(optional according to routine) true If this parameter is set to false, the counter value is reset to 0.

Counter Mode/Direction and Reset

In pulse counting with Direction and Reset, the state of input B is evaluated in addition to counter input A.

When the B input has a low signal while the counter recognizes a negative pulse edge at its A input, the value of the counter is incremented by 1. When there is a high signal at the B input, the counter is decremented by 1.

The counter is released or reset via the Z input. The counter is released when there is a constant LOW signal at the Z input. A constant HIGH signal halts the counter and a short-time HIGH signal resets the counter value to 0.

To enable the counter to function correctly, the following parameters have to be configured in the routine:

Parameter:	**Setting:**
Cnt[0x].5/24V Mode	true for 24V **or** false for 5V The adjusted level also applies to inputs B and Z. You must configure this parameter with a constant.
Cnt[0x].Auto Advance Sense	true to count up and down simultaneously
Cnt[0x].Direction	true to decrement (counts from 16,777,215 downward) **or** false to increment (standard setting)
Cnt[0x].Gray Code	false
Cnt[0x].Reset	true If this parameter is set to false, the counter value is reset to 0.

Decoder Mode/Gray Codes

The gray code is a binary code where the code only differs by one bit with two neighboring numbers. Gray codes are useful in mechanical encoders, since a slight change in location only affects one bit. The controller uses a Gray code (4 bits for a GuardPLC 2000 controller or 3 bits for GuardPLC 1200 and 1800 controllers) which has the following structure:

Step:	Gray Code (GuardPLC 2000):	Gray Code (GuardPLC 1200, 1600, and 1800):	Cnt[0x].Value:
0	0000	000	0
1	0001	001	1
2	0011	011	3
3	0010	010	2
4	0110	110	6
5	0111	111	7
6	0101	101	5
7	0100	100	4
8	1100		12
9	1101		13
10	1111		15
11	1110		14
12	1010		10
13	1011		11
14	1001		9
15	1000		8

Each counter input is fed to three internal counters. When a count is accomplished, the values of the three internal counters are compared, and if the three values differ by more than one bit, the measuring result is rejected and *Cnt[0x].State* indicates an error.

If the measuring result is valid, the system variable *Cnt[0x].Value* contains the associated value (see the above table).

To enable the Gray code decoder to work correctly, the following parameters have to be configured in the routine:

Parameter:	Setting:
Cnt[0x].5/24V Mode	true for 24V **or** false for 5V The adjusted level also applies to inputs B and Z. You must configure this parameter with a constant.
Cnt[0x].Auto Advance Sense	this setting has no function on the gray code (set to false)
Cnt[0x].Direction	this setting has no function on the gray code (set to false)
Cnt[0x].Gray Code	true
Cnt[0x].Reset	this setting has no function on the gray code (set to true)

Configuring the GuardPLC OPC Server

Using This Chapter

OLE for Process Control (OPC) is a standard interface for exchanging data between different applications. The GuardPLC OPC server provides an Ethernet interface between the GuardPLC system and other systems with OPC interfaces.

This chapter describes the steps required to configure the GuardPLC OPC server to read and write data to an OPC client, in this case, a GuardPLC 1600. RSLogix Guard PLUS is used to create a token group and make an HH network connection between the OPC server and the controller. Signals are connected to the input and output sections of this connection. Signals connected to the output section are sent out of the controller to the OPC server. Signals connected to the input sections are sent from the OPC server to the controller.

To create an XML file for use by the OPC Server, follow the steps below, which are described in detail in the following sections.

1. Select the IP Address for the OPC Server.

2. Add the controller and the OPC Server to an RSLogix Guard PLUS project.

3. Configure the GuardPLC System for OPC Communication.

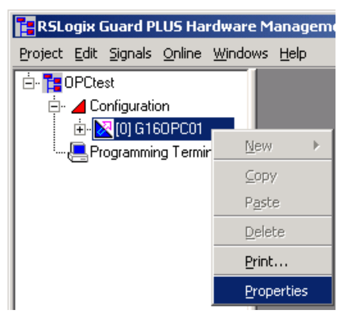
4. Generate XML Code for the OPC Server.

Select an IP Address

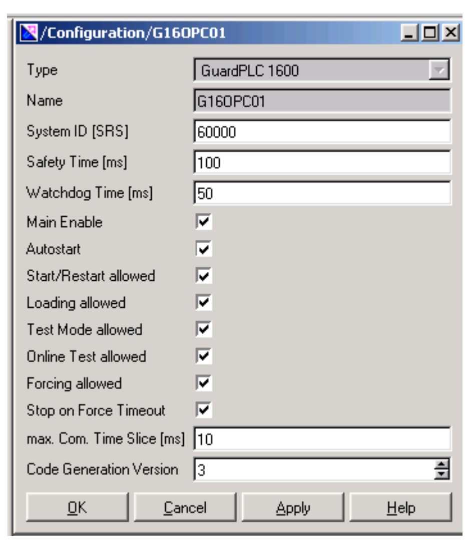
Select an IP address for the OPC Server and the GuardPLC controller. In this example, we used the default IP address of the GuardPLC controller, 192.168.0.99. The IP address of the OPC Server is 192.168.0.216.

Adding the GuardPLC Controller and the OPC Server to the Project

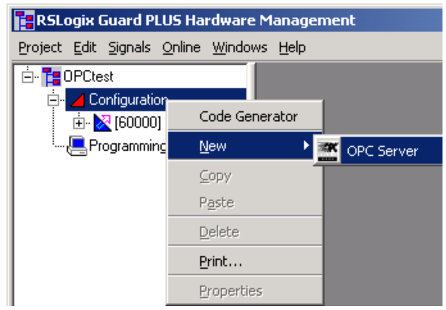
1. In RSLogix Guard PLUS, select *Project* →*New*.
2. Enter the name of the project and confirm *OK*. In this example, our project is called “OPCtest”. The Hardware Management window opens.
3. In the Hardware Management window, expand *Configuration* so that the Resource is visible. In this example, we renamed the resource to G16OPC01. It is not necessary to rename the resource.
4. Right-click on *G16OPC01* and select *Properties*.

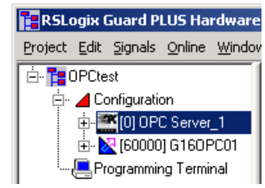


5. Change the *SRS* to 60000 and the *Type* to the controller you are using. This example uses a GuardPLC 1600.
6. Select *Apply* and check the remaining check boxes. Click *OK*.



7. In the project tree, right-click on *Configuration* and select *New* → *OPC-Server*. The OPC server appears in the project tree.





Configuring the GuardPLC System for OPC Communication

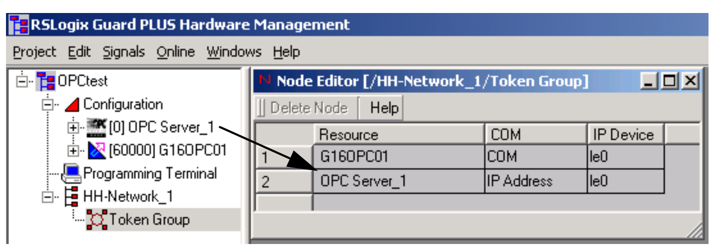
Configure the Communication Network

1. In the Hardware Management window, right-click on the *OPCtest Project* and select *New* → *HH-Network*.

2. Expand the HH-Network.

3. Right-click on *Token Group* and select *Node Editor*.

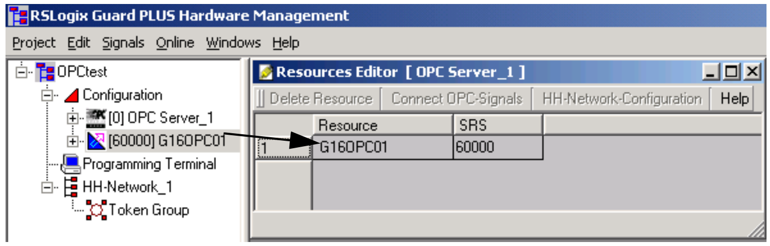
4. Drag the *OPC-Server* and *G16OPC01* onto the Node Editor.



5. Close the Node Editor.

6. Right-click on *OPC-Server* and select *Edit*.

7. Drag *G16OPC01* onto the OPC-Server Resources Editor.

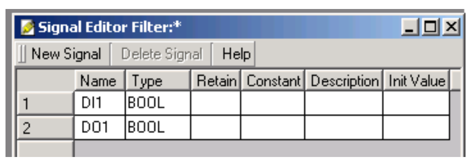


The Specify HH-Network Configuration for OPC dialog opens. Click *OK*.

Connecting Signals

1. Select the *Signals* pulldown menu from the menu bar and choose *Editor*.

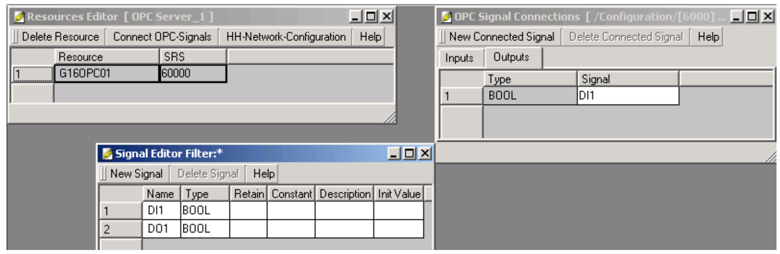
2. Add two new signals:
   - DI1 - input sent to the OPC Server
   - DO1 - output sent from the OPC Server



3. Left-click on *1* in the OPC-Server Resources Editor. Select *Connect OPC-Signals*.

4. Drag *DO1* from the Signal Editor to the Inputs section of the OPC Signal Connections window.

5. Click on the *Outputs* tab. Drag *DI1* from the Signal Editor to the Outputs section of the OPC Signal Connections window.

