AVEVA™ Communication Drivers Pack – Allen Bradley - ABTCP Driver

User Guide

Publication date: Tuesday, May 9, 2023

Publication ID: 868872

Contact Information

AVEVA Group Limited
High Cross
Madingley Road
Cambridge
CB3 0HB. UK

https://sw.aveva.com/

For information on how to contact sales and customer training, see https://sw.aveva.com/contact.

For information on how to contact technical support, see https://sw.aveva.com/support.

To access the AVEVA Knowledge and Support center, visit https://softwaresupport.aveva.com.

Contents

AVEVA

Chapter 1

Getting Started

This document describes the ABTCP Communication Driver, and the device and protocol environment in which it works. It includes application-level and bus-level communications protocols, item naming conventions, and Communication Driver features.

Welcome

The ABTCP Communication Driver (referred to as the Communication Driver through the remainder of this user's guide) is a Microsoft® Windows® application program that acts as a communications protocol server. Its user interface is a snap-in Microsoft Management Console (MMC) program, which is part of the Operations Control Management Console (OCMC) suite of utilities.

This Communication Driver allows other Windows application programs access to data in PLCs (also referred to as devices) attached to an Ethernet network or through a Pyramid Integrator module. The Communication Driver requires a TCP/IP package that supports the WinSock interface standard. It can access data directly via the Ethernet in programmable controllers such as SLC-5/05, Ethernet PLC-5, or PLC-5/250 using an Ethernet Interface Module (a Pyramid EI integrator module).

Supported Client Protocols

The client applications connect to the ABTCP Communication Driver using following protocols:

- OPC
- SuiteLink
- DDE/FastDDE
- PCS

For more information refer to the "Support Client Protocols" section of the AVEVA Communication Drivers Pack Help.

Supported Device Protocols

The ABTCP Communication Driver is designed to provide direct connectivity to the family of Allen Bradley PLC5, SLC500 and PLC5250 controllers.

The ABTCP Communication Driver uses TCP/IP (Transmission Control Protocol/Internet Protocol) bus-level protocol.

TCP is the lower-level transport and data-link vehicle for data delivery over an IP network. It provides reliable connection-oriented full-duplex data stream transport. IP is the basic protocol for the Internet which uses an IP address scheme to send data in packets across networks.

Supported Devices

The ABTCP Communication Driver will provide connectivity to the Allen-Bradley controllers described in this section:

AVEVA

Chapter 2

Configuration

Network Communication Bridge/Interface Modules are the communication links between the ABTCP Communication Driver and its supported Allen-Bradley controllers. You must create these links within the OI Server Manager hierarchy to bridge/route control and information data between different networks to target controllers.

This is accomplished by creating Port Objects. These Port Objects simulate the physical hardware layout and must be built to establish communications to each of the controllers. Once you have built the ABTCP hierarchy, you can configure the respective devices for communications. Finally, you can create the desired Device Groups for each controller.

Before you add these Ports in the OCMC, you need to identify your hardware topology to the devices being connected.

Finding Your Communication Driver in the Operations Control Management Console

Each Communication Driver is identified by a unique program name (ProgID) under the OCMC. The ProgID for this ABTCP Communication Driver is: OI.ABTCP.1.

On the computer where the Communication Driver is installed, it can be found in the local node of the default group of the OI Server Manager.

You do not need to install the OI Server Manager on the same computer as the Communication Driver. When you access the Communication Driver remotely, you will not find the Communication Driver node under the local node. You must locate and identify the Communication Driver on a computer in one of the node groups.

To find the Communication Driver

1. On the system Start menu, click **Programs**. Navigate to the AVEVA folder that contains the Operations Control Management Console and then click Operations Control Management Console.
2. In the **Operations Control Management Console**, expand OI Server Manager.

3. Locate the group with the ABTCP node.
4. Expand the **OI.ABTCP.1** node, and select the **Configuration** node to display the global parameters.

To view global parameter configuration and settings in detail, see "Configuring Global Parameters" in the Communication Drivers Pack Help.

Setting Up Your Communication Driver

This section describes the procedures required to set up the ABTCP Communication Driver for use with the supported devices gateways and communication-interfaces.

Many high-level functions and user-interface elements of the OI Server Manager are universal to all Communication Drivers. These universal functions are described in detail in the AVEVA Communication Drivers Pack Help.

See the Communication Drivers Core, to find more information on:

- Global parameters
- Configuration sets
- Time zone features
- Icon definitions
- Activation/deactivation
- Importing/exporting device items
- Standard diagnostics

The ABTCP Communication Driver uses the TCP/IP to communicate with all devices across an Ethernet network.

The ABTCP Hierarchy in the Communication Driver starts with the TCP/IP PORT Object, followed by supported ABTCP controllers.

Note: Before attempting to configure your Communication Driver, you should determine the hierarchical structure of your network/controller environment.

Before You Begin

Before configuring the Communication Driver, verify the following items:

- A PC is set up with the necessary network cards, and is connected to the necessary networks.
- The Windows administration account is created or identified.
- The Communication Driver and any other Wonderware software such as the OI Server Manager is installed with the proper licenses. For more information, see the AVEVA Enterprise Licensing documentation.
- The client software is installed.
- The device(s) is/are connected (networked) and, if necessary, programmed.

Before configuring the Communication Driver, you should know:

- The device network configuration and addresses.
- Which data items are needed for the client application.

- The device name/topic name/group name.
- The desired update intervals.

Configuring the Communication Driver

Note: This Communication Driver is hosted by the OI Server Manager, a Microsoft Management Console (MMC) snap-in, which is part of the Operations Control Management Console (OCMC) suite of utilities. Many high-level functions and user-interface elements of the OI Server Manager are universal to all Communication Drivers, and only the documentation for the OI Server Manager contains descriptions of those universal functions/UI elements. Therefore, reading the documentation for both the MMC and the OI Server Manager is critical to understanding this user's guide. To read the documentation about the MMC and OI Server Manager, click the Help topics on the OCMC Help menu. Both the MMC and OI Server Manager Help is displayed. An Adobe Acrobat version of the Communication Drivers Pack Help is provided.

Note: The shortcut menu items described in this document typically represent only a subset of any actual shortcut menu. Most items in each shortcut menu are standard Windows commands. See the MMC Help for more information about those commands.

Before the Communication Driver can be configured, the Ethernet adapter and TCP/IP software must be installed on your computer; and, if you plan to configure your Communication Driver using Host Names rather than the IP addresses directly, the Host Names for all PLCs on the Ethernet need to be defined by modifying the hosts file.

