ADAPTIVE®

Alpha® Gateway Interface Installation Instructions for Modbus Plus

Contents

Introduction 1
Related documents 2
Basic setup procedure 2
Technical specifications 3
Alpha® Gateway Interface 3
Module descriptions. 4
Module specifications 7
Installation. 8
Network diagrams. 8
Wiring diagrams and settings. 9
PLC programming examples. 13
MSTR block – Global 13
MSTR block – Peer-to-Peer 14
Register mapping – Global. 15
Safety and troubleshooting. 17
Configuration data for Alpha® Gateway networks 19
Appendices 31
Dip switch settings. 31

Introduction

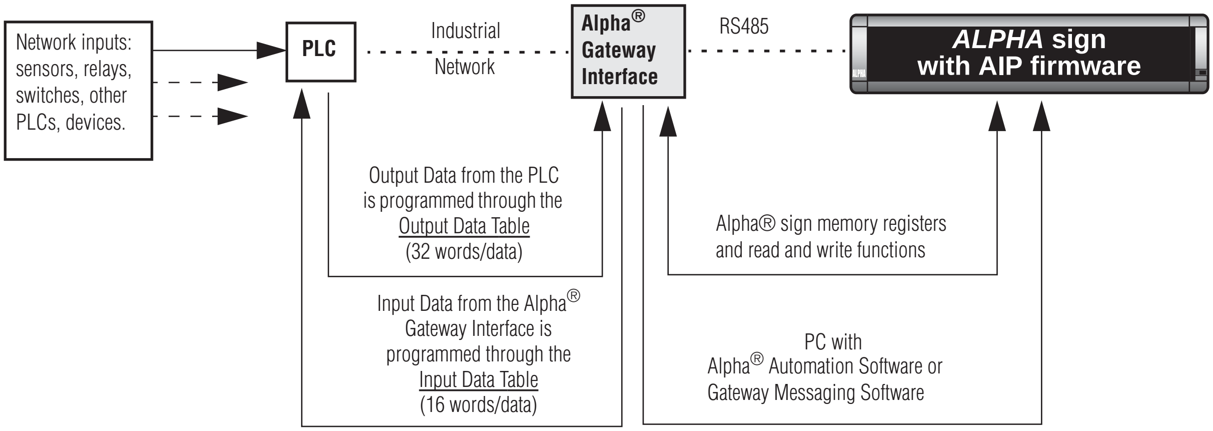
The Alpha® Gateway Interface (“AGI”) allows for the exchange of data between a Modbus Plus PLC and Alpha® signs to activate messages and show real-time data on a system.

The AGI can exchange data in either of two ways:

1. Using fast cyclic I/O data called Global Data
2. Using Modbus protocol for Peer-to-Peer data transfer

Refer to the “Basic setup procedure” on page 2 for an overview of all the steps needed to set up an Alpha® Gateway Interface with messages, hardware, and programming in a Modbus Plus PLC network.

Below is a high-level diagram of how the AGI works.



Network inputs: sensors, relays, switches, other PLCs, devices.
PLC
Industrial Network
Alpha® Gateway Interface
RS485
ALPHA sign with AIP firmware
Output Data from the PLC is programmed through the Output Data Table (32 words/data)
Input Data from the Alpha® Gateway Interface is programmed through the Input Data Table (16 words/data)
Alpha® sign memory registers and read and write functions
PC with Alpha® Automation Software or Gateway Messaging Software

INFORMATION FLOW — In a Modbus Plus network, a “device” is any point in the information pathway capable of sending or receiving a data signal. In the most basic network configuration, as shown above (one input, one PLC, one Alpha® Gateway Interface, one sign), the PLC, the Alpha® Gateway Interface, and the sign are all capable of both sending and receiving data signals.

NOTE: If there is a communication failure between any two points of the information pathway, as shown in the diagram above, messages may fail to display on a sign.

Adaptive Micro Systems • 7840 North 86th Street • Milwaukee, WI 53224 USA • 414-357-2020 • 414-357-2029 (fax) • http://www.adaptivedisplays.com

Revision date: March 4, 2003

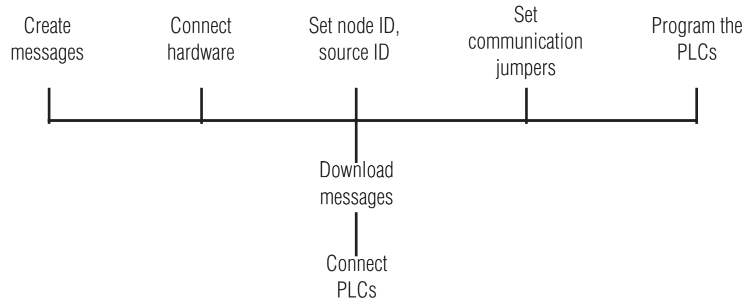
Related documents

Document name	Part number	Description
Alpha® Automation Software online Help system	—	Describes how to use the Alpha® Automation Software to compose and display messages on Alpha® signs.
Configuration Data for Alpha® Gateway Networks	TechMemo 00-0006	Operating summary, programming reference
Network Configurations	9708-8046	How to network Alpha® signs. Includes specific information on routing long distance RS485 connections in “Appendix G”.
Modbus Plus Network Planning and Installation Guide	—	Available from AEG Schneider Automation, Inc.
Alpha® Industrial Protocol (AIP)	9711-8814	Describes how the Modbus protocol is incorporated into the Alpha® Industrial Protocol firmware. This firmware allows signs to connect to a Modbus communications network.

Basic setup procedure

The sequence of these steps is recommended, but can be modified as shown in the flowchart below.

1. Create sign messages using a computer with the Alpha® Automation Software or Gateway Messaging Software. (Refer to the software’s online Help.)
2. Connect the computer, Alpha® sign(s), and, if needed, the Alpha® Gateway Interface, using one of the “Network diagrams” on page 8.
3. Download the sign messages to one or all of the signs on the network using the messaging software.
4. Determine the node ID and source ID addresses for the AGI. Then set the DIP switches for these addresses on the Network Interface Module. (See “Network Interface Module” on page 6, “Dip switch settings” on page 31, and the **Modbus Plus Network Planning and Installation Guide**.)
5. Determine the communication profile needed for your application: Global or Peer-to-Peer. Then set the jumpers for these on the Network Interface Module. See “Network Interface Module settings” on page 11.
6. Connect the AGI and Alpha® sign(s) to the PLC network, if not already done, using either the recommended setup or the optional setup shown in the “Network diagrams” on page 8.
7. Program the PLC using one of the following techniques:
   - MSTR block for Global communication — See “MSTR block – Global” on page 13.
   - MSTR block for Peer-to-Peer communication — See “MSTR block – Peer-to-Peer” on page 14.
   - Register mapping for Global communication — See “Register mapping – Global” on page 15.



Technical specifications

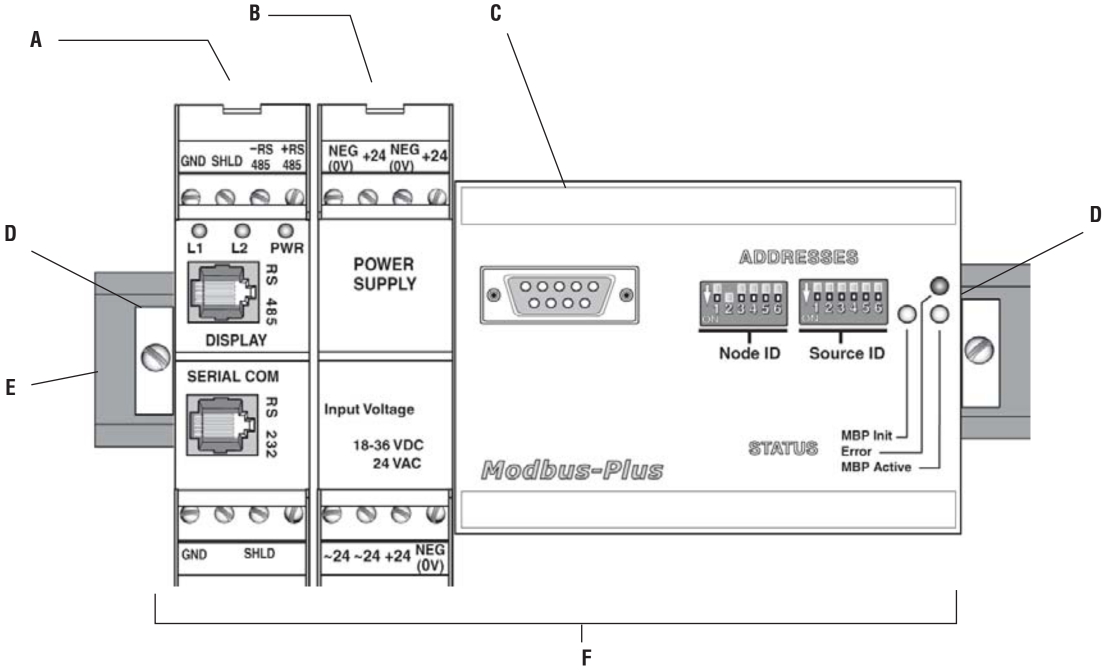
Alpha® Gateway Interface

The Alpha® Gateway Interface consists of three modules:

- CPU Module
- Power Supply Module
- Network Interface Module

The modules of the complete interface are mounted together on a DIN rail using end clamps.

NOTE: The CPU Module and Power Supply Module are physically interchangeable. Either one can be mounted to the left of the Network Interface Module.

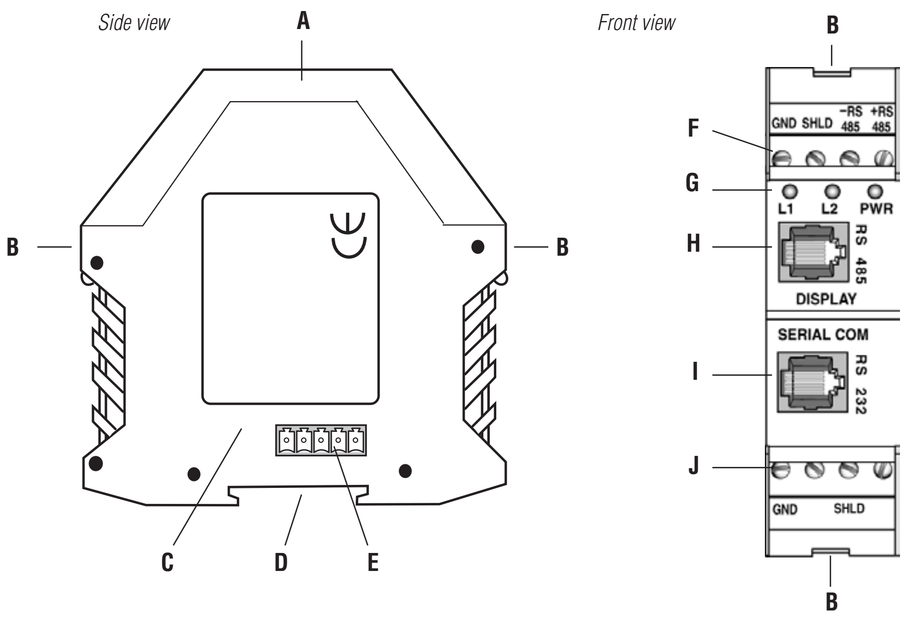


Item	Part Number	Name	Description
A	E9920-1116	CPU Module	Provides the connection to the Alpha® sign network.
B	E9920-1113	Power Supply Module	Supplies power for both the CPU Module and the Network Interface Module.
C	E9920-1115	Network Interface Module	Provides the connection to the PLC network. NOTE: All Network Interface Modules have the same Adaptive® part number. Specify your network when ordering.
D	—	End clamp	Use end clamps (2) to hold the modules in place.
E	—	DIN rail	The rail should be connected to earth GND because it serves as chassis GND for all the modules.
F	1188-9205	Complete assembly	All three component modules