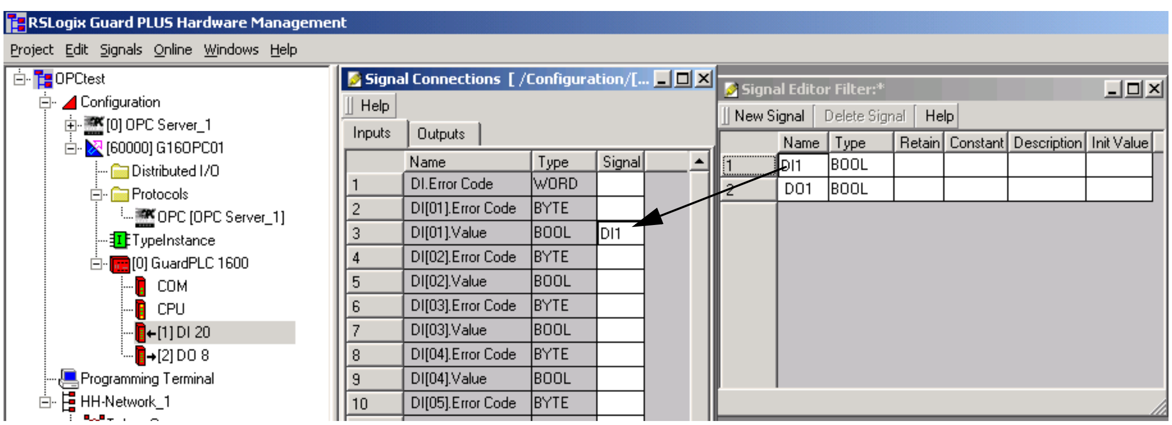


TIP Input and Output are identified from the controller's perspective. Therefore, output means signals sent out from the GuardPLC 1600 and input means signals sent in to the GuardPLC 1600.

6. Close the OPC Signal Connections window and the OPC-Server Resources Editor.

7. Completely expand the G16OPC01 tree. Right-click on *[1] DI 20* and select *Connect Signals*.

8. Drag *DI1* from the Signal Editor to *DI[01].Value* Signal field.

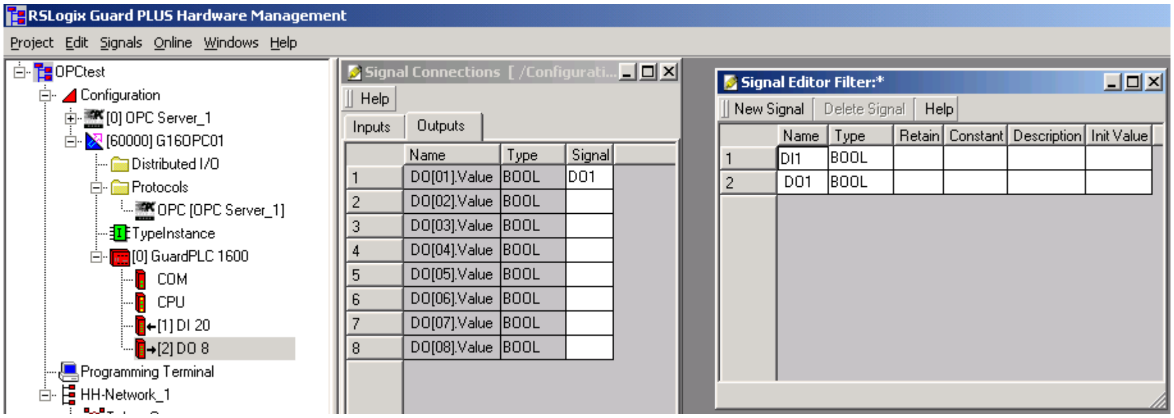


9. Close the DI1 Signal Connections window.

10. Right-click on *[2] DO 8* and select *Connect Signals*.

11. Select the *Outputs* tab.

12. Drag *DO1* from the Signal Editor to the *DO[01].Value* signal field.



13. Close the DO Signal Connections window and the Signal Editor.

Setting the System Properties

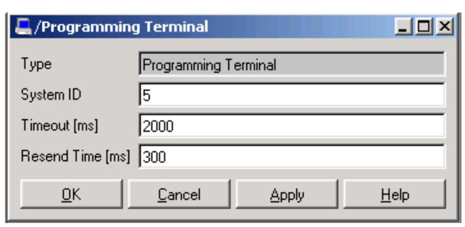
Controller

1. Right-click on *COM* under the GuardPLC 1600 tree and select *Properties*.

2. Set the IP Address to 192.168.0.99 and click OK.

3. Right-click on *Programming Terminal* and select *Properties*.

4. Set the System ID (SRS) to 5. Click *OK*.



TIP

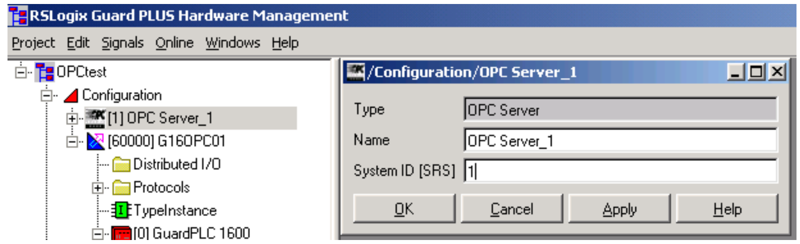
There are 3 devices in our system:

1. the PC running the Signal Editor,
2. the OPC Server, and
3. the GuardPLC controller.

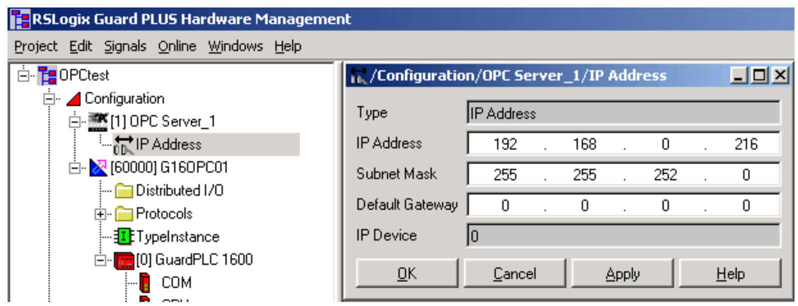
All must have unique SRS numbers.

4. Right-click on *OPC-Server* and select *Properties*.

5. Set the System ID (SRS) to 1 and click *OK*.



6. Expand the OPC-Server and double-click on *IP Address*. Set the IP Address to 192.168.0.216 and click *OK*.

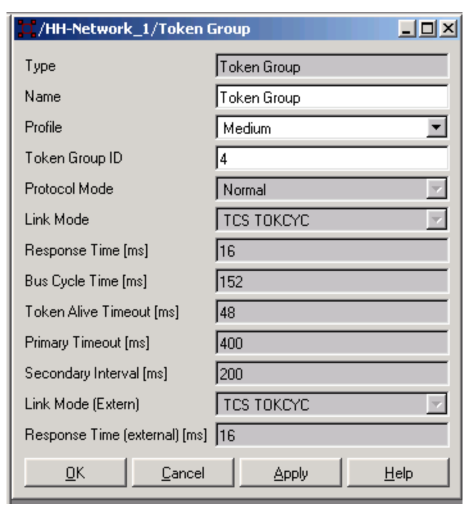


Token Group

1. Right-click on *Token Group* and select *Properties*.

2. Change the profile to "Medium".

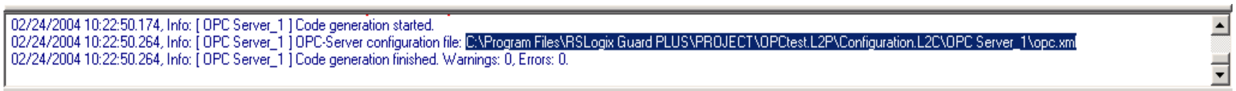
3. Verify that the Token Group ID is 4. Click *Apply* and then *OK*.



Generating Code for the OPC Server

1. To generate the XML file for the OPC server, right-click on the OPC Server and select *Code Generator*. The process should take only 1 or 2 seconds. Make sure there are no warnings or errors.

2. The resulting XML file is located in the project path of the RSLogix Guard PLUS project. Make note of this path so that you can point to it from the GuardPLC OPC Server.



3. Return to the Project Management window and save the project.

4. Right-click on *Configuration* and select *Code Generation*. Check the Error State Viewer to make sure there are no errors or warnings. Correct any errors indicated.