To install the Ethernet adapter

1. Install the Ethernet adapter and TCP/IP software following the instructions provided by the manufacturer.
2. The Communication Driver configuration accepts either Host Names or IP addresses directly.
   - You can configure Host Names directly in the OI Server Manager.
   - Optionally, if Host Names will be used, then the Host Names for all PLCs on the network need to be defined by modifying the hosts file.
   - Refer to the documentation provided by Microsoft for information about the location of the hosts file and the format of its contents.

Note: All configuration files will be moved to "C:\Documents and Settings\All Users\Application Data\Wonderware\DAServer\DASABTCP" on Windows XP and Windows Server 2003 operating systems and to "C:\ProgramData\Wonderware\DAServer\DASABTCP" on Windows Vista and above operating systems. The rule file will provide the version information.

1. Restart the computer.

To prepare the ABTCP Communication Driver

1. The ABTCP Communication Driver is installed along with the Communication Drivers Pack. It is a selectable option during the Communication Drivers Pack installation.
2. Accept all the default settings during installation.

   Note: Since there are no default values for security settings, you must take note of the User Name and password selected during install.

1. When the ABTCP hierarchy build has been completed, you can start configuring the respective devices for communications.

- Optionally, the desired device groups can be created under the **Device Groups** tabbed page with each of the PLC objects.
- Desired device items can also be optionally created under the **Device Items** tabbed page with each of the PLC objects.

Note: Any Global Parameters that appear dimmed are not supported. The hierarchy entry is added in the "edit mode," providing a convenient place for you to appropriately describe components of your specific hardware environment. Both hierarchy node name and device group name are numerically sequenced by default. They can be renamed at any time.

The Communication Driver will be ready to use after it is activated.

- The Communication Driver can be activated by right-clicking on **OI.ABTCP.1** and selecting **Activate Server** from the shortcut menu. For more information about different modes of activating the communication driver, see the Communication Drivers Pack Help.

Note: When any configuration view is in an open state and you open the same server the second time, the Communication Driver locks the second instance of this same-server access for any update or configuration activities. Access to this second opening instance will resume after the first one has been closed.

Configuring PORT_TCPIP Connection

The ABTCP Hierarchy in the Communication Driver starts with the PORT_TCPIP connection, followed by the supported communication-interface/gateway modules that allow the Communication Driver to access the supported networks and devices.The logical endpoint for each branch of the ABTCP hierarchy tree is always a Processor Type node which represents the controller device.

The following sections detail the steps necessary to configure your Communication Driver Port Objects according to your network type.

Note: Before attempting to configure your Communication Driver, you should determine the hierarchical structure of your device/network environment.

The server-specific configuration portion of the ABTCP Communication Driver hierarchy tree under the OI Server Manager starts at the **TCPIP** object. It is a logical representation of the Ethernet port for TCPIP communications in a computer.

Only one TCPIP object is allowed per ABTCP Communication Driver, and is added by default when you first install the OI Server. If you subsequently clear your configuration hierarchy, you must create this TCPIP port object.

To add PORT_TCPIP connection from the Configuration branch

1. Right-click **Configuration**, and select **Add PORT_TCPIP Connection**.

   An object called **New_PORT_CIP_000** is created.

2. Rename the newly created object as appropriate. The **Port_CIP_000 Parameters** configuration view is displayed in the **Configuration** branch of the hierarchy.

This configuration view has two parameters, one of which is configurable:

- **Port type:** This information is provided automatically by the OI Server Manager (TCP/IP).
- **Maximum outstanding messages:** The number of messages that the Communication Driver can send to an IP address before getting acknowledgement from that destination.
  - The default value is 4.

- The valid range is 1 to 20.

From the New_PORT_TCPIP_000 branch of the Communication Driver hierarchy, the following objects can be created:

- PLC5_TCPIP Object (representing PLC5/20E through PLC5/80E PLCs)
- SLC500_TCPIP Object (representing SLC5/05 PLC)
- PYRAMID_EI Object (representing your Ethernet Interface 5820-EI module)

Configuring Port Objects

From the **New_PORT_TCPIP_000** branch of the Communication Driver hierarchy, the following objects can be created:

- PLC5_TCPIP connection (representing PLC5/20E through PLC5/80E PLCs)
- SLC500_TCPIP connection (representing SLC5/05 PLC)
- PYRAMID_EI connection (representing your Ethernet Interface 5820-EI module)

PLC5_TCPIP Object

The PLC5_TCPIP object represents the logical endpoint to the hardware hierarchy. It is created from the New_PORT_TCPIP_000 branch of the Communication Driver hierarchy.

To add the PLC5_TCPIP object to your ABTCP hierarchy

1. Right-click on the **New_PORT_TCPIP_000** object.
2. Select **Add PLC5_TCPIP Connection** from the shortcut menu. A New_PLC5_TCPIP _000 object is created.
3. Rename the newly created object as appropriate. The PLC5_TCPIP parameters view is displayed.

Note: You can add up to 1024 of each type object to the hierarchy.

1. The PLC5_TCPIP parameters view is displayed.

This configuration view has six configurable elements:

- **Host Name:** Host name or IP address of the destination.
  - The number of characters cannot be more than 255.
  - The field cannot be blank.
  - The host name is defined in the system host file, usually found in: \WINNT\system32\drivers\etc\hosts).

Note: The Host Name defaults to the LocalHost. If the LocalHost is selected and deleted, resulting in a blank Host Name field, and you apply the changes, this will result in an error message.

- **Data block size:** The number of date bytes that can be sent in a message for this hierarchy.
  - For reading blocks:
    The default value is 2000 bytes.
    The valid range is 2 to 2000.
  - For writing blocks:
    The maximum writing block is 220 bytes, and not configurable.

- **Connection timeout:** Time (in milliseconds) allowed to establish a socket connection to a target device.
  - The default value is 2000 ms.
  - The valid range is 1000 to 20000 ms.
- **Reply timeout:** The time (in seconds) the Communication Driver will wait for the acknowledgement after it sends out a message. The message will be resent when time-out occurred.
  - The default value is 15.
  - The valid range is 1 to 300 seconds.
- **Supports PID and string files:** Set this flag if using a 1785 PLC-5 that supports PID, ASCII String, Block Transfer, and SFC Status Files.
- **Supports unsolicited 'CLIENT' messaging:** Set this flag to true to enable the Communication Driver to receive unsolicited "CLIENT" data updates from the selected PLC.
  - The Communication Driver is enabled automatically to accept normal, "Peer-to-Peer" unsolicited data updates without this flag set to true.
  - The reason to turn on this flag is if the MSG instructions programmed inside the PLC/Host is configured with "CLIENT" as the target station, rather than a specific IP address on the network.