Module descriptions

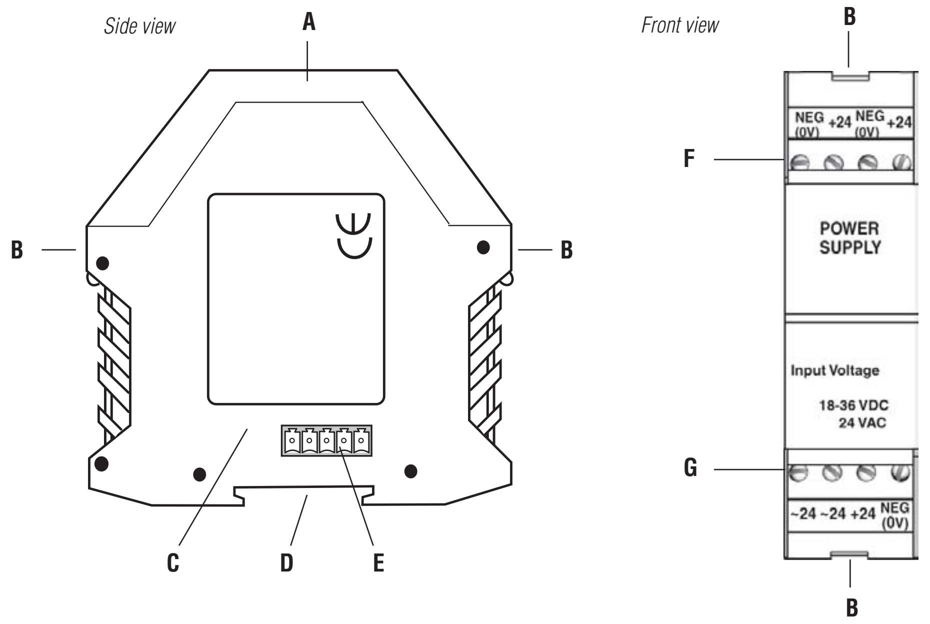
The Alpha® Gateway Interface is assembled with three modules that are described in detail below.

CPU Module



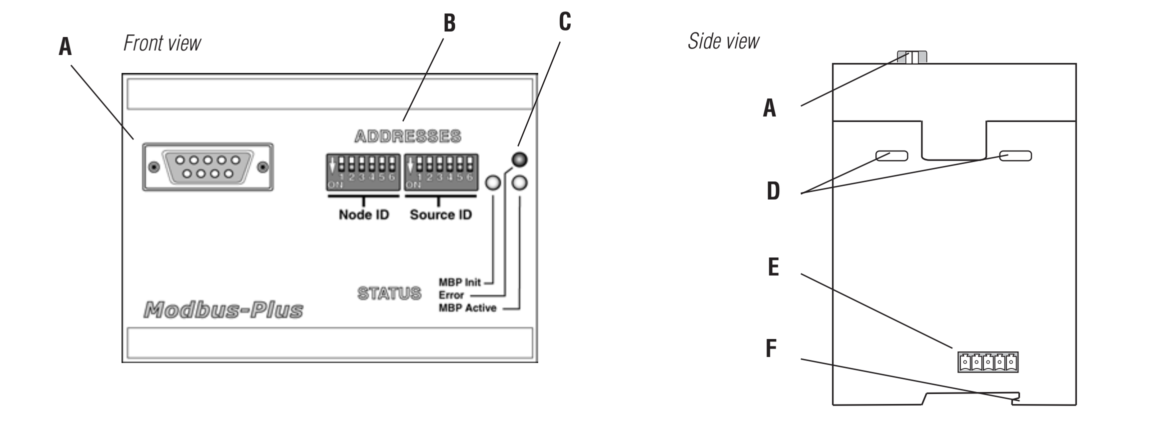
Item	Name	Description	
A	Module top	The internal PCB is attached to the Module top.	
B	Release buttons	Use a pen tip or other pointed tool to push release buttons and pull off the module bottom.	
C	Module bottom	After release buttons (item B) are unlatched, module bottom can be pulled away to expose internal PCB, to determine the revision of the firmware or the CPLD.	
D	DIN rail latch	Used to attach the module to a DIN mounting rail.	
E	Intermodule connector	Used to pass power and signals between modules.	
F	Serial output	Shielded RS485 output to one or more Alpha® signs.	
G	Status and diagnostic LEDs	L1	Red LED (Mode): • Flashes once a second while data is being downloaded to Alpha® signs. (Download mode) • Flashes when receiving data from an Alpha® sign. (Normal/receiving mode) • Flashes when the AGI encounters a fault. (Fault mode)
		L2	Yellow LED: • Flashes continuously at start-up. Variable data and registers in the Alpha® sign are cleared at start-up. • Flashes when transmitting data to an Alpha® sign (normal mode) *and/or* • Flashes once every 500 ms to indicate that the heartbeat circuit is functioning.
		PWR	Green LED: • Steady green while the unit has power.
H	DISPLAY connector	Unshielded RS485 output to a single Alpha® sign or display	
I	SERIAL COM connector	Unshielded RS232 input from a PC, to program messages and send them to a CPU Module that is *no farther than 50 feet* from the PC. This is not a telephone connector.	
J	Serial input	Not used.	

Power Supply Module



Item	Name	Description
A	Module top	The internal PCB is attached to the module top.
B	Release buttons	Use a pen tip or other pointed tool to push release buttons and pull off the module bottom.
C	Module bottom	
D	DIN rail latch	Used to attach the module to a DIN mounting rail.
E	Intermodule connector	Used to pass power and signals between modules.
F	Power output	24 VDC
G	Power input	24 VDC or 24 VAC

Network Interface Module



Item	Name	Description	
A	DB9 connector (female)	Serial interface connector.	
B	Addresses DIP switches	DIP switches for node ID and source ID. Refer to “Network Interface Module settings” on page 11.	
C	Status and diagnostic LEDs	ERROR	Active (red) indicates that a transmission error has occurred.
		MBP Init	• Flashes during start-up. • Steady green indicates the Network Interface Module is active.
		MBP Active	Flashing indicates the unit is online and active. The pattern of flashes indicate the state of the Network Interface Module node: • Flashes every 160 ms = Node is working normally. • Flashes every 1 second = Node in MONITOR_OFFLINE state. • 2 flashes, off 2 seconds = Node is in MAC_IDLE never-getting-token state. • 3 flashes, off 1.7 second = Node is not detecting any other nodes. • 4 flashes, off 1.4 second = Node has detected a valid packet that has a duplicated-node-address sent from another node on the network, using the same Node ID.
B	Release buttons	Use a pen tip or other pointed tool to push release buttons in and then pull the module sections apart.	
E	Intermodule connector	Used to pass power and signals between modules.	
F	DIN rail latch	Used to attach the module to a DIN mounting rail.	

Module specifications

The Alpha® Gateway Interface has Modbus Plus certification and is CE-compliant.

	Power Supply	**CPU**	**Network Interface**
Physical data			
Dimensions	2.75"W x 4.25"H x 1"D 70mm W x 108mm H x 25mm D	2.75"W x 4.25"H x 1"D 70mm W x 108mm H x 25mm D	3.54” W x 2.95” H x 4.13” D 90mm W x 75mm H x 105mm D
Weight	4 oz. (113 grams)	4 oz. (113 grams)	6.7 oz. (190 grams)
Operating temperature	140°F (60°C)	140°F (60°C)	41°F (5°C) – 140°F (60°C)
Humidity range	10 – 95% non-condensing		
Mounting	DIN EN 5Ø Ø22 rail 1.34” x .28” (35mm x 7mm)		
Operating specifications			
Operating voltage	24 VAC or 24 VDC	Bus powered by embedded +5VDC supply	

Voltage		Min = 4.75 Volts; Typical = 5.00 Volts;
		Max = 5.25 Volts	

Current draw		Typical = 150 mA	Typical = 350 mA;
Power consumption	Typical = 15W	Typical = 0.75 W	Typical = 1.75 W
AC input voltage	24 VAC		
Max. AC voltage	25 Vrms		
Min. AC voltage	14 Vrms		
DC input voltage	24 VDC		
Max. DC voltage	36 VDC		
Min. DC voltage	18 VDC		
Output voltage	24 VDC		
Max. voltage	36 VDC		
Min. voltage	18 VDC		
Max. current	700 mA total		
Bus output voltage	5 VDC		
Max. voltage	5.05 VDC		
Min. voltage	4.95 VDC		
Max. current	500 mA		
Protection			Per Modbus Plus standards
Type	Poly switch		
Self-resetting	Yes		
Terminals			
Type	Screw	Screw	9-PIN D-sub 2mm female
Wire size	Copper stranded wire: US spec: AWG 26 - 14 Euro spec: Ø,14 – Ø,25 ❏	Copper stranded wire: US spec: AWG 26 - 14 Euro spec: Ø,14 – Ø,25 ❏	RS485 twisted pair cable (1 pair + Shield) per Modbus Plus standards
Communications			
Serial (in)		Communication type: RS232 Terminal type: RJ11 female Protocol: Adaptive® Modbus ASCII/Modified EZ95 Protocol Communication parameters: either of the following: 1) 9600 baud, 8 data bits, no parity, 1 stop bit 2) 9600 baud, 7 data bits, even parity, 2 stop bits	1 M bits/s
Display (out)		Communication type: RS485 Terminal type: RJ11 female Protocol: Adaptive® Modbus ASCII Communication parameters: 9600 baud, 7 data bits, even parity, 2 stop bits	
Terminals (out)		Communication type: RS485 Protocol: Adaptive® Modbus ASCII Max. number of drops: 32 Max. distance: 4000 ft (1200 m) Communication parameters: 9600 baud, 7 data bits, even parity, 2 stop bits	

Network diagrams

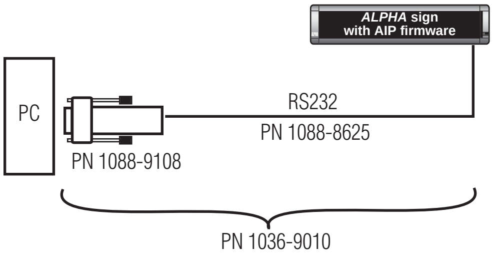
There are several basic diagrams for Modbus Plus networks. Detailed diagrams are in the following section.

For each diagram, additional signs can be networked to the first sign.