Going Online with the Controller

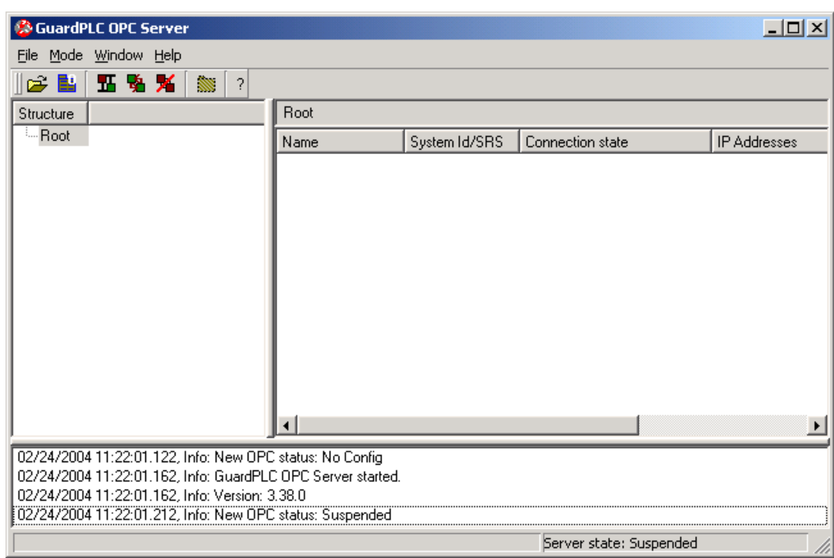
1. Return to the Hardware Management window and download the project to your controller.

2. Put the controller into RUN.

3. Minimize RSLogix Guard PLUS.

Using the OPC Server

1. Start the GuardPLC OPC Server by selecting *Start* → *Programs* → *RSLogix Guard PLUS* → *GuardPLC OPC Server*.

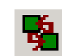


2. Select *File* → *Open*.

3. Point to your opc.xml file and click *Open*.

4. The controller and the OPC Server should appear in the Root window.

Root			
Name	**System ID/SRS**	**Connection state**	**IP Addresses**
G16OPC01	60000.1	Not connected	192.168.0.99, 127.0.0.1
OPC Server_1	1	Not connected	192.168.0.216, 127.0.0.1

5. Click on the *Connect to HIPRO* button. 

6. Answer *Yes* to the query.

7. A connection opens between the G16OPC01 and the OPC Server.

Root			
Name	**System ID/SRS**	**Connection state**	**IP Addresses**
G16OPC01	60000.1	Connected on channel one	192.168.0.99, 127.0.0.1
OPC Server_1	1	Connected on channel one	192.168.0.216, 127.0.0.1

8. In the structure view, expand *Root*.

9. Expand *G16OPC01*.

10. Use the Export and Import icons to view input (DI1) and output (DO1) data, respectively.

Specifications

GuardPLC 1200

Specifications	1754-L28BBB
Controller	
User Memory	500 KB application code memory 500 KB application data memory
Digital Inputs	
No. of Inputs	20 (not electrically isolated from each other, isolated from the backplane)
Nominal Input Voltage	24V dc
On-State Voltage	10V dc to 30V dc
On-State Current	2 mA @ 10V dc, 13 mA @ 30V dc
Off-State Voltage	5V dc (max.)
Off-State Current	1.5 mA max. (1 mA @ 5V)
Digital Outputs	
No. of Outputs	8 (not electrically isolated from each other, isolated from the backplane)
Output Voltage Range	18.4V to 26.8V
Output Current	0.5A per Channel (Channel 1 to 6) 2A per Channel (Channel 7, 8)
Surge Current per Channel	1A for 10 ms @ 1Hz (Channel 1 to 6) 4A for 10 ms @ 1Hz (Channel 7, 8)
Minimum Current Load	2.5 mA per Channel
On-State Voltage Drop	2.0V dc @ 500 mA (max.)
Off-State Leakage Current	1 mA per Channel (max.)
Temporary Overload	Output switches off until overload is eliminated
Counters	
No. of Counters	2
Inputs per Counter	3 (Input A, Input B, Z/Gate/Reset)
Counter Resolution	24 bit
Maximum Input Frequency	100kHz in Counter Modes (Input A)
Trigger	Negative Edge
Edge Steepness	1 V/μs
Duty Cycle	50% @ 100 kHz
Input Voltages	4.5V to 5.5V for 5V Input 13V to 26.4V for 24V Input
Input Current	Typ. 15 mA, ≤ 3 mA

Specifications	1754-L28BBB
Power Supply	
Supply Voltage (L+)	24V dc
Supply voltage range	20.4V dc to 28.8V dc (10 ms buffer), ripple ≤15%
Maximum Power Rating	8A (1A to run the controller, 7A for inputs and outputs)
Environmental Conditions	
Storage Temperature:	-40° C to +85° C without backup battery
Operating Temperature:	0° C to +60° C
Mechanical Dimensions	
Width x Height x Depth:	160 mm x 112 mm x 90 mm (6.3 in. x 4.41 in. x 3.54 in.)
Weight	680 g (1.5 lb.)
Agency Certifications (when product is marked)	C-UL Listed Industrial Control Equipment Marked for all applicable directives N223 Marked for all applicable acts Functional Safety 1oo2D (AK 1 to 6, SIL 1 to 3, according to DIN V 19250 and IEC 61508 respectively) Category 1 to 4, according to EN954-1

GuardPLC 1600

Specifications	1753-L28BBBM and 1753-L28BBBP
Controller	
User Memory	max. 250 KB user program memory max. 250 KB application data memory
Minimum Watchdog	10 ms
Minimum Safety Time	20 ms
Current Consumption	max. 8A (with max. load) 0.5A idle current (just running the controller)
Operating Voltage	24V dc, -15% to +20%, $w_{SS} \leq 15\%$ (from a power supply with protective separation conforming to IEC 61131-2 requirements)
Interfaces: GuardPLC Ethernet	4 x RJ-45, 10/100BaseT (with 100 Mbit/s) with integrated switch
Protection	IP20
Digital Inputs	
No. of Inputs	20 (not electrically isolated)
On-state	Voltage: 15V to 30V dc Current Consumption: ≥ 2 mA @ 15V 7.5 mA @ 30V
Off-state	Voltage: max. 5V dc Current Consumption: max 1.5 mA (1 mA @ 5V)
Switching Point	typically 7.5V
Supply	5 x 20V / 100 mA @ 24V short-circuit proof

Specifications	1753-L28BBBM and 1753-L28BBBP
Digital Outputs	
No. of Outputs	8 (not electrically isolated)
Output Voltage Range	18.4V to 26.8V
Output Current	Channels 1 to 3 and 5 to 7: 0.5 A @ 60°C (140°F) Channels 4 and 8: 1A @ 60°C (140°F); 2A @ 50°C (122°C)
Surge Current per Channel	1A for 10ms @ 1 Hz (Channels 1 to 3 and 5 to 7) 4A for 10ms @ 1 Hz (Channels 4 and 8)
Minimum Current Load	2 mA per Channel
On-State Voltage Drop	max. 2.0V dc @ 2A
Off-State Leakage Current	max. 1 mA @ 2V
Environmental Conditions	
Storage Temperature :	-40°C to +85°C (-40°F to +185°F)
Operating Temperature:	0°C to +60°C (+32°F to 140°F)
Mechanical Dimensions	
Width	257 mm (10.1 in.) including housing screws
Height	114 mm (4.49 in.) including latch
Depth	66 mm (2.60 in.) including grounding bolt
Weight	1.2 kg (2.64 lb.)
Agency Certifications (when product is marked)	C-UL Listed Industrial Control Equipment Marked for all applicable directives N223 Marked for all applicable acts Functional Safety 1oo2D (AK 1 to 6, SIL 1 to 3, according to DIN V 19250 and IEC 61508 respectively) Category 1 to 4, according to EN954-1

GuardPLC 1800

Specifications	1753-L32BBBM-8A and 1753-L32BBBP-8A
Controller	
User Memory	max. 250 KB user program memory max. 250 KB application data memory
Minimum Watchdog Time	10 ms
Minimum Safety Time	20 ms
Current Consumption	max. 9A (with max. load) 0.75A idle current (just running the controller)
Operating Voltage	24V dc, -15% to +20%, $w_{SS} \leq 15\%$ (from a power supply with protective separation conforming to IEC 61131-2 requirements)
Protection	IP 20
Digital Inputs	
No. of Inputs	24 (not electrically isolated)
On State	Voltage: 15V to 30V dc Current Consumption: approximately 3.5 mA @ 24V dc Current Consumption: approximately 4.5 mA @ 30V dc
Off State	Voltage: max. 5V dc Current Consumption: max. 1.5 mA (1 mA @ 5V dc)
Input Resistance	< 7kΩ
Overvoltage Protection	-10V, +35V
Max. line length	300 m (9.8 ft.)
Supply	20V / 100 mA, short-circuit proof
Digital Outputs	
No. of Outputs	8 (not electrically isolated)
Output Voltage Range	≥ L+ minus 2V
Output Current	Channels 1 to 3 and 5 to 7: 0.5 A @ 60°C (140°F) Channels 4 and 8: 1 A @ 60°C (140°F); 2 A @ 50°C (122°C)
Surge Current per Channel	1A for 10 ms @ 1 Hz (Channels 1 to 3 and 5 to 7) 4A for 10 ms @ 1 Hz (Channels 4 and 8)
Minimum Current Load	2 mA per Channel
Internal Voltage Drop	max. 2.0V dc @ 2A
Off-State Leakage Current	max. 1 mA @ 2V
Total Output Current	max. 7A