SLC500_TCPIP Object

The SLC500_TCPIP object is created from the New_PORT_TCPIP_000 branch of the Communication Driver hierarchy.

To add the SLC500_TCPIP object to your ABTCP hierarchy

1. Right-click on the **New_PORT_TCPIP_000** object.
2. Select **Add SLC500_TCPIP Connection** from the shortcut menu. A New_SLC500_TCPIP_000 object is created.
3. Rename the newly created object as appropriate. The SLC500_TCPIP parameters view is displayed.

Note: You can add up to 1024 of each type object to the hierarchy.

- The SLC500_TCPIP object represents the logical endpoint to the hardware hierarchy.
- The **New_SLC500_TCPIP_000 Parameters** configuration view is displayed.The SLC500_TCPIP parameters view is displayed.

The SLC500_TCPIP object is used to configure the SLC-500 processor on an Ethernet direct connection. This configuration view has four configurable elements:

- **Host name:** Host name or IP address of the destination.
  - The number of characters cannot be more than 255.
  - The field cannot be blank.
  - The host name is defined in the system host file (usually it is \WINNT\system32\drivers\etc\hosts).

Note: The Host Name defaults to the LocalHost. If the LocalHost is selected and deleted, resulting in a blank Host Name field, and the Apply button is clicked on, an error message will appear.

- **Data block size:** The number of date bytes that can be sent in a message for this hierarchy.

  - For reading blocks:
    The default value is 510 bytes.
    The valid range is 2 to 510.
  - For writing blocks:
    The maximum writing block is 220 bytes, and not configurable.
- **Connection timeout:** The time allowed to establish a socket connection to a target device.
  - The default value is 2000 milliseconds.
  - The valid range is 1000 to 20000 milliseconds.
- **Reply timeout:** The time (in Seconds) the Communication Driver will wait for the acknowledgement after it sends out a message.
  - The message will be resent when time-out occurred.
  - The default value is 15.
  - Valid range is 1 to 300 seconds.

Multi-Instance and Cloning

The ABTCP Communication Driver supports multiple server instances. You create additional server instances and even clone existing server instances in those server groups.

Note: Creating multiple instances requires a Professional level license.

For a detailed step-by-step description of creating a new instance, or cloning an existing instance of the Communication Driver, see "Instantiating Data Sources" in the Communication Drivers Pack Help.

AVEVA

Chapter 3

Device Groups and Device Items

The **Device Groups** tab in the OI Server Manager user interface is used to create new, modify, or delete device group definitions for an object. For DDE/SuiteLink communications, one or more device group definitions must exist for each PLC that the Communication Driver will communicate with.

Important! For DDE/SuiteLink, it is strongly recommended that each device group (topic) definition contain a unique name for the PLC associated with it. The OPC, however, has the flexibility to use any names, including duplicate names, for the device group definitions.

Device Group Definitions

The **Device Groups** dialog box, which is displayed by clicking the **Device Groups** tab in the **New_<Name>PLC_000** Parameters configuration view, is used to perform the following activities:

- Adding, defining, and deleting device groups.

Note: When you add a new device group, enter a unique name.

- Configuring default update intervals.
- Editing update intervals for the objects.

Note: When you select another part of the Communication Driver tree hierarchy, you are prompted to save the modifications to the configuration set.

To create or add device groups

1. Right-click in the **Device Groups** dialog box.
2. Select the **Add** command from the shortcut menu.
    - When you add a new device group, enter a unique name (up to 32 characters long).

To make edits on device groups

Make edits on a device group's name or update interval for an object as follows:

- In the **Name** column, double-click on the device group's name to be edited and make the edit.
- Double-click on the device group's value to be edited in the **Update Interval** column, and make the edit.

To delete device groups

Deleting a device group from the list can be performed as follows:

1. Right-click on the device group to be deleted.
2. Select the **Delete** command from the shortcut menu.

Note: When you select another part of the ABTCP Communication Driver tree hierarchy, you are prompted to save the modifications to the configuration set.

To configure default update intervals

1. To configure a default update interval for the object, right-click in the **Device Groups** dialog box.
2. Select **Config Default Update Interval** from the shortcut menu.

To edit update intervals

- To edit the update interval for an object, double-click its value in the **Update Interval** column and make the edit.
  - Update Interval is the frequency (in milliseconds) that the Communication Driver acquires data from the topics associated with that device group.
  - Different topics can be polled at different rates in a PLC by defining multiple device-group names for the same PLC and setting a different Update Interval for each device group.

Note: When you select another part of the ABTCP Communication Driver tree hierarchy, you are prompted to save the modifications to the configuration set.

Each configuration view associated with nodes/objects in the Communication Driver hierarchy tree has a common feature, the **Save** button.

1. When you modify any parameters in the **Device Groups** dialog box, click **Save** to save and implement the new modifications.
   - If you do not click **Save**, the configuration is reset to its original condition (since the last save).
2. After all modifications, you must save when prompted for the new data to be saved to the configuration set.

Device Item Definitions

The Device Items tab in the **New_<Name>PLC_000 Parameters** configuration view is used to define aliases to actual PLC items. The **Device Items** dialog box is the place where the following activities are performed:

- Creating new device item definitions for PLC items.
- Modifying the existing device items.
- Deleting device items.
- Archiving the created list of device items to a .csv file, a file with values separated by commas.
- Bringing a .csv file into the **Device Items** tab.

Each device item definition should contain a unique name for the PLC associated with it.

The **Device Items** dialog box has the following two columns:

- **Name:** This column defines the alias names to actual PLC items.
- **Item Reference:** The actual PLC item names, linked to the created aliases, are defined in this column.

For example:

Name	Item Reference
Timer	n7:0
Float	f8:1

Note: When you create or add a new device item, a unique name needs to be entered for it.

Once the Device Items feature is utilized to configure item names, it provides the Communication Driver with the capability to perform OPC Item browsing. When the Communication Driver is running and an OPC client requests item information, the configured items will show up under the PLC hierarchy node.

Note: Device items have the precedence in addressing items in the controller device at runtime. Items request from the client would be searched from the Device Items Name list first before going out to the controller.

To create or add device items

1. Right-click in the **Device Items** dialog box.
2. Select the **Add** command from the shortcut menu.
   - A device item is created in the **Name** column, and it is numerically named by default. For example, Item_0, Item_1, and so on.
3. Change the default name by double-clicking on it and entering the new name.
   - Enter a unique name for the new device item. For example, "Timer."

To add item references

Item references for each of the device items that have been created can be added as follows:

1. In the **Item Reference** column, double-click on the area in the same horizontal line as the selected device item.
2. Type in the actual PLC item name in the frame that appears.
   - For example, "n7:0."
3. Click anywhere in the dialog box or press the ENTER key to have the change take effect.