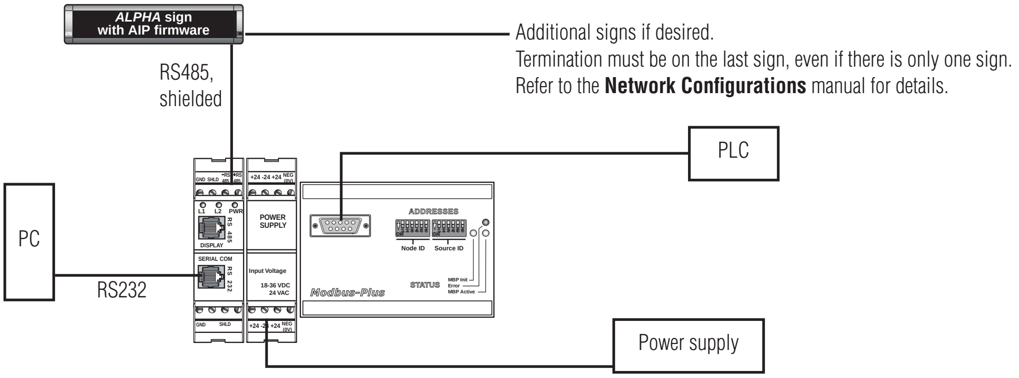
In each scenario, you can switch the Power Supply and CPU Modules with each other, therefore two configurations of the AGI are possible:

- CPU—Power Supply Module—Network Interface Module *or*
- Power Supply Module—CPU—Network Interface Module

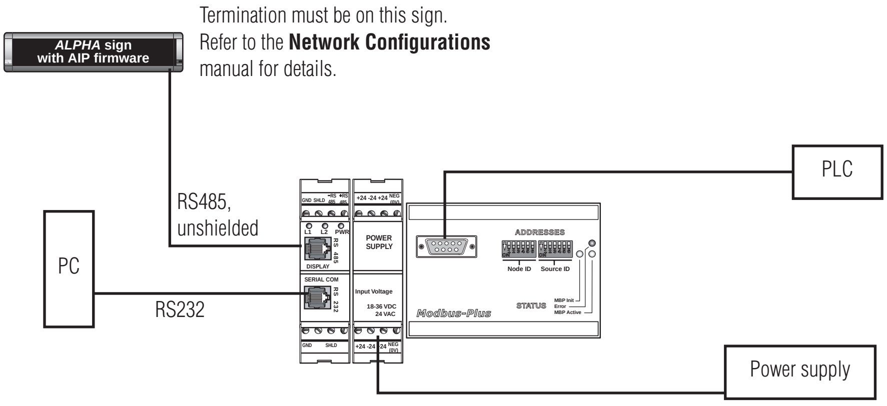
1. Simple setup: to download messages to sign prior to connecting to Modbus Plus network.



2. Recommended setup: for use with signs at any distance from the AGI.
   The AGI is connected to Alpha® signs via a multi-drop (RS485) network. This network will support up to 32 drops before requiring a repeater.

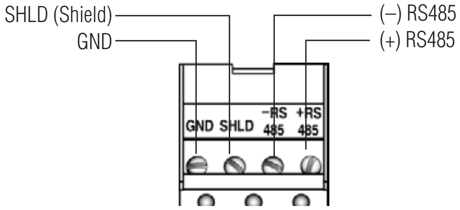
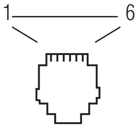
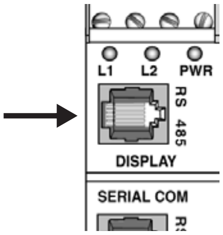
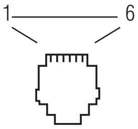
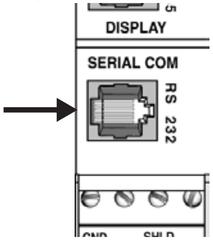


3. Optional setup: for use with any sign at a distance of *50 feet or less* from the AGI.

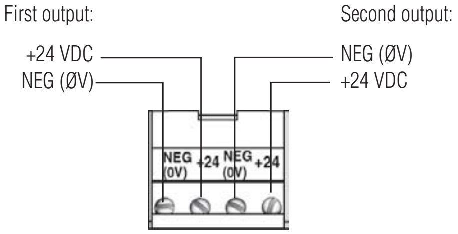
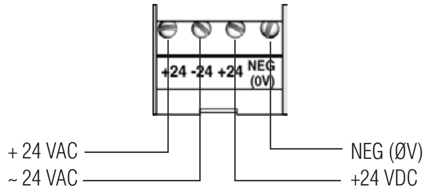


CPU Module wiring connections

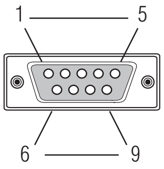
Choose *either* Step 1a (recommended) *or* Step 1b based on your sign network.

<table>
<tr><th>Step</th><th>Instruction</th></tr>
<tr><td>1a</td><td>Connect the serial output to any Alpha® sign(s) at any distance from the Alpha® Gateway Interface.
Use 120 ohm impedance 2-wire shielded RS485 cable, AWG 26 – 14 (Ø.14 – Ø.25 ❑), Adaptive® PN 1088-8002, 7122-0283, or 7122-0284; or Belden PN 9843.
<i>Because of the signal protection afforded by shielding, this is the recommended way of connecting Alpha® signs to the CPU Module.</i>
 </td></tr>
<tr><td>1b</td><td>Connect the DISPLAY connector using this unshielded RS485 output to any sign(s) at a distance of less than 50 feet from the AGI.
Use 4-wire shielded cable, Adaptive® PN 1088-8624.

Pinout (female)
<table><tr><th>Pin</th><th>Signal</th></tr><tr><td>1</td><td>GND</td></tr><tr><td>2</td><td>RS485 (+)</td></tr><tr><td>3</td><td>Not connected</td></tr><tr><td>4</td><td>Not connected</td></tr><tr><td>5</td><td>RS485 (-)</td></tr><tr><td>6</td><td>Not connected</td></tr></table>
NOTE: This is not a telephone connector.</td></tr>
<tr><td>2</td><td>Connect the SERIAL COM connector.
Use this unshielded RS232 input from a PC to program messages and send them to a CPU Module that is at a distance of less than 50 feet from the PC.
Use 6-conductor RS232 cable, such as Adaptive® PN 1088-8625 (25 feet) or 1088-8627 (50 feet).

Pinout (female)
<table><tr><th>Pin</th><th>Signal</th></tr><tr><td>1</td><td>GND</td></tr><tr><td>2</td><td>Not connected</td></tr><tr><td>3</td><td>RxD</td></tr><tr><td>4</td><td>TxD</td></tr><tr><td>5</td><td>Not connected</td></tr><tr><td>6</td><td>Not connected</td></tr></table>
NOTE: This is not a telephone connector.</td></tr>
</table>

Power Supply Module wiring connections

Step	Instruction
1	(Optional) Connect the power supply output if you need to power other devices. Note wire specifications: AWG 26 – 14 (Ø.14 – Ø.25 ❏)  First output: +24 VDC NEG (ØV) Second output: NEG (ØV) +24 VDC NEG (0V) +24 NEG (0V) +24
2	Connect the power input: 24 VDC or 24 VAC. Note wire specifications: AWG 26 – 14 (Ø.14 – Ø.25 ❏)  +24 -24 +24 NEG (0V) + 24 VAC ~ 24 VAC NEG (ØV) +24 VDC

Network Interface Module wiring connections

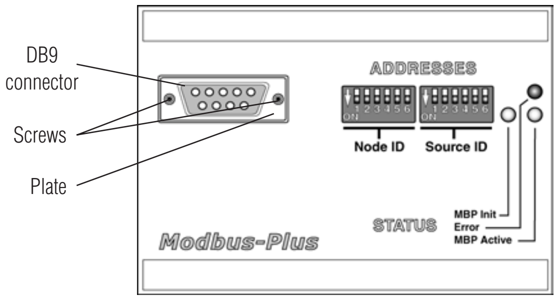
<table>
<tr><th>Step</th><th>Instruction</th></tr>
<tr><td>2</td><td>Connect the DB9 serial interface.


Pinout (female)
<table>
<tr><th>Pin</th><th>Name</th></tr>
<tr><td>Housing</td><td>Protective Earth Ground</td></tr>
<tr><td>Pin 1</td><td>Cable Shielding</td></tr>
<tr><td>Pin 2</td><td>Modbus Plus Line B</td></tr>
<tr><td>Pin 3</td><td>Modbus Plus Line A</td></tr>
<tr><td>Pin 4</td><td>No connection</td></tr>
<tr><td>Pin 5</td><td></td></tr>
<tr><td>Pin 6</td><td></td></tr>
<tr><td>Pin 7</td><td></td></tr>
<tr><td>Pin 8</td><td></td></tr>
<tr><td>Pin 9</td><td></td></tr>
</table>
</td></tr>
</table>

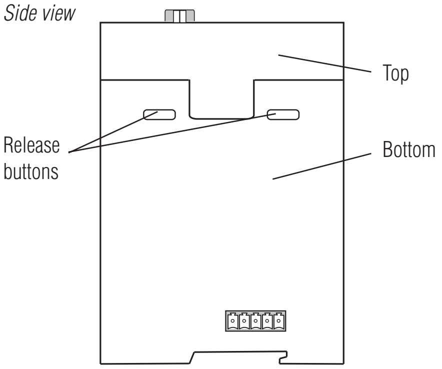
Network Interface Module settings

<table>
<tr><th>Step</th><th>Instruction</th></tr>
<tr><td>1</td><td>Open the Network Interface Module:
1. Take appropriate measures to avoid static electricity damage to the internal circuit boards.
2. Unscrew the DB9 serial interface connector.
3. Remove the separate plate for the DB9 connector on the top of the module.
4. Using the tip of a pen or similar tool, press the release buttons on both sides of the module in and up to release the catches.
5. Lift the top section of the module away from the bottom section of the module.

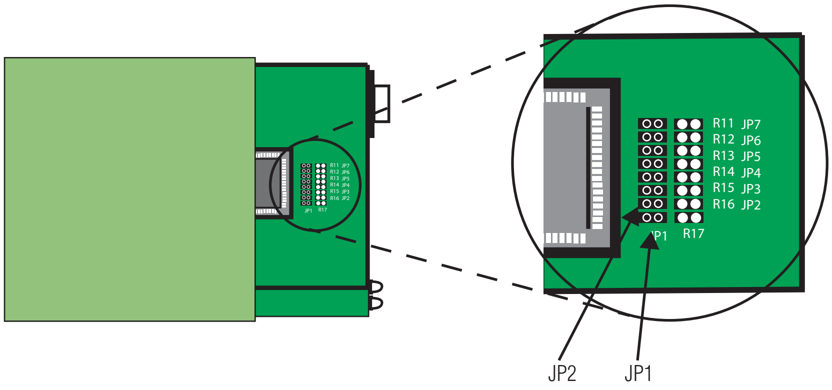
Top view



Side view



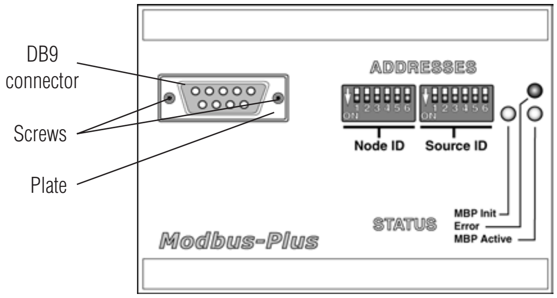
</td></tr>
<tr><td>2</td><td>Set the jumpers inside the Network Interface Module to determine the communication profile.



Jumper settings for Gateway firmware

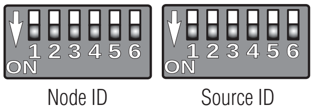
<table>
<tr><th>Communication method</th><th>Jumper settings</th><th>Motes</th></tr>
<tr><td>Global</td><td>JP1 = On
JP2 = Off</td><td>The AGI receives 32 words from the PLC.
The AGI sends 16 words back to the PLC.</td></tr>
<tr><td>Peer-to-Peer</td><td>JP1 = On
JP2 = On</td><td></td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Step</th><th>Instruction</th></tr>
<tr><td>3</td><td>
Close the Network Interface Module:
1. Replace the top of the module in the bottom.
2. Press the top and bottom sections of the module together so the release buttons lock.
3. Replace the separate plate for the DB9 connector on the top of the module.
4. Insert and tighten the screws for the DB9 connector.

Top view


</td></tr>
<tr><td>4</td><td>
Set the DIP switches on the top of the module for node ID and source ID.
• Node ID is the node address of the AGI.
• Source ID is the node address of the PLC that is communicating with the AGI.

NOTE: Node ID and source ID should be set *before* power is applied to the unit.
Any change of node ID or source ID is not valid until the next power cycle.