Specifications	1753-L32BBBM-8A and 1753-L32BBBP-8A
Counters	
Number of Counters	2 (not electrically isolated)
Inputs	3 per counter (A, B, Z)
Input Voltages	5V and 24V dc High signal (5V dc): 4V to 6V High signal (24V dc): 13V to 33V Low signal (5V dc): 0V to 0.5V Low signal (24V dc): -3V to 5V
Input Currents	1.4 mA @ 5V dc 6.5 mA @ 24V dc
Input Impedance	3.7 kΩ
Counter Resolution	24-bit
Max. Input Frequency	100 kHz
Triggered	on negative edge
Edge Steepness	1 V/µs
Pulse Duty Factor	1:1
Analog Inputs	
Number of Inputs	8 (unipolar, not electrically isolated)
External Shunt (for current measurement)	500 Ω for 0 to 20 mA
Input values related to L-	Nominal Value: 0 to +10V dc or 0 to +20 mA with 500 Ω shunt Service Value: -0.1 to +11.5V dc or -0.4 to +23 mA with 500 Ω shunt
Input Impedance	1 MΩ
Internal Resistance of the Signal Source	≤500 Ω
Overvoltage Protection	+15V, -4V
Resolution (A/D Converter)	12-bit
Accuracy	0.1% @ 25°C (77°F) 0.5% @ 60°C (140°F)
Transmitter Supplies	25.37 to 28.24V / ≤46 mA, short-circuit proof
Safety Accuracy	± 2%
Environmental Conditions	
Storage Temperature:	-40° C to +85° C (-40°F to +185°F)
Operating Temperature:	0° C to +60° C (+32°F to 140°F)

Specifications	1753-L32BBBM-8A and 1753-L32BBBP-8A
Mechanical Dimensions	
Width	257 mm (10.1 in.) including housing screws
Height	114 mm (4.49 in.) including latch
Depth	66 mm (2.60 in.) including grounding screw 80 mm (3.15 in.) including shield plate
Weight	1.2 kg (2.64 lb.)
Agency Certifications (when product is marked)	C-UL Listed Industrial Control Equipment Marked for all applicable directives Marked for all applicable acts N223 Functional Safety 1oo2D (AK 1 to 6, SIL 1 to 3, according to DIN V 19250 and IEC 61508 respectively) Category 1 to 4, according to EN954-1

Distributed I/O

1753-IB16 Input Module

Specifications	1753-IB16
General	
Interfaces: GuardPLC Ethernet	2 x RJ-45, 10/100BaseT (with 100 Mbit/s) with integrated switch
Operating Voltage	24V dc, -15% to +20%, w_{ss} 15% from a power supply with protective separation, conforming to IEC 61131-2 requirements
Response Time	≥ 10 ms
Current Consumption	max. 0.8A (with max. load) (0.4A idle current)
Digital Inputs	
No. of Inputs	16 (not electrically isolated)
1 Signal	Voltage: 15V to 30V dc, Current Consumption: ≥ 2 mA @ 15V
0 Signal	Voltage: max. 5V dc, Current Consumption: max 1.5 mA (1 mA @ 5V)
Switching Point	typically 7.5V
Switching Time	typically 250 μs
Sensor Supply	4 x 19.2V / 40 mA @ 24V short-circuit proof

Specifications	1753-IB16
Pulse Test Sources	
Number of Pulse Test Sources	4 (not electrically isolated)
Output Voltage Range	approximately 24V
Output Current	60 mA
Minimum Current Load	none
Response to Overload	4 x ≥ 19.2V, short circuit current 60 mA @ 24V
Environmental Conditions	
Storage Temperature	-40° C to +85° C (-40° F to +185° F)
Operating Temperature	0° C to +60° C (+32° F to +140° F)
Mechanical Dimensions	
Width	152 mm (5.99 in.) including housing screws
Height	114 mm (4.49 in.) including latch
Depth	66 mm (2.60 in.) including grounding bolt
Weight	0.7 kg (1.54 lb.)
Agency Certifications (when product is marked)	C-UL Listed Industrial Control Equipment Marked for all applicable directives Marked for all applicable acts N223 Functional Safety 1oo2D (AK 1 to 6, SIL 1 to 3, according to DIN V 19250 and IEC 61508 respectively) Category 1 to 4, according to EN954-1

1753-IB20XOB8 Combination I/O Module

Specifications	1753-IB20XOB8
General	
Interfaces: GuardPLC Ethernet	2 x RJ-45, 10/100BaseT (with 100 Mbit/s) with integrated switch
Operating Voltage	24V dc, -15% to +20%, w_{SS} 15% from a power supply with protective separation, conforming to IEC 61131-2 requirements
Response Time	≥ 10 ms
Battery Backup	none
Current Consumption	max. 8A (with max. load), idle current 0.4A @24V

Specifications	1753-IB20XOB8
Digital Inputs	
No. of Inputs	20 (not electrically isolated)
1 Signal	Voltage: 15V to 30V dc, Current Consumption: ≥ 2 mA @ 15V
0 Signal	Voltage: max. 5V dc, Current Consumption: max 1.5 mA (1 mA @ 5V)
Switching Point	typically 7.5V
Sensor Supply	5 x 20V / 100 mA @ 24V short-circuit proof
Digital Outputs	
No. of Outputs	8 (not electrically isolated)
Output Voltage Range	≥ L+ minus 2V
Output Current	channels 1 to 3 and 5 to 7: 0.5 A @ 60°C (140°F) channels 4 and 8: 1 A @60°C (140°F), 2 A @ 50°C (122°F)
Surge Current per Channel	1A for 10 ms @ 1 Hz (Channels 1 to 3 and 5 to 7) 4A for 10 ms @ 1 Hz (Channels 4 and 8)
Minimum Current Load	2 mA per channel
Internal Voltage Drop	maximum 2V @ 2A
Leakage Current (with 0 signal)	maximum 1 mA @ 2V
Total Output Current	maximum 7A with exceeding shut down of all outputs with cyclic reconnecting
Response Overload	shut down of the concerned output with cyclic reconnecting
Environmental Conditions	
Storage Temperature :	-40°C to +85°C (-40°F to +185°F) without backup battery
Operating Temperature:	0°C to +60°C (+32°F to +140°F)
Mechanical Dimensions	
Width	207 mm (8.16 in.) including housing screws
Height	114 mm (4.49 in.) including latch
Depth	66 mm (2.60 in.) including grounding bolt
Weight	1.0 kg (2.2 lb.)
Agency Certifications (when product is marked)	C-UL Listed Industrial Control Equipment Marked for all applicable directives N223 Marked for all applicable acts TÜV Functional Safety 1oo2D (AK 1 to 6, SIL 1 to 3, according to DIN V 19250 and IEC 61508 respectively) Category 1 to 4, according to EN954-1

1753-OB16 Output Module

Specifications	1753-OB16
General	
Interfaces: GuardPLC Ethernet	2 x RJ-45, 10/100Base T (with 100 Mbit/s) with integrated switch
Operating Voltage	24V dc, -15% to +20%, w_{ss} 15% from a power supply with protective separation, conforming to IEC 61131-2 requirements
Response Time	≥ 10 ms
Battery Backup	none
Current Consumption	approximately 0.2A per group (idle current)
Digital Outputs	
No. of Outputs	16 (not electrically isolated)
Output Voltage Range	≥ L+ minus 2V
Output Current	maximum 1A @ 60°C (140°F), maximum 2A @40°C (104°F)
Surge Current Per Channel	4A for 10 ms @ 1 Hz
Minimum Current Load	2 mA per channel
Current per Group (admissible total current)	maximum 8A (maximum 16A)
Lamp Load	maximum 10 W (for output 1A), maximum 25 W (for output 2A)
Inductive Load	maximum 500 mH
Internal Voltage Drop	maximum 2V @ 2A
Leakage Current (with 0 signal)	maximum 1 mA @ 2V
Response to Overload	shut down of concerned output with cyclic reconnecting
Environmental Conditions	
Storage Temperature	-40°C to +85°C (-40°F to +185°F)
Operating Temperature	0°C to +60°C (+32°F to +140°F)
Mechanical Dimensions	
Width	207 mm (8.16 in.) including housing screws
Height	114 mm (4.49 in.) including latch
Depth	66 mm (2.60 in.) including grounding bolt
Weight	0.85 kg (1.87 lb.)