Note: System items are not valid item reference, but Communication Driver-specific system items are ok.

To rename a device item from the list

1. Right-click on the device item to be renamed.
2. Select the **Rename** command from the shortcut menu and enter the new device item name.
3. Click anywhere in the dialog box or press the ENTER key to apply the change.

To delete a device item from the list

1. Right-click on the device item to be deleted.
2. Select the **Delete** command from the shortcut menu.
   - The device item and its corresponding actual PLC item name will be deleted from the dialog box.

Note: When you select another part of the ABTCP Communication Driver tree hierarchy, you are prompted to save the modifications to the configuration set.

To clear all device items

1. Right-click anywhere in the **Device Items** dialog box.
2. Select the **Clear All** command from the shortcut menu.
   - All the device items listed in the dialog box, including their corresponding actual PLC item names, will be deleted.

Exporting and Importing Communication Driver Item Data

The Export and Import commands on the shortcut menu enable you to export and import the Communication Driver item data to and from a CSV file, after the configuration of the Device Items has been completed. These commands will allow you to perform an off-line, large-scale edit on the item data configured for a controller, and import what has been edited back into the controller configuration.

The **Export** and **Import** features on the shortcut menu of the Device Items dialog box enable you to export and import the Communication Driver device item data to and from a CSV file, after the configuration of the **Device Items** has been completed. These features provide you with the following capabilities:

- Archive lists of device items.
- Bring an archived list of device items into the **Device Items** dialog box when you need to utilize or reconfigure any of the device items on the archived list.
- Perform an off-line, large-scale edit on the item data configured for a PLC.
- Import what has been edited back into the PLC configuration.

To export device items

1. Right-click anywhere in the **Device Items** dialog box.
2. Select the **Export** command from the shortcut menu.
   - The standard **Save As** dialog box appears.
   - The file name has defaulted into "PLC Hierarchyname.csv," within the current-system-configured default directory.
3. Accept the defaults to save the file.
   - The file is saved as New_PLC5_TCPIP_000.csv.
   - It is editable in Microsoft Excel.

   However, if you prefer to save the list someplace else and rename it, perform the following steps after step 2.

4. Select the folder into which the list is to be saved.
5. Name the list to be archived.
6. Click the **Save** button.
    - The whole list will be saved as a .csv file in Excel.

The file can now be edited off-line. It contains one row for each item configured with two columns, Name and Item Reference, respectively.

To import device items

1. To import the list, right-click anywhere in the **Device Items** dialog box.
2. Select the **Import** command from the shortcut menu.
3. Select the archived list (.csv file) to be imported from the folder in which it is saved.
4. Click the **Open** button.
    - The whole list will be brought into the **Device Items** dialog box.

Note: When the list to be imported contains duplicate names as found in the current list but the Item References are different, a dialog box will appear to prompt you to make a selection.

To import device item data that has been edited off-line

1. Right-click anywhere in the **Device Items** dialog box.
2. Clear all the item data you wish to replace with the edited .csv file by selecting the **Clear All** command.
    - The data will be cleared after you click on **Yes** to confirm the deletion.
3. Select the **Import** command from the shortcut menu.
    - The standard **Open** dialog box appears.
    - It defaults to the .csv file extension within the current-system-configured default directory.
4. Browse for the specific CSV file you want to import, select it, then click on the **Open** button.
    - The OI Server Manager will import the edited file and deposit it in the **Device Items** dialog box.
    - During the imported file processing:
    - New item references will be added based on unique names.
    - If there are duplicate names, you will be provided with the ability to replace the existing entry with the new entry, or ignore the new entry.

When the Communication Driver is running and an OPC client requests item information, the imported configured items will show up under the PLC hierarchy node.

Scan-Based Message Handling

Communication Driver are based on the concept of polling a hardware device for information. This polling is driven by a need which is expressed in the form of requests from one or more clients.

For DDE/SuiteLink, once a particular piece of information has been requested by a client, the Communication Driver formulates its own request and sends that request to the hardware device. The Communication Driver then waits for a response to its request. Once the information has been received, the Communication Driver

passes that information back to the client, and repeats the process until all clients have ceased requesting information.

The rate at which the Communication Driver will poll a particular device for a specific piece of information is defined in the device group (topic definition) inside the Communication Driver, using a parameter called the Update Interval. When setting this parameter, there is always a trade-off between the update speed of the device group and the resulting decrease in system responsiveness.

If you use OPC interface, in addition to the capabilities described in the preceding paragraph, the OPC Client also has additional capabilities on Update Interval control.

Since very fast response is usually desired, the temptation is to set the Update Interval to a value close to 0 seconds. However, if every point is polled at this rate, the entire system will suffer due to slow response time. Therefore, you should compromise, and set the Update Interval to a more reasonable value. You could also create multiple device groups for each device, setting the Update Interval to different values, then assigning different items to different device groups depending on how quickly the values change and how quickly you want to see an update of those changes.

Some items, like alarms, change very infrequently but because of their importance require very fast updates. For those kinds of items, you should set the Update Interval at a very small value. If you desire an immediate response, set the Update Interval at 1 (one). See the Unsolicited Message Handling section.

Unsolicited Message Handling

In the world of PLCs and Communication Drivers, it is obvious that a PLC will know when a critical event has occurred before the Communication Driver will have a chance to poll for that data. Therefore, it would seem natural that if a critical event occurs, the PLC should have the capability to inform the Communication Driver immediately, without having to wait for the Communication Driver to poll it.

This is the role of an unsolicited message. Once a PLC has determined that a critical condition exists, it can generate a message immediately sent to the Communication Driver without a prior request from the Communication Driver. The unsolicited message implementation requires both the messaging instructions properly programmed in the PLC logic and the device group appropriately configured in the Communication Driver.

The Allen-Bradley processors, specifically the 1785 PLC-5 and PLC-5/250 (EI), are capable of producing unsolicited messages that the Communication Drivers can understand.

There are two types of unsolicited messages supported by the ABTCP Communication Driver:

- **Peer-to-Peer:** This method involves IP-address-to-IP-address communications.
  - Configure which computer (by its IP address) receives unsolicited messages in your client application.
  - Two instances of peer-to-peer unsolicited messages are generated by the Communication Driver:
  - If the value of "Update Interval" for a topic is 0 (zero), the server will poll this topic only once at the start. After that, only an unsolicited message will update the data.
  - If the value of "Update Interval" for a topic is >0 (zero), the server will update the data for a particular item immediately upon receiving an unsolicited message for the item. The Communication Driver will also update the data at every Update Interval.
- **Client Messaging** (also called general broadcast): This method involves a general broadcast of the unsolicited message onto the Ethernet network.