Each of these addresses is set in binary format:


<table>
<tr><th>Address</th><th>MSB
1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>LSB
6</th></tr>
<tr><td>1</td><td>ON</td><td>ON</td><td>ON</td><td>ON</td><td>ON</td><td>ON</td></tr>
<tr><td>2</td><td>ON</td><td>ON</td><td>ON</td><td>ON</td><td>ON</td><td>OFF</td></tr>
<tr><td>3</td><td>ON</td><td>ON</td><td>ON</td><td>ON</td><td>OFF</td><td>ON</td></tr>
<tr><td colspan="7">. . .</td></tr>
<tr><td>64</td><td>OFF</td><td>OFF</td><td>OFF</td><td>OFF</td><td>OFF</td><td>OFF</td></tr>
<tr><td colspan="7">See “Dip switch settings” on page 31
for additional address settings.</td></tr>
</table>
</td></tr>
</table>

PLC programming examples

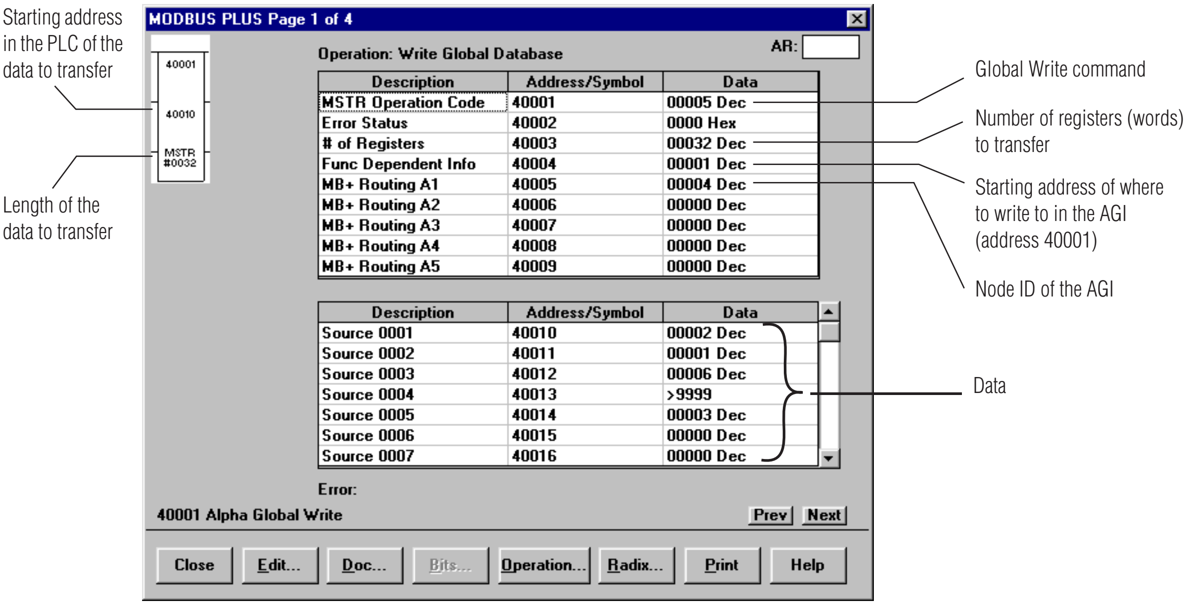
These examples illustrate the steps to program a PLC using two programming methods with each of the communication techniques.

	MSTR block	Register mapping
Global	Example 1	Example 3
Peer-to-Peer	Example 2	n/a

MSTR block – Global

This example uses ProWORX NxT software on a Modicon PLC.

1. These are the settings for a Write to the Global database:



MODBUS PLUS Page 1 of 4

Operation: Write Global Database

AR:

Description	Address/Symbol	Data
MSTR Operation Code	40001	00005 Dec
Error Status	40002	0000 Hex
# of Registers	40003	00032 Dec
Func Dependent Info	40004	00001 Dec
MB+ Routing A1	40005	00004 Dec
MB+ Routing A2	40006	00000 Dec
MB+ Routing A3	40007	00000 Dec
MB+ Routing A4	40008	00000 Dec
MB+ Routing A5	40009	00000 Dec

Description	Address/Symbol	Data
Source 0001	40010	00002 Dec
Source 0002	40011	00001 Dec
Source 0003	40012	00006 Dec
Source 0004	40013	>9999
Source 0005	40014	00003 Dec
Source 0006	40015	00000 Dec
Source 0007	40016	00000 Dec

Error:

40001 Alpha Global Write

Starting address in the PLC of the data to transfer

Length of the data to transfer

Global Write command

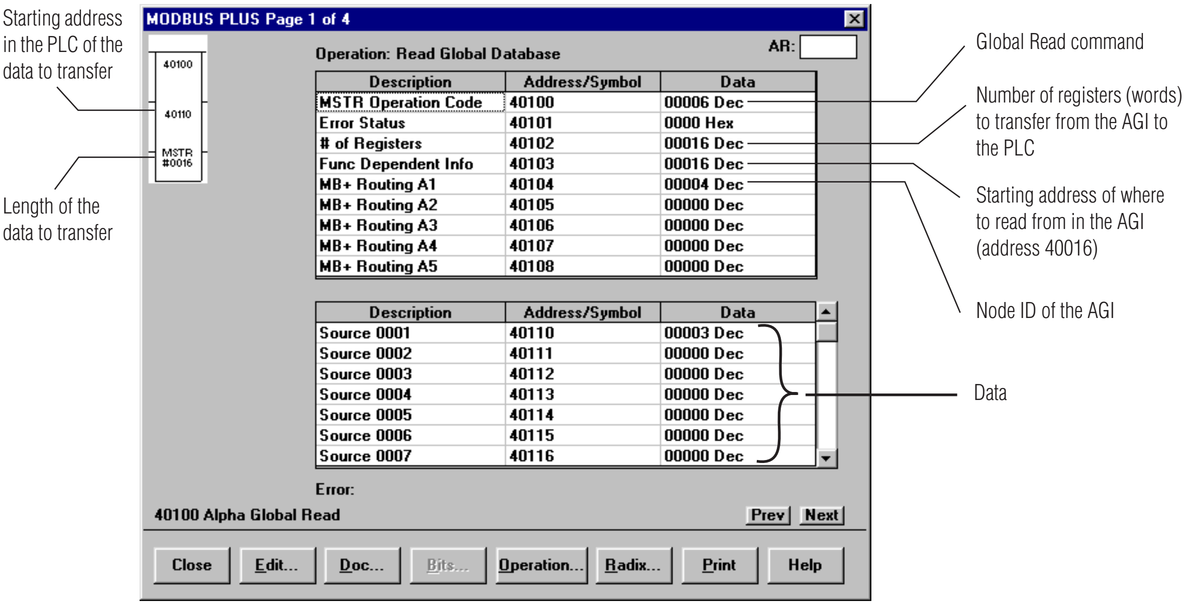
Number of registers (words) to transfer

Starting address of where to write to in the AGI (address 40001)

Node ID of the AGI

Data

2. These are the settings for a Read from the Global database:



MODBUS PLUS Page 1 of 4

Operation: Read Global Database

AR:

Description	Address/Symbol	Data
MSTR Operation Code	40100	00006 Dec
Error Status	40101	0000 Hex
# of Registers	40102	00016 Dec
Func Dependent Info	40103	00016 Dec
MB+ Routing A1	40104	00004 Dec
MB+ Routing A2	40105	00000 Dec
MB+ Routing A3	40106	00000 Dec
MB+ Routing A4	40107	00000 Dec
MB+ Routing A5	40108	00000 Dec

Description	Address/Symbol	Data
Source 0001	40110	00003 Dec
Source 0002	40111	00000 Dec
Source 0003	40112	00000 Dec
Source 0004	40113	00000 Dec
Source 0005	40114	00000 Dec
Source 0006	40115	00000 Dec
Source 0007	40116	00000 Dec

Error:

40100 Alpha Global Read

Starting address in the PLC of the data to transfer

Length of the data to transfer

Global Read command

Number of registers (words) to transfer from the AGI to the PLC

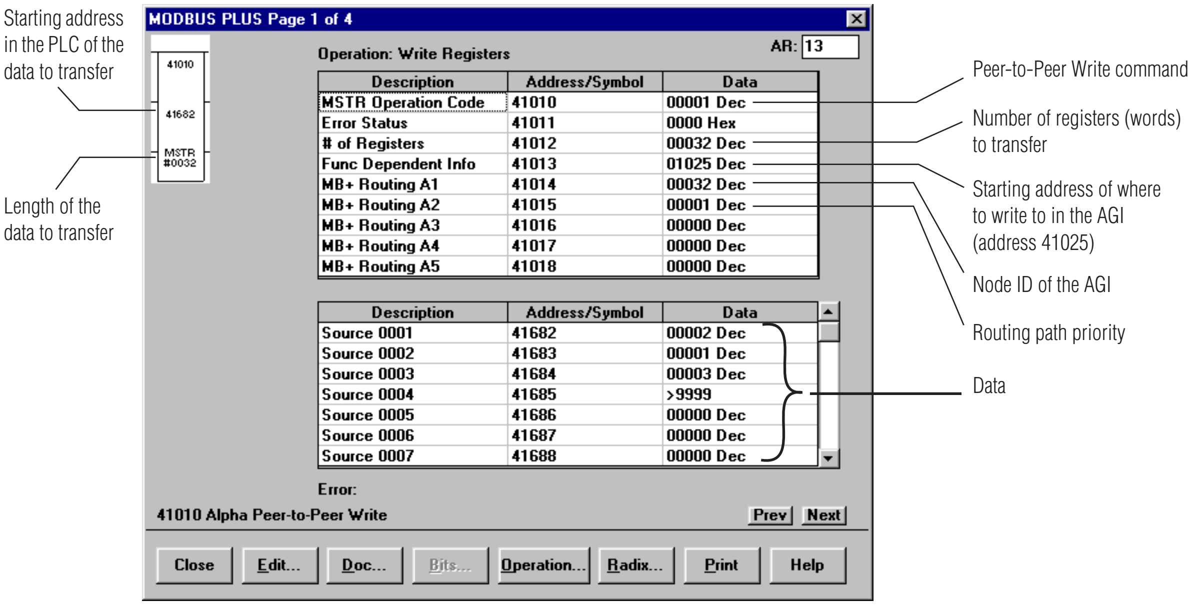
Starting address of where to read from in the AGI (address 40016)

Node ID of the AGI

Data

This example uses ProWORX NxT software on a Modicon PLC.