Specifications	1753-OB16
Agency Certifications (when product is marked)	C-UL Listed Industrial Control Equipment Marked for all applicable directives N223 Marked for all applicable acts Functional Safety 1oo2D (AK 1 to 6, SIL 1 to 3, according to DIN V 19250 and IEC 61508 respectively) Category 1 to 4, according to EN954-1

GuardPLC 2000 Controller

1755-L1 Specifications	
User Memory	500 KB application code memory 500 KB application data memory
Operating voltages	3.3V dc 5V dc
Current consumptions	3.3V / 1.5A 5V / 0.1A 24V dc / 1.0A
Front connectors	1 Ethernet connector for RSLogix Guard PLUS 2 ASCII connectors (RS-232)
Operating temperature	0°C to +60°C (+32°F to +140°F)
Storage temperature	-40°C to +85°C (-40°F to 185°F)
Weight	280 g (0.62 lb.)
Agency Certifications (when product is marked)	UL Listed Industrial Control Equipment Marked for all applicable directives N223 Marked for all applicable acts Functional Safety 1oo2D (AK 1 to 6, SIL 1 to 3, according to DIN V 19250 and IEC 61508 respectively) Category 1 to 4, according to EN954-1

GuardPLC 2000 Distributed I/O Modules

1755-IB24XOB16 Digital I/O Module

1755-IB24XOB16 Specifications	
Digital Inputs	
Quantity of inputs	24
Nominal input voltage (1 signal)	24V dc (10 to 30V)
Off-state input voltage (0 signal)	max. 5V dc
ON state current	2 mA at 10V, 13 mA at 30V (3 groups of 8, each group limited to 100 mA)
OFF state current	1.5 mA at 5V
Digital Outputs	
Quantity of outputs	16
Output voltage range	operating voltage minus 2V (depending on load)
Output current (30 °C)	2A per channel, overload protected, max. 8A per module
General Specifications	
Current consumption	0.3A / 3.3V dc 0.5A / 24V dc (Idle current to run module)
Operating voltage	24V dc, -15 to +20%, ripple ≤ 15%
Operating temperature	0°C to +60°C (+32°F to +140°F)
Storage temperature	-40°C to +85 °C (-40°F to +185°F)
Weight	260 g (0.57 lb.)
Agency Certifications (when product is marked)	UL Listed Industrial Control Equipment Marked for all applicable directives N223 Marked for all applicable acts Functional Safety 1oo2D (AK 1 to 6, SIL 1 to 3, according to DIN V 19250 and IEC 61508 respectively) Category 1 to 4, according to EN954-1

1755-IF8 Analog Input Module

1755-IF8 Specifications	
Number of inputs	8 single-ended or 4 differential
Input values rated values user values	 0 to ±10V dc or 0 to +20 mA (with shunt) 0 to ±10.25V dc or 0 to +20.5 mA (with shunt)
External shunt (for current input)	500 Ω
Overvoltage protection	30V (±15V dc)
Resolution	12 bit
Input impedance	1 MΩ (DC)
Input signal / source impedance	≤500 Ω
Accuracy	0.1% at 25°C (77°F) 0.5% at 60°C (140°F)
Operating voltage	24V dc -15 to +20% ripple ≤15%
Maximum common mode voltage to I-	±13V dc
Current consumption	150 mA / 3.3V dc 400 mA / 24V dc
Operating temperature	0°C to +60°C (+32°F to +140°F)
Storage temperature	-40°C to +85°C (-40° to +185°F)
Weight	240 g (0.53 lb.)
Agency Certifications (when product is marked)	UL Listed Industrial Control Equipment Marked for all applicable directives N223 Marked for all applicable acts Functional Safety 1oo2D (AK 1 to 6, SIL 1 to 3, according to DIN V 19250 and IEC 61508 respectively) Category 1 to 4, according to EN954-1

1755-OF8 Analog Output Module

1755-OF8 Specifications	
Quantity of outputs	8
Max. output values	0 to ±10V or 0 to +20 mA
Overvoltage protection	24V
Source value	UINT
Load impedance	load ≤ 600 Ω (current) limit resistance > 5 kΩ (voltage)
Accuracy	0.3% at +25°C (+77°F) 0.5% at +60°C (+140°F)
Safety relevant accuracy	1%
Operating voltage	24V dc -15 to +20% ripple ≤ 15%
Current consumption	150 mA / 3.3V dc 400 mA / 24V dc
Operating temperature	0°C to +60°C (+32°F to +140°F)
Storage temperature	-40°C to +85°C (-40°F to +185°F)
Weight	280 g (0.53 lb.)
Agency Certifications (when product is marked)	UL Listed Industrial Control Equipment Marked for all applicable directives N223 Marked for all applicable acts Functional Safety 1oo2D (AK 1 to 6, SIL 1 to 3, according to DIN V 19250 and IEC 61508 respectively) Category 1 to 4, according to EN954-1

1755-HSC High Speed Counter Module

1755-HSC Specifications	
Number of counters	2
Input voltage	5V or 24V
Input current	≤3 mA
Input signal frequency	0 to 1 MHz
Trigger	with falling edge
Edge Steepness	1V/μs
Input cables	≤500 m at 100 kHz, shielded, twisted
Input resistance	3.7 kΩ
Resolution	24 bit (value range 0 to 16,777,215)
Accuracy of time basis	0.2%
Quantity of outputs	4 digital
Output load	≤0.5A, voltage drop: ≤3V
Output load in summary	≤2A ≥ 18V
Operating Voltage	24V dc, -15 to +20%, ripple ≤15%
Current consumption	0.1A / 24V dc without load 0.8A (3.3V dc), 0.1A (5V dc)
Operating temperature	0°C to +60°C (+32°F to +140°F)
Storage temperature	-40°C to +85°C (-40°F to +185°F)
Weight	260 g (0.57 lb.)
Agency Certifications (when product is marked)	UL Listed Industrial Control Equipment Marked for all applicable directives N223 Marked for all applicable acts Functional Safety 1oo2D (AK 1 to 6, SIL 1 to 3, according to DIN V 19250 and IEC 61508 respectively) Category 1 to 4, according to EN954-1

GuardPLC 2000 Power Supply

1755-PB720 Specifications	
Supply voltage	24V dc
Supply voltage range	20.4V dc to 28.8V dc (10 msec buffer), ripple ≤15%
External fusing	30A[1] / IEC (This module has no overcurrent protection.)
Outputs	3.3V dc/10A, 5V dc/2A
Operating temperature	0°C to +60°C (+32°F to +140°F)
Storage temperature	-40°C to +60°C (-40°F to +140°F) with battery -40°C to +85°C (-40°F to +185°F) without battery
Weight	820 g (1.80 lb.)
Agency Certifications (when product is marked)	UL Listed Industrial Control Equipment Marked for all applicable directives Marked for all applicable acts N223 Functional Safety 1oo2D (AK 1 to 6, SIL 1 to 3, according to DIN V 19250 and IEC 61508 respectively) Category 1 to 4, according to EN954-1

(1) The power supply can supply up to 30A for I/O modules. Use an appropriate fuse for your system’s power requirements.

System Variables

Using This Appendix

Programming Controller Data

The controller supports system variables that you can configure.

The system variables are defined as:

- **SAFE**: the controller can use this information in safety-related functions
- **NON-SAFE**: additional information that safety functions must not rely on

The system variables are:

System Variable:	Unit/Value:	Read/Write:	Description:(1)
Contact Assembly 1 Contact Assembly 2 Contact Assembly 3 Contact Assembly 4	true false	Write	On true, the contact closes; on false the contact does not close. Only available for a GuardPLC 2000 controller. [BOOL] **NON-SAFE**
Cooling Fan State	0, 1, 2	Read	0 = normal 1 = fans OK 2 = fan error Only available for a GuardPLC 2000 controller. [BYTE] **NON-SAFE**
Cycle Time	milliseconds	Read	Duration of the last cycle [UDINT] **SAFE**
Date Time Seconds Date Time Milliseconds	seconds milliseconds	Read	Time passed since 1970. An automatic switchover from summer to winter time is not supported. [UDINT] **NON-SAFE**
Emergency Stop 1 Emergency Stop 2 Emergency Stop 3 Emergency Stop 4	true false	Write	True triggers Emergency Off [BOOL] **SAFE** Use these signals to force all inputs and outputs to the zero/OFF state from within the user program.

System Variable:	Unit/Value:	Read/Write:	Description:[1]	
Force Time	milliseconds	Read	Remaining running time during forcing; 0 if Force is inactive. [DINT] **NON-SAFE**	
Power Supply	0-255	Read	**GuardPLC 1200 and GuardPLC 2000** 0 = normal 1 = error of input power supply 24 VDC 2 = error of battery 4 = module error of power supply 5 V 8 = module error of power supply 3.3 V 16 = 5 V undervoltage 32 = 5 V overvoltage 64 = 3.3 V undervoltage 128 = 3.3 V overvoltage 255 = status does not exist [BYTE] **NON-SAFE**	**GuardPLC 1600 and GuardPLC 1800** 0 = normal 1 = 24 VDC undervoltage 4 = 5 V undervoltage 8 = 3.3 V undervoltage 16 = 3.3 V overvoltage [BYTE] **NON-SAFE**
System Tick High System Tick Low	milliseconds	Read	Ring counter with 64 bits which is incremented in millisecond steps. [UDINT] **SAFE**	
Temperature State	0, 1, 2, 3, 255	Read	0 = normal 1 = high 2 = faulty 3 = very high 255 = status does not exist [BYTE] **NON-SAFE** (but for additional switch-off)	

(1) Binary values are ORed.

I/O Variables

Depending upon the type of controller, the various GuardPLC controllers support variables for digital and analog I/O parameters that you can configure or monitor.