- Only the first node on the network that is capable of receiving the message gets it.
- If this option is not enabled, the Communication Driver will ignore all client messages sent out from the PLC.
- If this option is enabled, the Communication Driver will attempt to respond to the PLC when it gets the first client message from the PLC. Note that if the response is accepted by the PLC, the PLC will send unsolicited messages exclusively to the Communication Driver thereafterwards.
- If the value of "Update Interval" for a topic is 0 (zero), the server will poll this topic only once at the start. After that, only an unsolicited message will update the data.
- If the value of "Update Interval" for a topic is >0 (zero), the server will update the data for a particular item immediately upon receiving an unsolicited message for the item. The Communication Driver will also update the data at every Update Interval.

Note: For a more specific Allen Bradley definition of peer-to-peer and client-messaging unsolicited messaging, please refer to the Ethernet AB PLC-5 Family documentation.

To receive unsolicited messages

- Set a device group's "Update Interval" to 0.

To access the settings for device groups

1. Click on the PLC's name in the **Configuration** hierarchy of your Communication Driver.
2. Select the **Device Groups** tab of the configuration view pane at right.
3. Double-click the number in the Update Interval column of the desired device group and type the number 0 (zero).

Save the configuration change by clicking the **Save** icon Located at the upper-right corner of the configuration view pane.

AVEVA

Chapter 4

Item Names

The ABTCP Communication Driver currently supports item names that follow the conventions described for PLC-5, PLC-5/250, and SLC 500. This chapter describes the item naming conventions for these devices.

For any file type described in this section, in which the .field defines the item as discrete, any /bit designation is ignored.

PLC-5 Item Naming

The general format of item names for data from 1785 PLC-5 controllers matches the naming convention used by the programming software. The following is the format:

[$] identifier [file #] : element [.field] [/bit]

Note: The parts of the name shown in square brackets ([]) are optional.

$ – Purely optional.

identifier – Identifies the file type. The following table summarizes the valid file types, the default file number for each type, and the fields allowed (if any).

file # – File number (0 - 999 decimal).

File 0 must be Output.
File 1 must be Input.
File 2 must be Status.

element – Element number within the file.

For Input and Output files it is also called rack-and-group number and must be 0 - 277 octal.
For all other file types, it must be 0 - 999 decimal.

.field – Valid only for Counter, Timer, Control, ASCII String, PID, SFC Status, and Block Transfer files. Refer to the following table.

/bit – Valid for all file types except ASCII String and Floating Point.

For Input and Output files it must be 0 - 17 octal.
For all other file types it must be 0 - 15 decimal.

Identifier	File Type	Default File #	.fields
O	Output	0	
I	Input	1	
S	Status	2	
B	Binary	3	
T	Timer	4	.PRE .ACC .EN .TT .DN
C	Counter	5	.PRE .ACC .CU .CD .DN .OV .UN
R	Control	6	.LEN .POS .EN .EU .DN .EM .ER .UL .IN .FD
N	Integer	7	
F	Floating Point	8	
A	ASCII	none	
D	BCD	none	
ST	ASCII String*	none	.LEN
PD	PID*	none	.ADRF .ADRE .BIAS .CA .CL .CT .DB .DO .DVDB .DVN .DVNA .DVP .DVPA .EN .ERR .EWD .INI .KD .KI .KP .MAXI .MAXO .MAXS .MINI .MINO .MINS .MO .OLH .OLL .OUT .PE .PV .PVDB .PVH .PVHA .PVL .PVLA .PVT .SO .SP .SPOR .SWM .TIE .UPD
SC	SFC Status*	none	.DN .ER .FS .LS .OV .PRE .SA .TIM
BT	Block Transfer* (Read-Only)	none	.EN .ST .DN .ER .CO .EW .NR .RW .TO .RLEN .DLEN .FILE .ELEM

* Available only on certain PLC-5 models. Check the Processor Manual for the model being used.

Output File Items

O[n]:e[/b]	n represents the file number and is optional. If specified, it must be 0 (zero).
	e indicates the element number in the file.

O[n]:e[/b]	n represents the file number and is optional. If specified, it must be 0 (zero).
	e indicates the element number in the file.
	b specifies the bit (0 - 15 decimal). /b may be omitted, if necessary, to treat the I/O group as a numeric value.

Note: The elements in I/O modules are sequentially mapped into a memory table, and are different from the item names in the PLC programming software. Refer to the Addressing SLC I/O Modules section.

Examples:

O0:0/0

$O:2/15

O:3 4BCD(for 16-bit 7-segment display)

Input File Items

I[n]:e[/b]	n represents the file number and is optional. If specified, it must be 1.
	e indicates the element number in the file.
	b specifies the bit (0 - 15 decimal). /b may be omitted if necessary to treat the I/O group as a numeric value.

Note: The elements in I/O modules are sequentially mapped into a memory table and are different from the item names in the PLC programming software. See Addressing SLC I/O Modules.

Examples:

I1:0/0

I:2/15

I:3 4BCD(for 16-bit thumbwheel input)

Status File Items

S[n]:e[/b]	n represents the file number and is optional. If specified, it must be 2.
	e indicates the element number in the file.
	b is optional. If specified, it indicates the bit (0 - 15 decimal).

Note: Refer to the SLC-500 Family Processor Manual (Allen-Bradley Publication) for a complete description of the Status file information.

Examples:

S2:6(major error fault)

S2:13(math register)

S:1/5(forces enabled)

Binary File Items

B[n]:e/b or B[n]/m	n represents the file number and is optional. If not specified, it is assumed to be 3. If specified, the file number must be 3 or 9 - 255 decimal.
	e specifies the element (word) number within the Binary file. It must be 0 - 255 decimal.
	b specifies the bit number within the word. In the first form (where :e is present), the bit number must be 0 - 15 decimal.
	m also represents the bit number. However, in the second form, no word numbers are specified and the bit number may be 0 - 4095.

Examples:

B:33

B:6/4(same bit as B/100)

B3/4095(same bit as B:255/15)

Timer File Items

T[n]:e[.f][/b]	n represents the file number and is optional. If not specified, it is assumed to be 4. If specified, the file number must be 4 or 9 - 255 decimal.
	e specifies the element number (three words per element) within the Timer file. It must be 0 - 255 decimal.
	.f identifies one of the valid Timer fields. The valid fields for Timer Files are listed in the table. If .f is omitted, it is assumed to be the word containing the status bits.
	b is optional and is normally not used. All of the fields of a timer can be accessed by specifying the .f fields. However, it is possible to use /b to single out a bit in the .PRE or .ACC fields (which are words). The bit number must be 0 - 15 decimal.