1. These are the settings for a Write to a Peer-to-Peer database:



Starting address in the PLC of the data to transfer

Length of the data to transfer

Operation: Write Registers

AR: 13

Description	Address/Symbol	Data
MSTR Operation Code	41010	00001 Dec
Error Status	41011	0000 Hex
# of Registers	41012	00032 Dec
Func Dependent Info	41013	01025 Dec
MB+ Routing A1	41014	00032 Dec
MB+ Routing A2	41015	00001 Dec
MB+ Routing A3	41016	00000 Dec
MB+ Routing A4	41017	00000 Dec
MB+ Routing A5	41018	00000 Dec

Description	Address/Symbol	Data
Source 0001	41682	00002 Dec
Source 0002	41683	00001 Dec
Source 0003	41684	00003 Dec
Source 0004	41685	>9999
Source 0005	41686	00000 Dec
Source 0006	41687	00000 Dec
Source 0007	41688	00000 Dec

Error:

41010 Alpha Peer-to-Peer Write

Peer-to-Peer Write command

Number of registers (words) to transfer

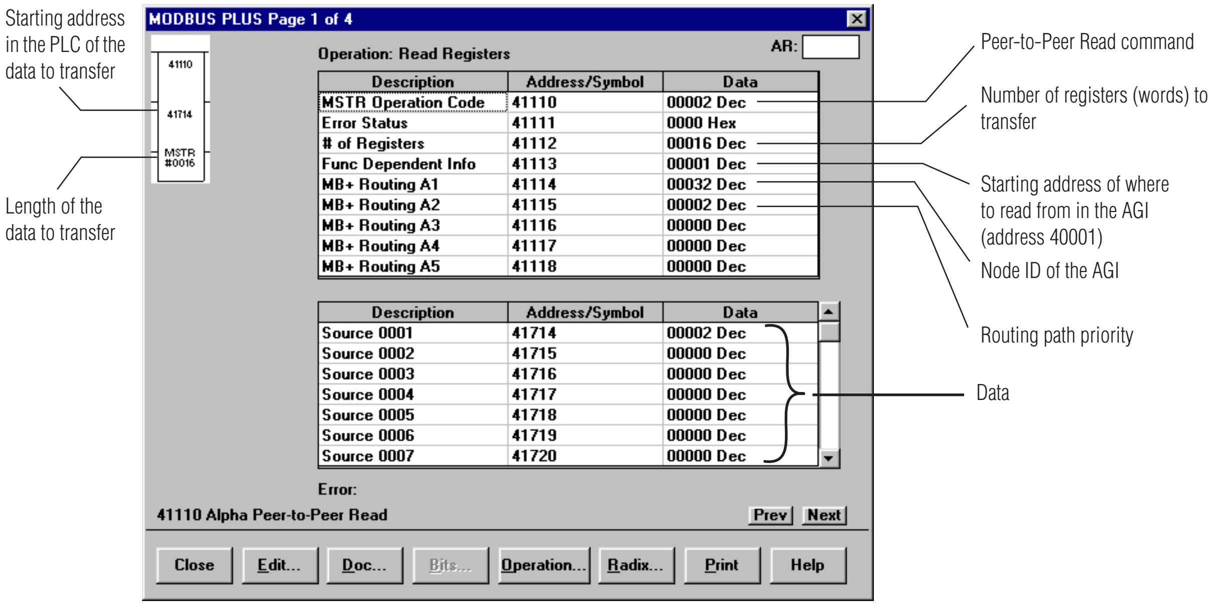
Starting address of where to write to in the AGI (address 41025)

Node ID of the AGI

Routing path priority

Data

2. These are the settings for a Read from a Peer-to-Peer database:



Starting address in the PLC of the data to transfer

Length of the data to transfer

Operation: Read Registers

AR:

Description	Address/Symbol	Data
MSTR Operation Code	41110	00002 Dec
Error Status	41111	0000 Hex
# of Registers	41112	00016 Dec
Func Dependent Info	41113	00001 Dec
MB+ Routing A1	41114	00032 Dec
MB+ Routing A2	41115	00002 Dec
MB+ Routing A3	41116	00000 Dec
MB+ Routing A4	41117	00000 Dec
MB+ Routing A5	41118	00000 Dec

Description	Address/Symbol	Data
Source 0001	41714	00002 Dec
Source 0002	41715	00000 Dec
Source 0003	41716	00000 Dec
Source 0004	41717	00000 Dec
Source 0005	41718	00000 Dec
Source 0006	41719	00000 Dec
Source 0007	41720	00000 Dec

Error:

41110 Alpha Peer-to-Peer Read

Peer-to-Peer Read command

Number of registers (words) to transfer

Starting address of where to read from in the AGI (address 40001)

Node ID of the AGI

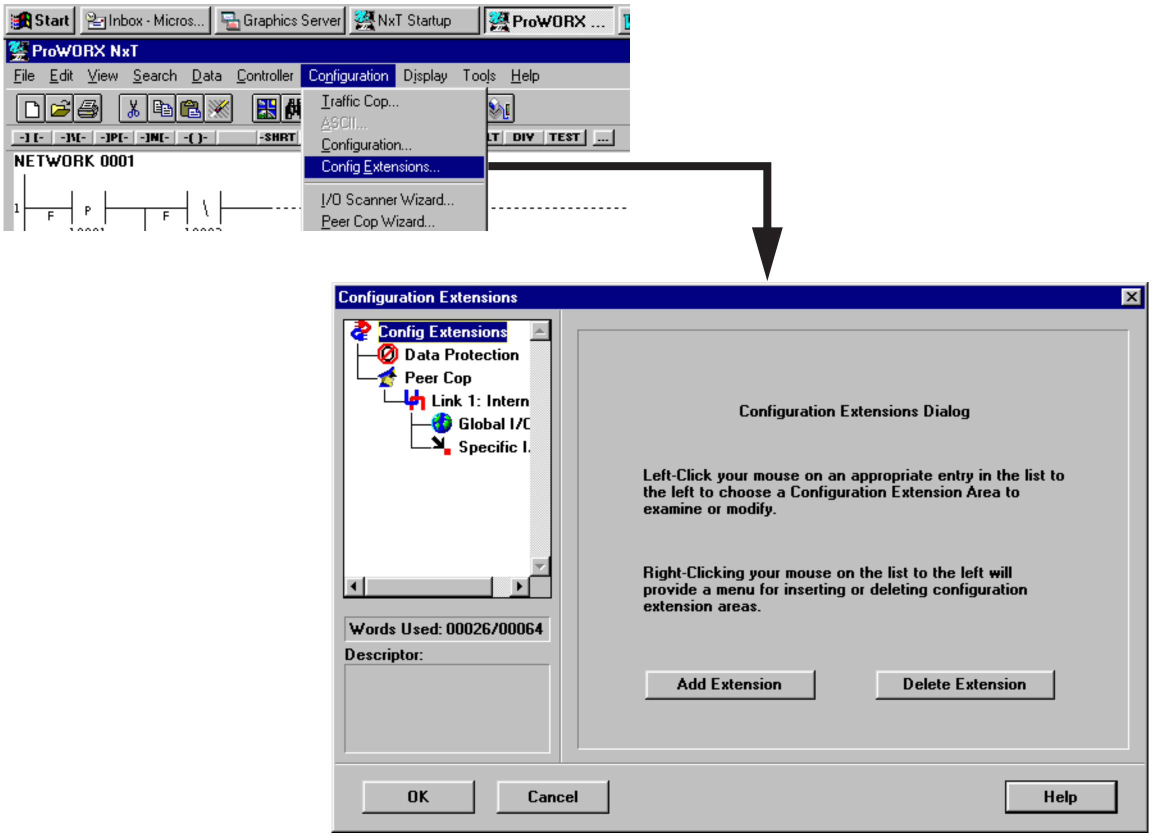
Routing path priority

Data

Register mapping – Global

This example uses ProWORX NxT software on a Modicon PLC to map data between the PLC and the AGI using configuration extensions.

1. Add an Extension:



Words Used: 00026/00064

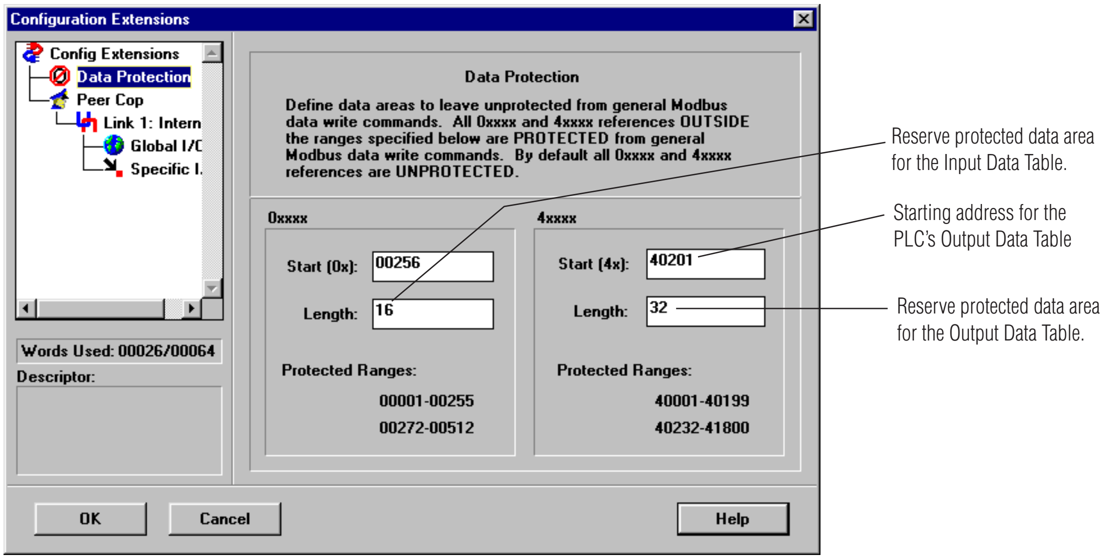
Descriptor:

Configuration Extensions Dialog

Left-Click your mouse on an appropriate entry in the list to the left to choose a Configuration Extension Area to examine or modify.

Right-Clicking your mouse on the list to the left will provide a menu for inserting or deleting configuration extension areas.

2. Define data areas for Global information:



Words Used: 00026/00064

Descriptor:

Data Protection

Define data areas to leave unprotected from general Modbus data write commands. All 0xxxx and 4xxxx references OUTSIDE the ranges specified below are PROTECTED from general Modbus data write commands. By default all 0xxxx and 4xxxx references are UNPROTECTED.

0xxxx		4xxxx	
Start (0x):	00256	Start (4x):	40201
Length:	16	Length:	32
Protected Ranges:		Protected Ranges:	
	00001-00255		40001-40199
	00272-00512		40232-41800

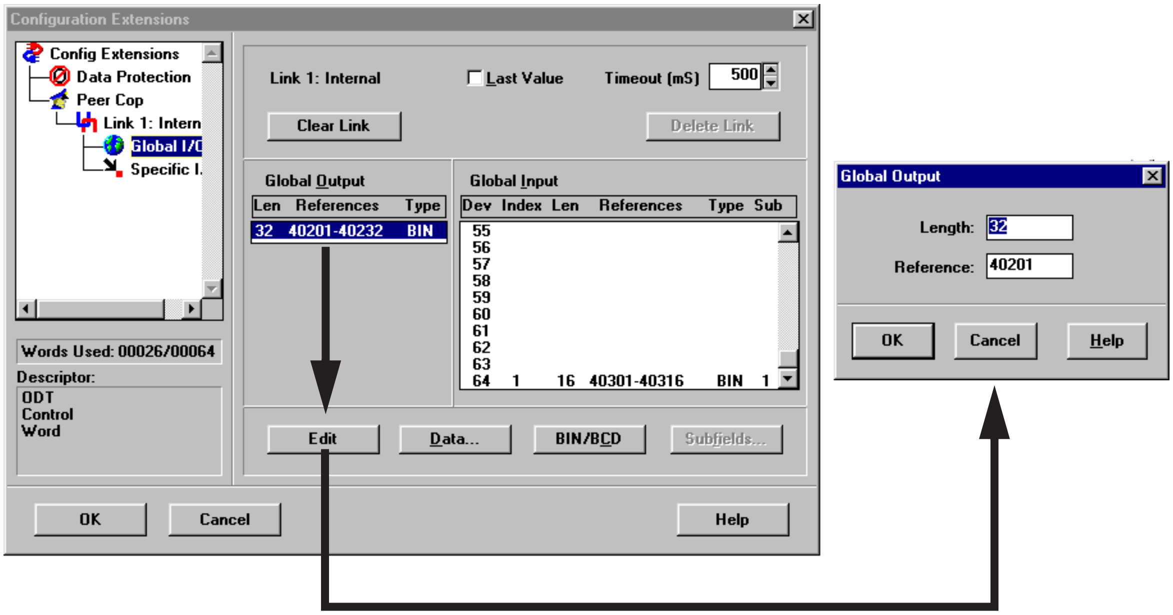
Reserve protected data area for the Input Data Table.