Digital I/O Module Variables (AB-DIO) for GuardPLC 1200 and 2000

The GuardPLC 1200 and 2000 controllers support these digital I/O parameters:

I/O Data:	Read/Write:	Description:	
Board.SRS	Read	System.Rack.Slot	
Board.Type	Read	Module type	
		0x00E1	digital I/O module for GuardPLC 1200 controller
		0xFE01	digital I/O module for GuardPLC 2000 controller
		0xFFFF	missing module in GuardPLC 2000 chassis
Board.State(1)	Read	Error mask for the module	
		0x000	I/O processing may be running with errors
		0x001	No I/O processing (CPU not in RUN)
		0x002	No I/O processing during start-up tests
		0x004	Manufacturing interface running
		0x010	No I/O processing due to faulty parameterization
		0x020	No I/O processing due to exceeded fault rate
		0x040	No I/O processing because configured module is not plugged in
DO.State(1)	Read	Error mask for all digital outputs	
		0x0000	No errors detected
		0x0001	Error of the DO section of the module
		0x0002	Within the multiple error occurrence time: safety switch 1 faulty
		0x0004	Within the multiple error occurrence time: safety switch 2 faulty
		0x0008	Within the multiple error occurrence time: test sample tests faulty
		0x0010	Within the multiple error occurrence time: readback channels faulty
		0x0020	Within the multiple error occurrence time: active switch-off faulty
		0x0100	Within the safety time: CS signals faulty
		0x0200	All outputs switched off; total current too high
		0x0400	Within the safety time: temperature limit 1 exceeded
		0x0800	Within the safety: temperature limit 2 exceeded
		0x01000	Within the safety time: auxiliary voltage monitoring: undervoltage
		0x02000	Within the multiple error occurrence time: status of the safety switches

I/O Data:	Read/Write:	Description:	
DO[0x].State[1][2]	Read	Error mask for digital output channels	
		0x00	No error detected; outputs driven as expected
		0x01	Error in digital output module; outputs not driven
		0x02	Output switched off due to overcurrent; outputs not driven
		0x04	Error during readback of the digital output; outputs not driven
DO[0x].Value[1]	Write	Output value of digital output channels	
		0	Output de-energized
		1	Output activated
DI.State	Write	Error mask for all digital inputs	
		0x0000	No error detected
		0x0001	Error of the DI section of the module
		0x0002	Within the safety time: test sample test faulty
DI[xy].State[3]	Read	Error mask of digital input channels	
		0x00	No error detected
		0x01	Error in the digital input module; input value set to 0
DI[xy].Value[2]	Read	Input values of digital input channels	
		0	Input not activated
		1	Input activated

(1) Values are ORed.
(2) 0x = output channel 01 to 16 for GuardPLC 2000 controller and 01 to 08 for GuardPLC 1200, 1600, and 1800 controllers.
(3) xy = input channel 01 to 24 for GuardPLC 2000 and GuardPLC 1800 controllers and 01 to 20 for GuardPLC 1200 and 1600 controllers.

Analog Input Module Variables (AB-AI) for GuardPLC 2000

The GuardPLC 2000 controller supports these analog input parameters:

I/O Data:	Read/Write:	Description:	
AI.Mode	Write	Mode for all channels of the analog input module	
		0	unipolar (single-ended)
		1	differential
AI.State	Read	Error mask for all analog inputs	
		0x0000	No errors detected
		0x0001	Error of the module
		0x0008	Within the safety time: data bus walking bit error
		0x0010	Within the safety time: coefficient table check error
		0x0020	Within the safety time: supply voltages error
		0x0040	Error on A/D conversion (DRDY_HIGH)
		0x0080	Within the multiple error occurrence time: error in multiplexer crosslink
		0x0100	Within the multiple error occurrence time: data bus walking bit error
		0x0200	Within the multiple error occurrence time: multiplexer address error
		0x0400	Within the multiple error occurrence time: supply voltages error
		0x0800	Within the multiple error occurrence time: error in characteristic curve (unipolar mode)
		0x1000	Within the multiple error occurrence time: limit values/zero point error (unipolar mode)
		0x2000	Within the multiple error occurrence time: error in characteristic curve (differential mode)
		0x4000	Within the multiple error occurrence time: limit values/zero point error (differential mode)
		0x8000	Error in A/D conversion (DRDY_LOW)
AI[0x].State(1)	Read	Error mask for analog input channels	
		0x00	No error detected
		0x01	Error in analog input channel
		0x02	Invalid measurement values
		0x04	A/D converters faulty
		0x08	Measurement values are not within the safety accuracy
		0x10	Measurement value overflow
		0x20	Channel not in use
		0x40	Addressing error of the two A/D converters

I/O Data:	Read/Write:	Description:	
AI[0x].Used	Write	Configuration of analog input channel	
		0	not used
		1	used
AI[0x].Value(1)	Read	Analog value of input channel (WORD) -10V to +10V = -1000 to +1000	
Board.SRS	Read	System.Rack.Slot	
Board.Type	Read	Module type	
		0xFD02	analog input module for GuardPLC 2000 controller
		0xFFFF	missing module in GuardPLC 2000 chassis
Board.State	Read	Error mask for the module	
		0x000	I/O processing may be running with errors
		0x001	No I/O processing (CPU not in RUN)
		0x002	No I/O processing during start-up tests
		0x004	Manufacturing interface running
		0x010	No I/O processing due to faulty parameterization
		0x020	No I/O processing due to exceeded fault rate
		0x040	No I/O processing because configured module is not plugged in

(1) 0x = input channel 01 to 08.

Analog Output Module Variables (AB-AO) for GuardPLC 2000

The GuardPLC 2000 controller supports these analog output parameters:

I/O Data:	Read/Write:	Description:	
AO.State	Read	Error mask for all analog outputs	
		0x0000	No errors detected
		0x0001	Error of the module
		0x0002	Within the safety time: co-efficient table check error
		0x0004	No communication with the module due to controller error
AO[0x].Mode	Write	Mode of analog output channel	
		0	voltage
		1	current

I/O Data:	Read/Write:	Description:	
AO[0x].State(1)	Read	Error mask for analog output channels	
		0x0000 0001	CPU detected error on AB-AO module
		0x0000 0002	CPU detected faulty monotony counter
		0x0000 0004	CPU detected error in safe addressing
		0x0000 0008	CPU detected faulty CRC
		0x0000 0010	CPU detected error in watchdog time of the AB-AO onboard microprocessor
		0x0000 0020	CPU cannot communicate with the AB-AO onboard microprocessor
		0x0000 0040	CPU detected that the present operating mode (current/voltage) is different from the initialized operating mode
		0x0001 0000	AB-AO onboard microprocessor detected read back error
		0x0004 0000	AB-AO onboard microprocessor detected wrong supply voltage
		0x0008 0000	Within the multiple error occurrence time: AB-AO onboard microprocessor detected faulty safety switch
		0x0080 0000	AB-AO onboard microprocessor detected both safety switches as faulty
		0x0200 0000	AB-AO onboard microprocessor INITIALIZE
		0x1000 0000	AB-AO onboard microprocessor detected error because of module over temperature
		0x2000 0000	AB-AO onboard microprocessor detected module over temperature
		0x8000 0000	CPU detected error on redundant AB-AO onboard microprocessor channel
AO[0x].Used	Write	Configuration of analog output channel	
		0	not used
		1	used
AO[0x].Value(1)	Write	Output value of analog output channels Voltage mode: -10V to +10V = -1000 to +1000 Current mode: 0mA to +20mA = 0 to +1000 for values between -1000 to 0, the output current is 0mA	
Board.SRS	Read	System.Rack.Slot	
Board.Type	Read	Module type	
		0xFB04	analog output module for GuardPLC 2000 controller
		0xFFFF	missing module in GuardPLC 2000 chassis

I/O Data:	Read/Write:	Description:	
Board.State	Read	Error mask for the module	
		0x000	I/O processing may be running with errors
		0x001	No I/O processing (CPU not in RUN)
		0x002	No I/O processing during start-up tests
		0x004	Manufacturing interface running
		0x010	No I/O processing due to faulty parameterization
		0x020	No I/O processing due to exceeded fault rate
		0x040	No I/O processing because configured module is not plugged in

(1) 0x = output channels 01 to 08.

High-Speed Counter Variables For GuardPLC 1200 and 2000

The GuardPLC 1200 and GuardPLC 2000 controllers support the following variables for counter I/O parameters:

I/O Data:	Read/Write:	Description:	
Board.SRS	Read	System.Rack.Slot	
Board.Type	Read	Module type	
		0x0003	counter module for GuardPLC 1200 controller
		0xFC03	counter module for GuardPLC 2000 controller
		0xFFFF	missing module in GuardPLC 2000 chassis
Board.State	Read	Error mask for the module	
		0x000	I/O processing may be running with errors
		0x001	No I/O processing (CPU not in RUN)
		0x002	No I/O processing during start-up tests
		0x004	Manufacturing interface running
		0x010	No I/O processing due to faulty parameterization
		0x020	No I/O processing due to exceeded fault rate
		0x040	No I/O processing because configured module is not plugged in