Examples:

T4:0.ACC

T4:3.DN

T4:1.PRE

Counter File Items

C[n]:e[.f][/b]	n represents the file number and is optional. If not specified, it is assumed to be 5. If specified, the file number must be 5 or 9 - 255 decimal.
	e specifies the element number (three words per element) within the Counter file. It must be 0 - 255 decimal.
	.f identifies one of the valid Counter fields. The valid fields for the Counter Files are listed in the table. If .f is omitted, it is assumed to be the word containing the status bits.
	b is optional and is normally not used. Specifying the .f fields can access all of the fields of a counter. However, it is possible to use /b to single out a bit in the .PRE or .ACC fields (which are words). The bit number must be 0 - 15 decimal.

Examples:

C5:0.ACC

C5:3.OV

C5:1.PRE

Control File Items

R[n]:e[.f][/b]	n represents the file number and is optional. If not specified, it is assumed to be 6. If specified, the file number must be 6 or 9 - 255 decimal.
	e specifies the element number (three words per element) within the Control file. It must be 0 - 255 decimal.
	f identifies one of the valid Control fields. The valid fields for the Control files are listed in the table. If .f is omitted, it is assumed to be the word containing the status bits.
	b is optional and is normally not used. All of the fields of a Control file can be accessed by specifying the .f fields. However, it is possible to use /b to

	single out a bit in the .LEN or .POS fields (which are words). The bit number must be 0 - 15 decimal.

Examples:

R6:0.LEN

R6:3.EN

R6:1.POS

Integer File Items

N[n]:e[/b]	n represents the file number and is optional. If not specified, it is assumed to be 7. If specified, the file number must be 7 or 9 - 255 decimal.
	e specifies the element number within the Integer file. It must be 0 - 255 decimal.
	b is optional. If specified, it indicates the bit (0 - 15 decimal).

Examples:

N7:0

N7:0/15

N7:3

Floating Point File Items

F[n]:e	n represents the file number and is optional. If not specified, it is assumed to be 8. If specified, the file number must be 8 - 255 decimal.
	e specifies the element number within the Floating Point file. It must be 0 - 255 decimal.

Examples:

F8:0

F8:3

ASCII File Items

An:e[/b]	n represents the file number (NOT optional) and must be 9 - 255 decimal.

	e specifies the element number within the ASCII file. It must be 0 - 255 decimal. Each element in an ASCII file contains two ASCII characters.
	b is optional. If specified, it indicates bit (0 - 15 decimal).

Examples:

A20:3

A10:0/0

BCD File Items

Dn:e[/b]	n represents the file number (NOT optional) and must be 3 - 999 decimal.
	e specifies the element number within the BCD file. It must be 0 - 999 decimal. Each element in a BCD file contains a number between 0 - 9999.
	b is optional. If specified, it indicates the bit (0 - 15 decimal).

Examples:

D20:3

D10:0/3

ASCII String Section Items

STn:e	n represents the file number (NOT optional) and must be 9 - 255 decimal.
	e specifies the element number within the String file. It must be 0 - 255 decimal. Each element in a String file contains an ASCII string with a maximum length of 78 characters.

Examples:

ST9:0

ST9:200

Block Transfer Section Items

BTn:e[.f][/b]	n represents the file number (NOT optional) and must be 3 - 999 decimal.
	e specifies the element number (three words per element) within the Block Transfer file (0 - 999 decimal).

	f identifies one of the valid Block Transfer fields. The valid fields for Block Transfer items are listed in the table. If .f is omitted, it is assumed to be the word containing the status bits.
	b is optional and is normally not used. Specifying the .f fields can access all of the fields of a Block Transfer. However, it is possible to use /b to single out a bit in the .FILE or .ELEM fields (which are words). For Block Transfer files, the bit number must be 0 - 15 decimal.

Note: Block Transfer files are read-only.

Examples:

BT9:0.EN

BT9:3.RLEN

BT9:3.FILE

PID Section Items

[m]PD[n]:e[.f][/b]	m indicates the module to access. If not specified, it is assumed to be 0 (zero), which selects the Resource Manager. If specified, it must be 0 (zero) or the thumbwheel setting for a Logic Processor.
	n represents the file number. If not specified, it is assumed to be 0 (zero). If specified, the file number must be 0 - 9999 decimal.
	e specifies the element number within the PID file. It must be 0 - 9999 decimal.
	.f identifies one of the valid PID fields. The valid fields for PID files are listed in the table. If PID field .ADDR is needed, use .ADRM for module, .ADRE for element, or .ADRF for file.
	b is optional and is normally not used. Specifying the .f fields can access all of the fields of a PID. If specified, it indicates the bit (0 - 15 decimal).

WARNING! Access to PID files may degrade the Communication Driver's performance, due to the extreme size of the PID element (82 words each). If accessing only a few PIDs at one time, performance will not be affected greatly. However, if accessing a few fields of many PIDs at once, it may be faster to move the needed fields to an intermediate file (Floating Point or Binary) and let the Communication Driver access the intermediate files.

Examples:

1PD:0.SP

1PD9:3.OLH

0PD1:0.INI

SFC Status Section Items

SCn:e[.f][/b]	n represents the file number (NOT optional) and must be 3 - 999 decimal.
	e specifies the element number within the SFC Status file. It must be 0 - 999 decimal.
	f identifies one of the valid SFC fields. The valid fields for SFC files are listed in the table.
	b is optional and is normally not used. Specifying the .f fields can access all of the fields of an SFC. For SFC Status items, the bit number must be 0 - 15 decimal.

Examples:

SC9:0

SC9:0.PRE

SC9:0.SA

SLC-500 Item Naming

The general format of item names for data from SLC-500 controllers matches the naming convention used by the programming software. The format is as follows:

[$] identifier [file #] : element [.field] [/bit]

Note: The parts of the name shown in square brackets ([]) are optional.

$ – Purely optional.

identifier – Identifies the file type. The following table summarizes the valid file types, the default file number for each type, and the .fields allowed (if any).

file # – Identifies the file number.

File numbers 0 - 8 decimal are reserved for predefined file types as listed in the table.
All other file numbers, 9 - 255 decimal, are open to all file types.

element – Element number within the file.

For Input and Output files, it must be 0 - 30 decimal.
For all other file types, it must be 0 - 255 decimal.

.field – Valid only for Counter, Timer and Control files; see the following table.

/bit – Valid for all file types except ASCII String and Floating Point.

Must be 0 - 15 decimal.