Starting address for the PLC's Output Data Table

Reserve protected data area for the Output Data Table.

NOTE: When using register mapping for Global communication, you do not have to use all 32 output or 16 input words. Registers not used will be ignored.

3. Map the PLC output data table to the AGI:



Link 1: Internal □ Last Value Timeout (mS) 500

Global Output

Len	References	Type
32	40201-40232	BIN

Global Input

Dev	Index	Len	References	Type	Sub
55					
56					
57					
58					
59					
60					
61					
62					
63					
64	1	16	40301-40316	BIN	1

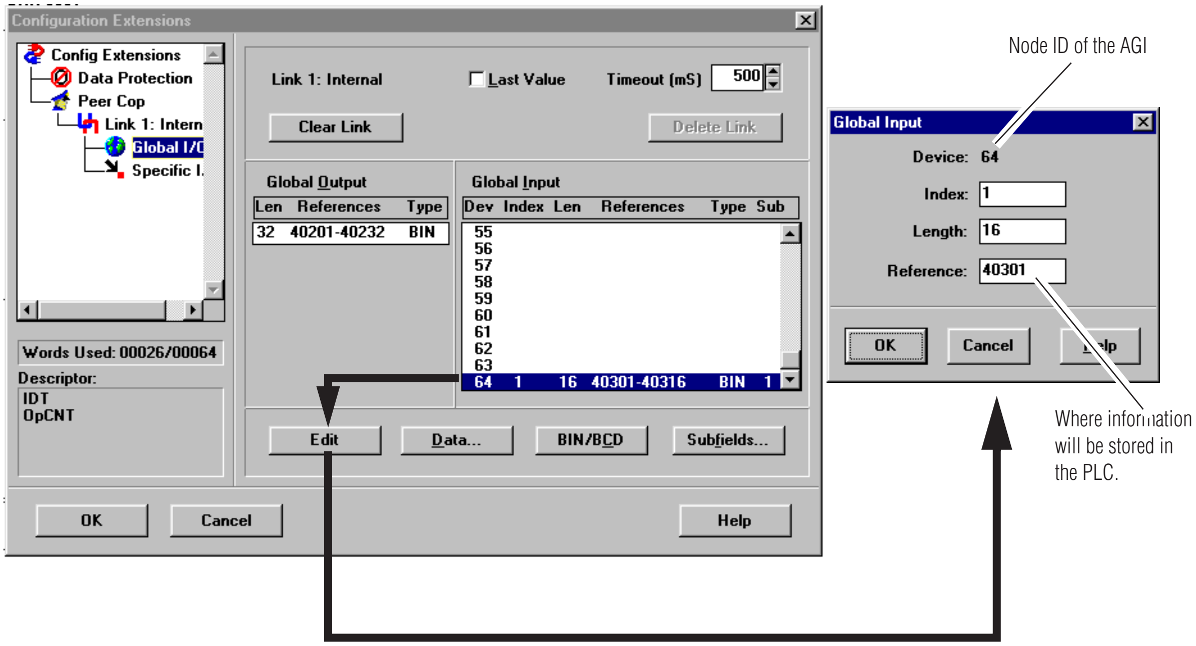
Words Used: 00026/00064

Descriptor:
ODT
Control
Word

Length: 32

Reference: 40201

4. Map the AGI input data table to the PLC:



Link 1: Internal □ Last Value Timeout (mS) 500

Global Output

Len	References	Type
32	40201-40232	BIN

Global Input

Dev	Index	Len	References	Type	Sub
55					
56					
57					
58					
59					
60					
61					
62					
63					
64	1	16	40301-40316	BIN	1

Words Used: 00026/00064

Descriptor:
IDT
OpCNT

Device: 64

Index: 1

Length: 16

Reference: 40301

Node ID of the AGI

Where information will be stored in the PLC.

Safety and troubleshooting

When successfully connected to a Modbus Plus industrial network, there should *always* be some type of message on each Alpha® sign connected to this network *unless the default message length has been changed*.

General

Problem	Possible causes
No message appears on Alpha® sign.	• Network wiring fault. • PLC fault. • Alpha® sign fault possible sign hardware failure or a PLC is trying to display a message that was not programmed into the sign. • Message(s) too long for preset file size. • Not switched on/plugged in. • The only character programmed into the message is a “space”.
“No Network Activity” message appears on Alpha® sign.	• Network wiring fault. • PLC fault. • Alpha® sign fault. • Alpha® sign timeout– no network activity for at least 3 seconds. • Gateway offline/configured for RS232 data.
“NO BACKGROUND MESSAGE”[1] appears on Alpha® sign.	• Sign address is not correct. • The sign has not received any message to display. (This is not an error condition.) • Sign is receiving information, but the information is not for this sign or is invalid. • Sign has not received any valid serial data.
Message Error: Specific message number is displayed, for example “Message # 0024”	• “Blank” message: Either This message was never edited (and never downloaded to the display), or Messages that are invalid (with *Alpha® Automation Software* or *Gateway Messaging Software* syntax errors) never make it to the display; they can not be downloaded with invalid content. • Message length exceeds number of bytes the sign has been configured to accept. Messages that are too long will not display properly.
[1] Wording of the “background message” can be changed in the Alpha® Automation Software or Gateway Messaging Software.	

CPU Module

Problem	Possible causes and corrective actions
LEDs do not light.	• Check power connections. • Check line fuse.
Red LED (L1) flashes.	• Check to see that the Network Interface module is correctly seated with the CPU Module or the Power Supply Module. • Check the jumper settings for JP1 and JP2. • Consult the factory.
Yellow LED (L2) does not flash after the power-up cycle.	• Check the jumper settings for JP1 and JP2. • Consult the factory.
LEDs on the Network Interface Module do not light up but those on the CPU Module do.	• Check to see that the 5 pins on the main circuit board ar properly seated. • Check to see that the Network Interface module is correctly seated with the CPU Module or the Power Supply Module.
[1] Wording of the “background message” can be changed in the Alpha® Automation Software or Gateway Messaging Software.	

Network Interface Module

Problem	Possible causes and corrective actions
LEDs do not light.	• Check power connections. • Check line fuse.
Error LED is steady red.	• Check the Network Interface Module cabling. • Check the number of registers used for Data tables.
MBP Init LED flashed continuously at power-up or is steady on.	• Check the power supply voltage. • Check the jumper settings for JP1 and JP2. • Possible fault with Modbus Plus printed circuit board. Consult the factory.
MBP Active light is flashing.	The pattern of flashes indicate the state of the Network Interface Module node: • 1 flash every 160 ms = Node is working normally. • 1 flash every 1 second = Node in MONITOR_OFFLINE state. • 2 flashes, then off 2 seconds = Node is in MAC_IDLE state, not receiving a token as it should. • 3 flashes, then off 1.7 second = Node is not detecting any other nodes. • 4 flashes, then off 1.4 second = Node has detected a valid packet that has a duplicated-node-address sent from another node on the network, using the same Node ID.

[1] Wording of the “background message” can be changed in the Alpha® Automation Software or Gateway Messaging Software.

Configuration data for Alpha® Gateway networks

Overview

The Alpha® Gateway Interface maps Output and Input Data tables in the PLC to the Gateway interface's Input and Output Data tables. Once mapped, the PLC, operating through the Alpha® Gateway Interface (the "AGI"), controls the function and operations of the Alpha® sign network.

The Gateway network is configured to allow two different types of messaging, "Add/remove messages" and "Priority messaging".

- "Add/remove messages" allows several different messages to all be active at once. The messaging control function allows different priority levels to be applied to the messages that are programmed with *Alpha® Automation Software* (or *Gateway Messaging Software*.)
- "Priority messaging" activates only one message at a time, leaving full control of messaging display with the PLC.

Variable data corresponding to values in the Output Data Table and in the *Alpha® Automation Software* (or *Gateway Messaging Software*) may be included in any message.

NOTE: All Alpha® signs used in a Gateway network must have Alpha Industrial Protocol firmware.

Information flow

In an Alpha® sign network, a device is any point in the information pathway that is capable of sending and receiving a data signal.

In the most basic network configuration (one input, one PLC, one AGI, and one sign), the PLC, AGI, and sign are all capable of both sending and receiving data.

See the illustration in "Introduction" on page 1 for a description of information flow.

Functional capacity

Data mapping between the PLC and AGI allows the industrial network to control the following operations of an Alpha® sign network:

- Priority messaging
- Add/remove messages in queue
- Read currently-running messages in a sign
- Read variable data
- Update variable data
- Monitor the status of signs on the network, individually or all at once
- Determine if the AGI is processing data
- Clear any status bits set by the AGI
- Clear all messages currently being displayed

Alpha® sign network data storage

An Alpha® Gateway Interface is connected to the Alpha® signs via a multi-drop RS485 network. This network will support up to 32 drops before requiring a repeater. These signs can be addressed from 001 to 254 by using *Alpha® Automation Software* (see the *Alpha® Automation Software* online Help system) or through each sign's internal DIP switches.

Network device GSD and EDS files

The AGI operating on an industrial network may need to be identified to the network using an initialization file. For example, Profibus DP networks use *.gsd files. DeviceNet networks use *.eds files. Check your network documentation.

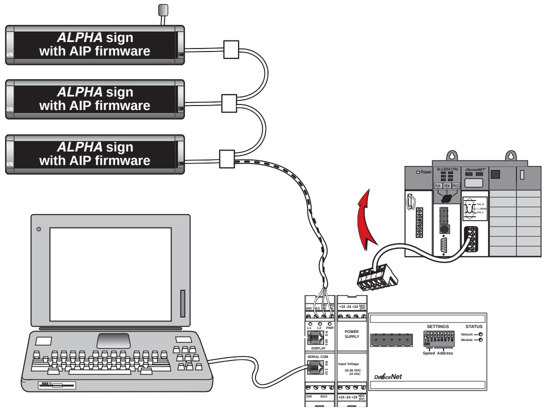
Look for GSD and EDS files in the same directory path used to install *Alpha® Automation Software* (or *Gateway Messaging Software*.) When loading the GSD or EDS files to your system, set the input and output parameters as follows:

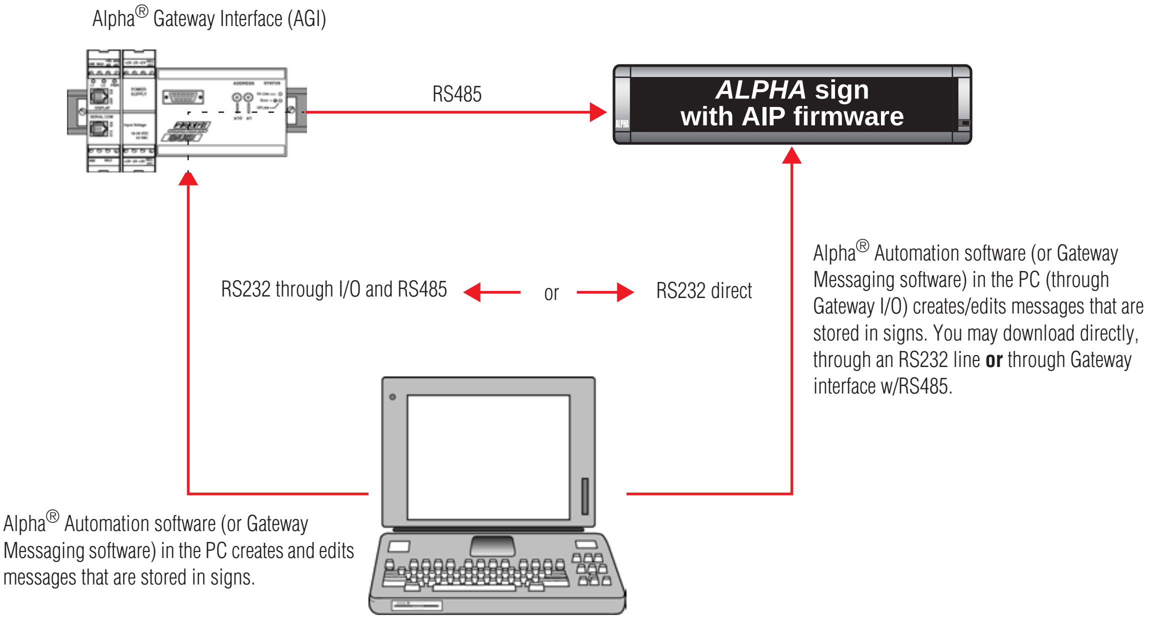
- 32 Words (64 Bytes) out for the PLC Output Data table from the PLC to the Gateway
- 16 Words (32 Bytes) in for the PLC Input Data table from the Gateway to the PLC

Downloading messages to an Alpha® sign network

To download messages from the *Alpha® Automation Software* (or *Gateway Messaging Software*) to the Alpha® sign network, *physically disconnect the Network Interface Module from the industrial network*. Reconnect the network input to the Network Interface Module only after you have completed all downloads.