I/O Data:	Read/Write:	Description:	
Cnt.State	Read	Error mask of both counters	
		0x0000	No errors detected
		0x0001	Error of the counter section of the module
		0x0002	Error while comparing the time base
		0x0004	Addressing error while reading the time base
		0x0008	Parameterization of the time base corrupted
		0x0010	Addressing error while reading the counts
		0x0020	Parameterization of counter corrupted
		0x0040	Addressing error while reading the Gray codes
		0x0080	Within the multiple error occurrence time: test sample test faulty
		0x0100	Error of the module
Cnt[0x].Value(1)	Read	Counts of counter 1 or 2 (cyclic 24-bit) 24 bits for pulse counter 4 bits for Gray code for GuardPLC 2000; 3 bits for Gray code for GuardPLC 1200	
Cnt[0x].5/24V Mode(1)	Read/Write	5V or 24V mode of counter 1 or 2 The write values must have initial values or constants.	
		0	5V
		1	24V
Cnt[0x].Auto Advance Sense(1)	Read/Write	Automatic recognition of direction of counting for counter 1 or 2	
		0	Manual setting of direction of counting
		1	Automatic recognition of direction of counting
Cnt[0x].Direction(1)	Read/Write	Direction of counting for counter 1 or 2 (only when Automatic Counter Advance Sense = false)	
		0	Up
		1	Down
Cnt[0x].Dummy1	Read/Write	reserved memory space for future use	
Cnt[0x].Dummy2	Read/Write	reserved memory space for future use	
Cnt[0x].GrayCode(1)	Read/Write	Gray code mode of counter 1 or 2	
		0	Pulse
		1	Gray
Cnt[0x].Halt(1)	Read/Write	currently not used	
Cnt[0x].Reset(1)	Read/Write	Reset for counter 1 or 2	
		0	Resetting of counter
		1	No resetting of counter

I/O Data:	Read/Write:	Description:	
Cnt[0x].State(1)	Read	Error mask of counter 1 or 2	
		0x01	Error in counter unit
		0x02	Error while comparing the counts
		0x04	Error while comparing the time stamps
		0x08	Error resetting counter
Cnt[0x].Time Overflow(1)	Read	Overflow indicator of time stamp of counter 1 or 2	
		true	24 bits overflow since last cycle
		false	No 24 bits overflow since last cycle
Cnt[0x].Time Stamp(1)	Read	Time stamp for Cnt[0x].Value (cyclic 24-bit) 24 bits, time resolution 1µs	
Cnt[0x].Value Overflow(1)	Read	Overflow indicator of counter 1 or 2	
		true	24 bits overflow since last cycle (only when Automatic Counter Advance Sense = false)
		false	No 24 bits overflow since last cycle
DO.State	Read	Error mask for all counter outputs	
		0x0001	Error of the DO section of the module
		0x0002	Within the multiple error occurrence time: safety switch 1 faulty
		0x0004	Within the multiple error occurrence time: safety switch 2 faulty
		0x0008	Within the multiple error occurrence time: test sample tests faulty
		0x0010	Within the multiple error occurrence time: readback channels faulty
		0x0020	Within the multiple error occurrence time: active switch-off faulty
		0x0100	Within the safety time: CS signals faulty
		0x0200	All outputs switched off; total current too high
		0x0400	Within the safety time: temperature limit 1 exceeded
		0x0800	Within the safety time: temperature limit 2 exceeded
		0x01000	Within the safety time: auxiliary voltage monitoring: undervoltage
		0x02000	Within the multiple error occurrence time: status of the safety switches
DO[0y].State(2)	Read	Error mask for counter outputs 1 to 4	
		0x01	Error in output channel
		0x02	Output channel switched off due to overcurrent
		0x04	Error during readback of the output channel
		0x08	Faulty initialization after counter reset
DO[0x].Value(2)	Write	Output value of counter outputs 1 to 4 (These 4 outputs cannot be driven by counter presets. They are driven by user software only.)	
		0	Output de-energized
		1	Output activated

(1) 0x = counter 01 or 02.

(2) 0y = outputs 01, 02, 03, or 04

Digital Input Module Variables for GuardPLC 1600 and DIO

The GuardPLC 1600 controllers and distributed I/O support the following digital input parameters.

I/O Data:	Read/Write	Description:	
Module.SRS	Read	Slot number (System.Rack.Slot)	
Module.Type	Read	Module type	
		0x00A5	Digital input module (DI20) for GuardPLC 1600
		0x00A5	Digital input module (DI20) for 1753-IB20XOB8
		0x002D	Digital input module (DI16) for 1753-IB16
Module.Error.Code	Read	Error mask for the module	
		0x0000	I/O processing may be running with errors
		0x0001	No I/O processing (CPU not in RUN)
		0x0002	No I/O processing during start-up tests
		0x0004	Manufacturing interface running
		0x0010	No I/O processing due to incorrect configuration
		0x0020	No I/O processing due to exceeded fault rate
		0x0040	No I/O processing because configured module is not plugged in
DI.Error Code	Read	Error mask for all digital inputs	
		0x0001	Error in digital input range
		0x0002	FTZ test of test pattern failed
DI[xy].Error Code(1)	Read	Error mask of all digital input channels	
		0x01	Error in digital input module
		0x10	Short-circuit of the channel
		0x80	Line interrupt between pulse output (DO) and pulse input (DI)
DI[xy].Value(1)	Write	Input value of digital input channels	
		0	Input not set
		1	Input set

I/O Data:	Read/Write	Description:	
DI.Number of Pulse Channel	Write	Number of pulse outputs (feed outputs)	
		0	No output channel provided for line monitoring
		1	Output channel 1 provided for line monitoring
		2	Output channels 1 and 2 provided for line monitoring
		3	Output channels 1, 2, and 3 provided for line monitoring
		4	Output channels 1to 4 provided for line monitoring
		5	Output channels 1to 5 provided for line monitoring
		6	Output channels 1to 6 provided for line monitoring
		7	Output channels 1to 7 provided for line monitoring
		8	Output channels 1to 8 provided for line monitoring
DI.Pulse Slot	Write	Pulse module slot (LC)	
DI.Pulse Channel	Write	Source channel of pulse feed	
		0	Input channel
		1	Pulse from first DO channel
		2	Pulse from second DO channel
		3	Pulse from third DO channel
		4	Pulse from fourth DO channel
		5	Pulse from fifth DO channel
		6	Pulse from sixth DO channel
		7	Pulse from seventh DO channel
		8	Pulse from eighth DO channel
DI.LC Delay	Write	Waiting time for pulse output (short-circuit-proof)	

(1) xy = input channel 01 to 24 for GuardPLC 1800 controllers and 01 to 20 for GuardPLC 1600 controllers.

Digital Output Module Variables for GuardPLC 1600, 1800 and DIO

The GuardPLC 1600 and 1800 controllers and distributed I/O support the following digital output parameters.

I/O Data:	Read/Write	Description:	
Module.SRS	Read	Slot number (System.Rack.Slot)	
Module.Type	Read	Module type	
		0x00B4	digital output module (DO8) for GuardPLC 1600, 1800, and 1753-IB20XOB8
		0x005A	digital output module (DO16) for 1753-OB16
Module.Error.Code	Read	Error mask for the module	
		0x0000	I/O processing may be running with errors
		0x0001	No I/O processing (CPU not in RUN)
		0x0002	No I/O processing during start-up tests
		0x0004	Manufacturing interface running
		0x0010	No I/O processing due to incorrect configuration
		0x0020	No I/O processing due to exceeded fault rate
		0x0040	No I/O processing because configured module is not plugged in
DO.Error Code	Read	Error mask for all digital outputs	
		0x0001	Error in digital output range
		0x0002	MEZ test of test pattern failed
		0x0004	MEZ test, auxiliary supply failed
		0x0010	FTZ test of test pattern failed
		0x0020	FTZ test of test pattern of the output switch failed
		0x0040	FTZ test of the test pattern of the output switch (disconnection test of outputs) failed
DO[0x].Error Code(1)	Read	Error mask of all digital output channels	
		0x01	Error in digital output module
		0x02	Output switched off due to overload
		0x04	Error when reading back the activation of the digital outputs
		0x08	Error when reading back the status of the digital outputs

(1) 0x = output channels 01 to 08.

Counter Module Variables for GuardPLC 1800 Controllers

The GuardPLC 1800 controllers support the following counter parameters.

I/O Data:	Read/Write	Description:	
Module.SRS	Read	Slot number (System.Rack.Slot)	
Module.Type	Read	Module type	
		0x0003	high speed counter module for GuardPLC 1800
Module.Error.Code	Read	Error mask for the module	
		0x0000	I/O processing may be running with errors
		0x0001	No I/O processing (CPU not in RUN)
		0x0002	No I/O processing during start-up tests
		0x0004	Manufacturing interface running
		0x0010	No I/O processing due to incorrect configuration
		0x0020	No I/O processing due to exceeded fault rate
		0x0040	No I/O processing because configured module is not plugged in
Cnt.Error Code	Read	Error mask of counter module	
		0x0001	Error in module
		0x0002	Error comparing the time base
		0x0004	Address error reading the time base
		0x0008	Parameters for the time base are faulty
		0x0010	Address error reading the counter content
		0x0020	Configuration of counter damaged
		0x0040	Address error reading the Gray Code
		0x0080	FTZ test of the test pattern failed
		0x0100	FTZ test, error checking the coefficients
Cnt[0x].Error Code(1)	Read	Error mask of counter channels 1 and 2	
		0x01	Error in counter module
		0x02	Error comparing contents of counters
		0x04	Error comparing the timestamps of the counters
		0x08	Error setting the parameters (reset)
Cnt[0x].Value(1)	Read	Content of counters: 24-bit for pulse counter, 3-bit for Gray Code	
Cnt[0x].Timestamp(1)	Read	Time stamp for *Cnt[0x].Value* 24-bit, time resolution 1μs	
Cnt[0x].Value Overflow(1)	Read	Counter overflow indication	
		True	24-bit overflow since last measurement (only if *Cnt[0x].Auto Advance Sense* = False)
		False	No overflow since last cycle

I/O Data:	**Read/Write**	**Description:**	
Cnt[0x].Time Overflow[1]	Read	Overflow indication for the time stamp of the counters	
		True	24-bit overflow since last measurement
		False	No 24-bit overflow since last measurement
Cnt[0x].Direction[1]	Read/Write	Counting direction of the counter (only if *Cnt[0x].Auto Advance Sense* = False)	
		True	upward (increment)
		False	downward (decrement)
Cnt[0x].Auto Advance Sense[1]	Read/Write	Automatic counter direction recognition	
		True	Automatic recognition on
		False	Manual setting of counter direction
Cnt[0x].Reset[1]	Read/Write	Reset counter	
		True	No reset
		False	Reset
Cnt[0x].5/24V Mode[1]	Read/Write	Counter input 5V or 24V	
		True	24V
		False	5V
Cnt[0x].Gray Code[1]	Read/Write	Decoder or pulse operation	
		True	Gray Code decoder
		False	Pulse operation

(1) 0x = counter 01 or 02.