Identifier	File Type	Default File #	.fields
O	Output	0	
I	Input	1	
S	Status	2	
B	Binary	3	
T	Timer	4	.PRE .ACC .EN .TT .DN
C	Counter	5	.PRE .ACC. CU .CD .DN .OV .UN .UA
R	Control	6	.LEN .POS .EN .DN .ER .U L .IN .FD
N	Integer	7	
F	Floating Point*	8	
A	ASCII*	none	
ST	ASCII String*	none	

*Available only on certain SL-500 models. Check the Processor Manual for the model being used.

Output File Items

O[n]:e[/b]	n represents the file number and is optional. If specified, it must be 0 (zero).
	e indicates the element number in the file.
	e indicates the element number in the file.
	b specifies the bit (0 - 15 decimal). /b may be omitted, if necessary, to treat the I/O group as a numeric value.

Note: The elements in I/O modules are sequentially mapped into a memory table, and are different from the item names in the PLC programming software. Refer to the Addressing SLC I/O Modules section.

Examples:

O0:0/0

$O:2/15

O:3 4BCD(for 16-bit 7-segment display)

Input File Items

I[n]:e[/b]	n represents the file number and is optional. If specified, it must be 1.

	e indicates the element number in the file.
	b specifies the bit (0 - 15 decimal). /b may be omitted if necessary to treat the I/O group as a numeric value.

Note: The elements in I/O modules are sequentially mapped into a memory table and are different from the item names in the PLC programming software. See Addressing SLC I/O Modules.

Examples:

I1:0/0

I:2/15

I:3 4BCD(for 16-bit thumbwheel input)

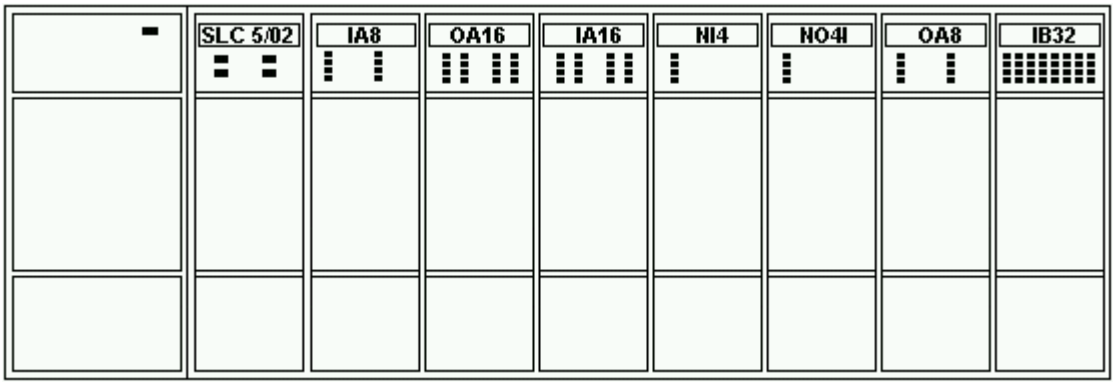
Addressing SLC I/O Modules

The elements (words) in I/O modules are mapped into a memory table. If the Analog I/O modules are being used, then the point naming will differ from the point naming in the programming software. The Communication Driver item name must be computed from the sum total of words used by the previous input or output blocks. The operator can use the programming software Data Monitor to look at the memory map of the I file or O file to verify your address. If the address is unsure, or if the PLC configuration is likely to change, copy the points in question to the N table or B table, and access the data from there.

The naming conventions used in the Allen-Bradley programming software are not supported by the Allen-Bradley Ethernet Direct Communication Driver. The addressing convention is similar to that of the PLC-5 family processors. To derive the correct address for each I/O point, see Diagram System. Also see the following topics, Label I/O Modules with "Word Counts"," Sequentially Number the Input Modules, and Sequentially Number the Output Modules, to complete addressing the SLC I/O modules.

Diagram System

Addressing of the I/O points begins by drawing a schematic of the system. The following figure is a diagram of the SLC-5/02 system.



The far left unit is the power supply. From left to right, the modules are:

1747-L524	SLC-5/02 Module Processor

1746-IA8	8-point 120VAC input module
1746-OA16	16-point 120VAC output module
1746-IA16	16-point 120VAC input module
1746-NI4	4-point 20mA analog input module
1746-NO4I	4-point 20mA analog output module
1746-0A8	8-point 120VAC input module
1746-IB32	32-point DC input module

Label I/O Modules with "Word Counts"

The address of any point within the I/O data table space, in an SLC processor, is the sum of the words occupied by previous modules (to the left in the rack) of the same type. Therefore, to determine the correct address for any particular point in the I/O data table, the number of words each module will consume must be known, as listed in the table below.

Number of Words	Module	
0	1747-L524	SLC-5/02 Module Processor
1	1746-IA8	8-point 120VAC input module
1	1746-OA16	16-point 120VAC output module
1	1746-IA16	16-point 120VAC input module
4	1746-NI4	4-point 20mA analog input module
4	1746-NO4I	4-point 20mA analog output module
1	1746-0A8	8-point 120VAC input module
2	1746-IB32	32-point DC input module

Note: In the preceding table, the minimum number of words which can be consumed by a module is 1 (16-bits). This is due to the memory scheme of all Allen-Bradley processors.