Also, you could bypass the Network Interface Module and physically connect your PC to the Alpha® sign network directly via an RS232 connection:


ALPHA sign with AIP firmware
ALPHA sign with AIP firmware
ALPHA sign with AIP firmware
DeviceNet


Alpha® Gateway Interface (AGI)
RS485
ALPHA sign with AIP firmware
RS232 through I/O and RS485
or
RS232 direct

Alpha® Automation software (or Gateway Messaging software) in the PC (through Gateway I/O) creates/edits messages that are stored in signs. You may download directly, through an RS232 line **or** through Gateway interface w/RS485.

Alpha® Automation software (or Gateway Messaging software) in the PC creates and edits messages that are stored in signs.

Output Data table

The Output Data table mapped in the PLC to the AGI Input Data table is used to control and execute all available operations of the message sign network.

The Output Data table consists of 32 words, reserved for the functions described in the table below. Valid values are listed for each word. (Invalid values are shown in Table 9 on page 28.)

Table 1: Output Data table

Word	Description	Valid values (decimal)
0	Control Word	2 = Write 3 = Read Message Queue 4 = Read Variable Data -1 = Clear Status bits
1	Sign Address (SA)	1 – 254, 255 is broadcast address.
2	Add Message Number to queue (AM) (When the Control Word is a 2)	1 – 4000 and 4095 are valid message numbers; 4001–4094 are *not* valid. (4095 identifies the background message.)
	Clearing a single bit in Words 1 and 2 of the Input Data table (CB) (When the Control Word is a -1)	0 = Used for clearing a Status bit for an individual sign address
	Starting Register Index (RI) (When the Control Word is a 3)	1-6 when reading Message queue data
	Starting Register Index (RI) (When the Control Word is a 4)	1-9 when reading Variable data.
3	Remove Message Number from queue (When the Control Word is a 2)	0 = Used for adding multiple messages to the queue 1 – 4000 and 4095 valid Message Numbers: 4001–4094 *not* valid. 4095 always used to identify the background message -1 = Used for priority messaging and clearing the message queue
4	Variable data #0	0 – 65535
5	Variable data #1	0 – 65535
6	Variable data #2	0 – 65535
7	Variable data #3	0 – 65535
8	Variable data #4	0 – 65535
9	Variable data #5	0 – 65535
10	Variable data #6	0 – 65535
11	Variable data #7	0 – 65535
12	Variable data #8	0 – 65535
13	Variable data #9	0 – 65535
14	Variable data #10	0 – 65535
15	Variable data #11	0 – 65535
16	Variable data #12	0 – 65535
17	Variable data #13	0 – 65535
18	Variable data #14	0 – 65535
19	Variable data #15	0 – 65535
20	User defined Variable ID #1	16 – 99
21	Variable data	0 – 65535
22	User defined Variable ID #2	16 – 99
23	Variable data	0 – 65535
24	User defined Variable ID #3	16 – 99
25	Variable data	0 – 65535
26	User defined Variable ID #4	16 – 99
27	Variable data	0 – 65535
28	User defined Variable ID #5	16 – 99
29	Variable data	0 – 65535
30	User defined Variable ID #6	16 – 99
31	Variable data	0 – 65535

Displaying messages on Alpha® signs

There are three different ways in which messages can be displayed or managed on an Alpha® sign:

- **Add/remove messages —** Allows more than one message to be displayed at a time, with priority levels set for each message. This allows the ability to manage display sequences and timing.
- **Priority messaging —** Used to display one message at a time.
- **Clearing the Message Queue —** Removes messages from the message queue, to display the background message.

Table 2: Messaging methods

Method	Description	Examples	Benefit	Valid values
Add/remove messages	Allows multiple messages to run concurrently on the same sign(s). When this method is used it also allows the use of priority levels to control the timing and display sequence for all the messages in the queue.	• Table 3 on page 23 • Table 4 on page 23	Allows some scheduling of messaging display during initial creation of the message.	• Word 0 = 2 (only valid) • Word 1 = 1-255d* • Word 2** = 1-4000, 4095d • Word 3** = 1-4000, 4095d
Priority messaging	Allows one message number to be used to show a single message while automatically removing the single message that was running previously. This frees you from having to remove the message that was running before you display a new message	• Table 5 on page 24.	This method allows the PLC to have full control of message display.	• Word 0 = 2 • Word 1 = 1-255d* • Word 2** = 1-4000, 4095d • Word 3** = 0xFFFF(-1d)
Clearing the Message Queue	Allows you to clear all currently running messages on a sign, by storing 0 in word 2 and –1d (0xFFFF) in word 3, as shown in Table 2 on page 22. When this is done, the background message that was programmed using the Alpha® Automation software (or Gateway Messaging software) is displayed. (This is the default background message “NO BACKGROUND MESSAGE” if no other has been programmed.)	• Table 12 on page 29.	Clears all message queue data.	• Word 0 = 2 • Word 1 = 1-255d* • Word 2** = 0 (only valid) • Word 3** = 0xFFFF(-1d)

* When the sign address is 255 decimal/0xFFFF hexadecimal, this is a broadcast message. There is NO reply from sign(s) on the network.
** The value of Words 2 and 3 can not be the same. The Gateway will not process the information. See Output Data Table Integrity Bit Table.

Add/remove messages example 1

This is an example of the steps required to add three different messages, one at a time, on the same Alpha® sign with *Add/remove messages* method.

In this example, we want to display messages 40, 42, and 500 individually on the same sign. The sign address is 6 (decimal).

Using information found in Table 1 on page 21, store the desired message numbers and sign information into words 0-3 of the Output Data table.

Table 3: Add/Remove messages: Example 1

Word	Step 1	Step 2	Step 3	Step 4
0	2	2	2	2
1	6	6	6	6
2	0	40	42	500
3	-1	0	40	42
Result	Message queue is cleared and the background message is displayed.	Message 0 is removed and Message 40 is added to the queue.	Message 40 is removed and Message 42 is added to the queue.	Message 42 is removed and Message 500 is added to the queue.
NOTE: All values for words 0 through 3 are shown in decimal.				

Add/remove messages example 2

This example shows how to add three different messages to the queue at the same time on an Alpha® sign. This requires that each message be added without removing the previous message.

In this example, we want to display messages 84, 589 and 34 at the same time on a sign. The sign address is 55 (decimal).

Using information found in Table 1 on page 21, store the desired message numbers and sign information into words 0-3 of the Output Data table.

Table 4: Add/Remove messages: Example 2

Word	Step 1	Step 2	Step 3	Step 4
0	2	2	2	2
1	55	55	55	55
2	0	84	589	34
3	-1	0	0	0
Result	Message queue is cleared and the background message is displayed.	Message 0 is removed and message 84 is displayed.	Message 0 is removed and messages 84 and 589 are displayed.	Message 42 is removed and messages 84, 589, and 34 are displayed.
NOTE: All values for words 0 through 3 are shown in decimal.				

NOTE: Messages can be removed individually (see steps 3 and 4 in Table 3) or all at the same time (step 1 in Table 5).

Priority messaging example

This example shows how to display individual messages using *Priority messaging*, and the steps required to add three different messages on an Alpha® sign with *Priority messaging*. This means that only the number of the message to be displayed is stored in word 2.

In this example, we want to display messages 954, 26, and 50 individually on the same sign. The sign address is 3 (decimal). Using information found in Table 1, store the desired message numbers and sign information into words 0-3 of the Output Data table.

Table 5: Priority Message example

Word	Step 1	Step 2	Step 3	Step 4
0	2	2	2	2
1	3	3	3	3
2	0	954	26	50
3	-1	-1	-1	-1
Result	Message Queue is Cleared and the background message is displayed.	Message 954 is displayed.	Message 26 is displayed.	Message 50 is displayed.
NOTE: All values shown in decimal.				

Displaying variable data on Alpha® signs

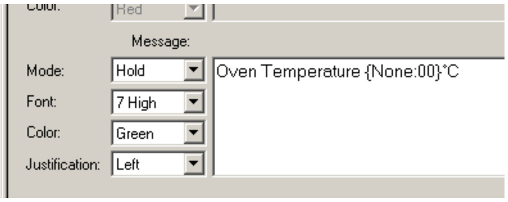
An Alpha® sign has the capability to store up to 100 variables in its memory. These variables are stored in a sign's memory register (001 to 100) corresponding to the variable number (0 to 99) used in the Alpha® Automation software (or Gateway Messaging software.)

There are 16 words in the Output Data table that are reserved for variable numbers 0-15. Words 20-31 are used to allow access to variable numbers 16-99. Variable data stored in words 4-31 are sent to all signs by using defined variable ID numbers and variable data in the next word following (see Output Data Table, odd-numbered variable data in words 20-31 must correspond to even-numbered variable ID numbers in the preceding word), once the Gateway receives the change of state for a particular variable.

The two examples that follow show display of messages containing embedded variable data.

Displaying variable data example 1

In this first example, message 5 contains variable 0, used for a processor's oven temperature. Message 5 is created in Alpha® Automation software (or Gateway Messaging software) in this format:



Color: Red

Message:

Mode: Hold

Font: 7 High

Color: Green

Justification: Left

Oven Temperature {None:00}°C

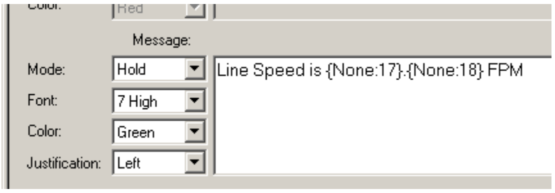
We want to display this on the sign with address 20 using the *Add/remove messages* method. This only requires that the message be triggered once and the variable data that is required stored in word 4 of the Output Data table.

Table 6: Displaying Variable data: Example 1

Word	Step 1	Step 2	Step 3	Step 4
0	2	2	2	2
1	20	20	20	20
2	0	5	5	5
3	-1	0	0	0
4	0	30	31	32
Result	Message queue is cleared and the background message is displayed.	Message 0 is removed and message 5 is displayed as "Oven Temperature 30°C".	Message 0 is removed and message 5 is displayed as "Oven Temperature 31°C".	Message 0 is removed and message 5 is displayed as "Oven Temperature 32°C".
NOTE: All values shown in decimal.				