Digital (Analog) Input Variables for the GuardPLC 1800 Controller

The digital inputs on the GuardPLC 1800 are actually analog inputs with the following configurable parameters:

I/O Data:	Read/Write	Description:	
Module.SRS	Read	Slot number (System.Rack.Slot)	
Module.Type	Read	Module type	
		0x00D2	Digital input module (MI24/8 FS:1000) for GuardPLC 1800
		0x0096	Digital input module (MI24/8 FS:2000) for GuardPLC 1800
Module.Error.Code	Read	Error mask for the module	
		0x0000	I/O processing may be running with errors
		0x0001	No I/O processing (CPU not in RUN)
		0x0002	No I/O processing during start-up tests
		0x0004	Manufacturing interface running
		0x0010	No I/O processing due to incorrect configuration
		0x0020	No I/O processing due to exceeded fault rate
		0x0040	No I/O processing because configured module is not plugged in
AI.Error Code	Read	Error mask for all digital (analog) inputs	
		0x0001	Error in input range
		0x0008	FTZ test: walking bit of data bus faulty
		0x0010	FTZ test: error checking coefficients
		0x0020	FTZ test: operating voltages faulty
		0x0040	A/D conversion faulty (DRDY_LOW)
		0x0080	MEZ test: cross links of MUX faulty
		0x0100	MEZ test: walking bit of data bus faulty
		0x0200	MEZ test: multiplexer addresses faulty
		0x0400	MEZ test: operating voltages faulty
		0x0800	MEZ test: measuring system (characteristic) faulty (unipolar)
		0x1000	MEZ test: measuring system (final values, zero point) faulty (unipolar)
		0x8000	A/D conversion faulty (DRDY_HIGH)

I/O Data:	Read/Write	Description:	
AI[xx].Error Code DI[xx].Error Code	Read Read	Error mask for analog input channels (1 to 8) Error mask for digital input channels (9 to 32)	
		0x01	Error in input module
		0x02	Measured values invalid
		0x04	A/D converter faulty
		0x08	Measured value not within the safety accuracy
		0x10	Measured value overflow
		0x20	Channel not in operation
		0x40	Address error of both A/D converters
		0x80	Configuration of hysteresis faulty
AI[xx].Value Analog	Read	Analog value of AI channels (1 to 8) [WORD] from 0 to +1000 The validity is dependent on the error mask.	
DI[xx].Value Analog	Read	Analog value of the DI channels (9 to 32) [WORD] from 0 to +3000 The validity is dependent on the error mask.	
DI[xx].Value Bool	Read	Digital value of DI channels (9 to 32) [BOOL] according to hysteresis The validity is dependent on the error mask.	
AI[xx].Hysteresis LOW	Write	Upper limit of the 0-signal voltage range *DI[xx].Value Bool*	
AI[xx].Hysteresis HIGH	Write	Lower limit of the 1-signal voltage range *DI[xx].Value Bool*	
AI[xx].Used DI[xx].Used	Write Write	Configuration for indicating utilization of channels 1 to 8 Configuration for indicating utilization of channels 9 to 32	

Replacing the Backup Battery

The following procedures apply only to GuardPLC 1200 controllers and GuardPLC 2000 power supplies. Other GuardPLC controllers and I/O modules are not equipped with backup batteries.

Preventing Electrostatic Discharge

Only qualified personnel with knowledge of ESD protective measures may replace the back-up battery.

ATTENTION

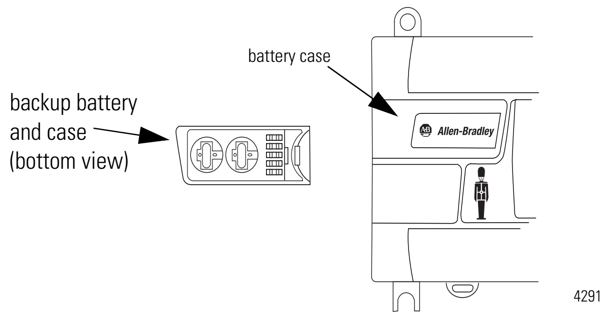
Electrostatic discharge can damage integrated circuits or semiconductors. Follow these guidelines when you handle the module:

- Touch a grounded object to discharge static potential.
- Wear an approved wrist-strap grounding device.
- If available, use a static-safe workstation.
- When not in use, keep the GuardPLC controller in its static-shield box.

GuardPLC 1200

Replace the backup battery on your GuardPLC 1200 controller every two years. The battery case is located on the left-hand side of the cabinet (see drawing below). The battery must be replaced together with the case. Replacements are available from Rockwell Automation under part number 1754-BAT.

Follow the steps on page C-2 to replace the battery.



ATTENTION

Make sure that the GuardPLC 1200 is powered on. Replacing the backup battery while the controller is de-energized causes a reset. All data including the clock settings will be lost.

1. Pull the left side of the battery case toward you to remove the battery case.

2. Insert a new battery case making sure that the case is correctly aligned and the pins inside the GuardPLC 1200 are not bent. Press on the left edge of the case until the battery snaps into place.

Used batteries must be packaged and transported to a proper disposal site in accordance with local regulations.

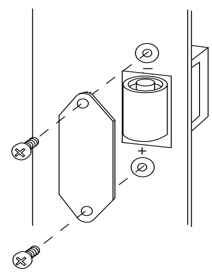
GuardPLC 2000 Power Supply

Replace the backup battery every four years. Replacement batteries are available from Rockwell Automation (1755-BAT). Follow the steps on page C-3 to replace the battery:

ATTENTION

Make sure that the GuardPLC 2000 is powered on. Replacing the backup battery while the controller is off causes a reset. All data including the clock settings will be lost.

1. Remove the lid by removing the two screws.



2. Use a flat-head screwdriver to remove the battery from its compartment.

3. Insert a new battery, following the polarity shown on the compartment.

   Make sure that the contact pins inside the battery compartment are not damaged.

Used batteries must be packaged and transported to a proper disposal site in accordance with local regulations.

Rockwell Automation Support

Rockwell Automation provides technical information on the web to assist you in using our products. At http://support.rockwellautomation.com, you can find technical manuals, a knowledge base of FAQs, technical and application notes, sample code and links to software service packs, and a MySupport feature that you can customize to make the best use of these tools.

For an additional level of technical phone support for installation, configuration and troubleshooting, we offer TechConnect Support programs. For more information, contact your local distributor or Rockwell Automation representative, or visit http://support.rockwellautomation.com.

Installation Assistance

If you experience a problem with a hardware module within the first 24 hours of installation, please review the information that's contained in this manual. You can also contact a special Customer Support number for initial help in getting your module up and running:

United States	1.440.646.3223 Monday – Friday, 8am – 5pm EST
Outside United States	Please contact your local Rockwell Automation representative for any technical support issues.

New Product Satisfaction Return

Rockwell tests all of our products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned:

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

www.rockwellautomation.com

Corporate Headquarters
Rockwell Automation, 777 East Wisconsin Avenue, Suite 1400, Milwaukee, WI, 53202-5302 USA, Tel: (1) 414.212.5200, Fax: (1) 414.212.5201

Headquarters for Allen-Bradley Products, Rockwell Software Products and Global Manufacturing Solutions
Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe: Rockwell Automation SA/NV, Vorstlaan/Boulevard du Souverain 36-BP 3A/B, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, 27/F Citicorp Centre, 18 Whitfield Road, Causeway Bay, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Headquarters for Dodge and Reliance Electric Products
Americas: Rockwell Automation, 6040 Ponders Court, Greenville, SC 29615-4617 USA, Tel: (1) 864.297.4800, Fax: (1) 864.281.2433
Europe: Rockwell Automation, Brühlstraße 22, D-74834 Elztal-Dallau, Germany, Tel: (49) 6261 9410, Fax: (49) 6261 17741
Asia Pacific: Rockwell Automation, 55 Newton Road, #11-01/02 Revenue House, Singapore 307987, Tel: (65) 351 6723, Fax: (65) 355 1733

Publication 1753-UM001A-EN-P - April 2004
Supersedes Publication 1755-UM001C-EN-P - June 2002