Displaying variable data example 2

In this second example, message 30 contains variables 17 and 18 that are used for a manufacturer's production line speed in XX.X FPM. Message 30 is created using the Alpha® Automation software (or Gateway Messaging software) in a format that will look like this:



Color: Red

Message:

Mode: Hold

Font: 7 High

Color: Green

Justification: Left

Line Speed is {None:17}.{None:18} FPM

We want to display this message on the sign with address 4 using the *Priority messaging* method. This only requires that the message be triggered once, with variable data stored in the appropriate words of the Output Data table (in this case words 20–23).

Table 7: Displaying Variable data: Example 2

Word	Step 1	Step 2	Step 3	Step 4
0	2	2	2	2
1	4	4	4	4
2	0	30	30	30
3	-1	-1	-1	-1
20	17	17	17	17
21	6	14	14	14
22	18	18	18	18
23	0	3	2	4
Result	Message queue is cleared and the background message is displayed.	Message 30 is displayed as “Line Speed is 14.3 FPM”.	Message 30 is displayed as “Line Speed is 14.2 FPM”.	Message 30 is displayed as “Line Speed is 14.4 FPM”.
NOTE: All values shown in decimal.				

Input Data table from the AGI to the PLC

The Input Data table consists of 16 words that are used in conjunction with the Output Data table for the function that is being used.

Various words are used to determine the status of a sign, the status of the Gateway and requested data.

The Input Data table also gives feedback to the PLC. If there is a problem trying to communicate with a specific sign address (1-31), it will alert the PLC if there is a major fault and store data values requested by the PLC.

The functions/features in the Input Data table are:

- Operation Counter
- Operation Status of the Gateway
- Output Data Table Integrity (O.D.T.I.)
- Sign Status
- Requested Data

Table 8: Input Data table

Word	Meaning	Valid data (decimal)
0	Operation Counter	0 – 65535
1	Error Status (sign address #1-15) / Output Data Table Integrity	Output Data Table Integrity bit (Bit 0) 0 = Gateway completed processing of data or no error 1 = Improper data in Output Data Table Error Status (Bits 1-15) 0 = Gateway received a response from a sign 1 = Gateway did not receive a response from a sign
2	Error Status (sign address #16-31)	Error Status (Bits 0-15) 0 = Gateway received a response from a sign 1 = Gateway did not receive a response from a sign
3	Reserved	Reserved for future use
4	Requested Data #1	0 – 65535
5	Requested Data #2	
6	Requested Data #3	
7	Requested Data #4	
8	Requested Data #5	
9	Requested Data #6	
10	Requested Data #7	
11	Requested Data #8	
12	Requested Data #9	
13	Requested Data #10	
14	Requested Data #11	
15	Requested Data #12	

Operation Counter/Output Data table Integrity bit (ODTI)

The Operation Counter (word 0) is used to let the PLC know that it has completed a task. When the Gateway receives data from the PLC, it begins to process the operation immediately. Once it has successfully completed the operation, the Operation Counter is incremented by 1. The counter automatically resets itself back to zero when it reaches 65,535 (0xFFFF).

The Output Data Table Integrity bit (or ODTI, word 1, bit 0) is similar to the Operation Counter, but it lets the PLC know that the Gateway is currently processing data. If the bit is set high (1), it indicates that the Gateway is processing data. If the bit is set low (0), it indicates that the Gateway has completed processing of data. If the Gateway encounters a problem, such as invalid data sent by the PLC, it will also set this bit high (1). This will remain high until the Gateway receives valid information. The Operation Counter is then incremented.

Table 9: ODTI bit invalid data

If Output Data table Word 0 is:	Then Word(s):	(whose function is:)	will have Word 1, Bit 0 set to 1 (invalid data) if:
2	1	Sign Address (SA)	SA > 255 OR S A = 0
	2	Add Message (AM)	AM < 1 or (≥ 4001 and ≤ 4094) or ≥ 4096
	3	Remove Message (RM)	RM (> -1 and < 1) or (≥ 4001 and ≤ 4094) or ≥ 4096
	2, 3	Add Message / Remove Message	AM = RM
	20, 22, 24, 26, 28, 30	Valid Variable ID (VID)	VID = 0 or < 0 or >100
3	1	Read message queue for Sign Address (SA)	SA > 254 or SA = 0
	2	Read Register Index (RI) for message queue	RI > 6 or RI = 0
4	1	Read variable data for Sign Address (SA)	SA > 254 or SA = 0
	2	Read Register Index (RI) for variable data	RI > 9 or RI = 0

Sign Status

The Sign Status bit consists of two words that indicate whether the sign gave an acknowledge back to the Gateway.

When the Gateway receives an acknowledgement, it will clear (0) the appropriate bit. The Gateway will set the bit if the sign does not give an acknowledgement back after two attempts, 500 milliseconds apart.

- Word 1 — Bit 0 of the word is for the Output Data Table Integrity bit (ODTI). Bits 1-15 of the word contain the Error Status for sign addresses 1-15.
- Word 2 — Bits 0-15 of the word contain the Error Status for sign addresses 16-31.

Table 10: Word 1 of Input Data table

Word 1	Most Significant Byte								Least Significant Byte (ODTI = Output Data Table Integrity bit)							
	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Sign Number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	ODTI
Data	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 11: Word 2 of Input Data table

Word 2	Most Significant Byte								Least Significant Byte							
	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Sign Number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Data	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Clearing the Sign Status bits

There are two ways to clear the bits in words 0-2: all at once or individually:

- To clear all the bits in words 0-2, configure the Output Data table using the information found in step 1 of Table 12.
- To clear a bit in words 1 and 2 representing a specific sign address, configure the information in the Output Data table using the information found in step 2 of Table 12:

Table 12: Methods of clearing the Sign Status bits

Word	Step 1	Step 2
0	-1 (0xFFFF)	-1 (0xFFFF)
1	3	3
2	1–4000, 4095	0
Result	Clears words 0-2 of the Input Data table. Sets all bits to 0.	Clears only bit 3 in word 1 representing sign number 3.

Input Data table words 0 - 2 example

The following table shows information relating to Words 0, 1 and 2 in the Input Data table.

In this example, the Gateway has completed 100 operations (word 0). Bit 0 word 1 is set high (1) and is indicating that the Gateway is currently processing data or that the PLC sent invalid data. Bit 8 of word 1 and bits 1 and 2 of word 2 are set high (1) representing that the Gateway did not receive any response from sign numbers 8, 17, and 18:

Table 13: Input Data table example

Word	Decimal	MSB	LSB	MSB	LSB	Meaning
0	00100 operations	00	64	0000 0000	0110 0100	Operation Process Counter
1	00257	01	01	0000 0001	0000 0001	Error Status (#1-15) /Process Status
2	00006	00	06	0000 0000	0000 0110	Error Status (#16-31)

Sign address 8
(bit 8, Word 1)

Output Data Table Integrity bit
(bit 0, Word 1)

Sign address 18 and 17
(See Table 9,
least significant byte.
See bit 1 and 2 of word 2.)

Requesting data from an Alpha® sign

When data is requested by the PLC, the PLC stores the requested information in words 4-15 of the Input Data table.

This data can either be currently running messages (in the Message Queue) or variable data from a sign. These words are reserved in the Input Data table that the Alpha® Gateway Interface uses only for returning message information or variable data to the PLC.

Requesting currently running messages in a sign

The Alpha® Gateway Interface allows you to read currently running messages in a sign's queue by using the Read Message Queue function.

The message queue allows for up to 64 messages running at one time on any given sign. These message numbers are stored in the sign's memory from registers 103 to 166. Register 103 represents the most recent message number to be added. Register 166 represents the message number that has not been activated for the longest interval.

Since there are only 12 words available in the Input Data table for the requested data, the data is returned to the PLC in 6 groups of 12 words.

The group of 12 words that are returned to the PLC is determined by the Starting index (word 3 of the Output Data table). The Starting Index value and the associated registers that will be returned to the PLC are shown in the following section of the Output Data table.

Table 14: Requesting Message Queue data (Output Data table)

Word	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Case 7
0	Control Word	3	3	3	3	3	3
1	Sign #	6	6	6	6	6	6
2	Starting Index	1	2	3	4	5	6
Meaning	Read registers from sign #6	Read registers 103-114	Read registers 115-126	Read registers 127-138	Read registers 139-150	Read registers 151-162	Read registers 163-166 NOTE: Words 8–15 set to -1d (0xFFFF)

The requested data is stored in the Input Data table in words 4 through 15.

Requesting variable data from a sign

The Gateway allows you to read the variable data that is currently stored in a sign.

All signs on the network will have the same variable data. There are 100 variable registers in a sign. This variable data is stored in the sign memory from registers 001 to 100. Since there are only 12 words available in the Input Data table for the requested data, the data is returned to the PLC in one of 9 groups of 12 words.

The group of 12 words that are returned to the PLC is determined by the Starting index (word 3 of the Output Data table).

The Starting index value and the associated registers that will be returned to the PLC are shown in the following section of the Output Data table:

Table 15: Requesting variable data (Output Data table)

Word	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Case 7	Case 8	Case 9
0	Control Word	4	4	4	4	4	4	4	4
1	Sign #	6	6	6	6	6	6	6	6
2	Starting Index	1	2	3	4	5	6	7	8
Meaning	Read registers from sign #6	Read registers 001-013	Read registers 014-026	Read registers 027-039	Read registers 040-052	Read registers 053-065	Read registers 066-078	Read registers 079-091	Read registers 092-100 NOTE: Words 13–15 set to -1d (0xFFFF)

The data requested will be sent back to the PLC using words 4–15 in the Input Data table.

Dip switch settings

Address	MSB	1 = On, 0 = Off				LSB		Address	MSB	1 = On, 0 = Off				LSB
	1	2	3	4	5	6			1	2	3	4	5	6
1	1	1	1	1	1	1		33	0	1	1	1	1	1
2	1	1	1	1	1	0		34	0	1	1	1	1	0
3	1	1	1	1	0	1		35	0	1	1	1	0	1
4	1	1	1	1	0	0		36	0	1	1	1	0	0
5	1	1	1	0	1	1		37	0	1	1	0	1	1
6	1	1	1	0	1	0		38	0	1	1	0	1	0
7	1	1	1	0	0	1		39	0	1	1	0	0	1
8	1	1	1	0	0	0		40	0	1	1	0	0	0
9	1	1	0	1	1	1		41	0	1	0	1	1	1
10	1	1	0	1	1	0		42	0	1	0	1	1	0
11	1	1	0	1	0	1		43	0	1	0	1	0	1
12	1	1	0	1	0	0		44	0	1	0	1	0	0
13	1	1	0	0	1	1		45	0	1	0	0	1	1
14	1	1	0	0	1	0		46	0	1	0	0	1	0
15	1	1	0	0	0	1		47	0	1	0	0	0	1
16	1	1	0	0	0	0		48	0	1	0	0	0	0
17	1	0	1	1	1	1		49	0	0	1	1	1	1
18	1	0	1	1	1	0		50	0	0	1	1	1	0
19	1	0	1	1	0	1		51	0	0	1	1	0	1
20	1	0	1	1	0	0		52	0	0	1	1	0	0
21	1	0	1	0	1	1		53	0	0	1	0	1	1
22	1	0	1	0	1	0		54	0	0	1	0	1	0
23	1	0	1	0	0	1		55	0	0	1	0	0	1
24	1	0	1	0	0	0		56	0	0	1	0	0	0
25	1	0	0	1	1	1		57	0	0	0	1	1	1
26	1	0	0	1	1	0		58	0	0	0	1	1	0
27	1	0	0	1	0	1		59	0	0	0	1	0	1
28	1	0	0	1	0	0		60	0	0	0	1	0	0
29	1	0	0	0	1	1		61	0	0	0	0	1	1
30	1	0	0	0	1	0		62	0	0	0	0	1	0
31	1	0	0	0	0	1		63	0	0	0	0	0	1
32	1	0	0	0	0	0		64	0	0	0	0	0	0