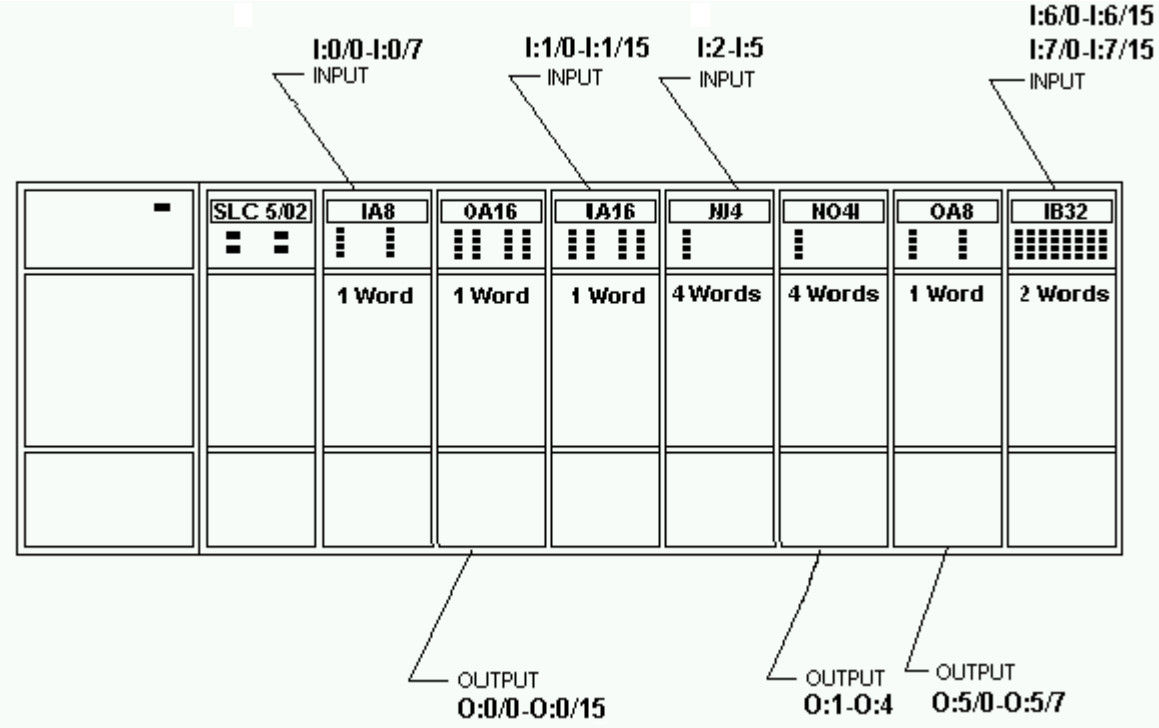
Sequentially Number the Input Modules

In the following I/O diagram, the first input module's addressing should start with "I:0." As previously noted, this module consumes one datatable word. Therefore, the addressing of the next INPUT module encounter, moving from left to right, will begin with "I:1," regardless of the module's physical location.

Sequentially Number the Output Modules

In the following I/O diagram, the first output card encountered is the OA16. Although it is not in the first slot, its address will be "O:0" ("OHH, colon, ZERO"). This module consumes one datatable word. Therefore, the addressing of the next OUTPUT module, moving from left to right, will begin with "O:1," regardless of the module's physical location.

I/O Diagram



Status File Items

S[n]:e[/b]	n represents the file number and is optional. If specified, it must be 2.
	e indicates the element number in the file.
	b is optional. If specified, it indicates the bit (0 - 15 decimal).

Note: Refer to the SLC-500 Family Processor Manual (Allen-Bradley Publication) for a complete description of the Status file information.

Examples:

S2:6(major error fault)

S2:13(math register)

S:1/5(forces enabled)

Binary File Items

B[n]:e/b or B[n]/m	n represents the file number and is optional. If not specified, it is assumed to be 3. If specified, the file number must be 3 or 9 - 255 decimal.
	e specifies the element (word) number within the Binary file. It must be 0 - 255 decimal.
	b specifies the bit number within the word. In the first form (where :e is present), the bit number must be 0 - 15 decimal.
	m also represents the bit number. However, in the second form, no word numbers are specified and the bit number may be 0 - 4095.

Examples:

B:33

B:6/4(same bit as B/100)

B3/4095(same bit as B:255/15)

Timer File Items

T[n]:e[.f][/b]	n represents the file number and is optional. If not specified, it is assumed to be 4. If specified, the file number must be 4 or 9 - 255 decimal.
	e specifies the element number (three words per element) within the Timer file. It must be 0 - 255 decimal.
	.f identifies one of the valid Timer fields. The valid fields for Timer Files are listed in the table. If .f is omitted, it is assumed to be the word containing the status bits.
	b is optional and is normally not used. All of the fields of a timer can be accessed by specifying the .f fields. However, it is possible to use /b to single out a bit in the .PRE or .ACC fields (which are words). The bit number must be 0 - 15 decimal.

Examples:

T4:0.ACC

T4:3.DN

T4:1.PRE

Counter File Items

C[n]:e[.f][/b]	n represents the file number and is optional. If not specified, it is assumed to be 5. If specified, the file number must be 5 or 9 - 255 decimal.
	e specifies the element number (three words per element) within the Counter file. It must be 0 - 255 decimal.
	.f identifies one of the valid Counter fields. The valid fields for the Counter Files are listed in the table. If .f is omitted, it is assumed to be the word containing the status bits.
	b is optional and is normally not used. Specifying the .f fields can access all of the fields of a counter. However, it is possible to use /b to single out a bit in the .PRE or .ACC fields (which are words). The bit number must be 0 - 15 decimal.

Examples:

C5:0.ACC

C5:3.OV

C5:1.PRE

Control File Items

R[n]:e[.f][/b]	n represents the file number and is optional. If not specified, it is assumed to be 6. If specified, the file number must be 6 or 9 - 255 decimal.
	e specifies the element number (three words per element) within the Control file. It must be 0 - 255 decimal.
	f identifies one of the valid Control fields. The valid fields for the Control files are listed in the table. If .f is omitted, it is assumed to be the word containing the status bits.
	b is optional and is normally not used. All of the fields of a Control file can be accessed by specifying the .f fields. However, it is possible to use /b to single out a bit in the .LEN or .POS fields (which are words). The bit number must be 0 - 15 decimal.

Examples:

R6:0.LEN

R6:3.EN

R6:1.POS

Integer File Items

N[n]:e[/b]	n represents the file number and is optional. If not specified, it is assumed to be 7. If specified, the file number must be 7 or 9 - 255 decimal.
	e specifies the element number within the Integer file. It must be 0 - 255 decimal.
	b is optional. If specified, it indicates the bit (0 - 15 decimal).

Examples:

N7:0

N7:0/15

N7:3

Floating Point File Items

F[n]:e	n represents the file number and is optional. If not specified, it is assumed to be 8. If specified, the file number must be 8 - 255 decimal.
	e specifies the element number within the Floating Point file. It must be 0 - 255 decimal.

Examples:

F8:0

F8:3

ASCII File Items

An:e[/b]	n represents the file number (NOT optional) and must be 9 - 255 decimal.
	e specifies the element number within the ASCII file. It must be 0 - 255 decimal. Each element in an ASCII file contains two ASCII characters.
	b is optional. If specified, it indicates bit (0 - 15 decimal).

Examples:

A20:3

A10:0/0

ASCII String Section Items

STn:e	n represents the file number (NOT optional) and must be 9 - 255 decimal.
	e specifies the element number within the String file. It must be 0 - 255 decimal. Each element in a String file contains an ASCII string with a maximum length of 78 characters.

Examples:

ST9:0

ST9:200

AVEVA

Chapter 5

Troubleshooting

This chapter describes troubleshooting tools you can use to deal with the ABTCP Communication Driver problems you may encounter.

The OI Server Manager provides access to diagnostics and other statistical data, and the Log Viewer provides access to event messages logged during the operation of a Communication Driver. Also, your client (for example, InTouch) can monitor connectivity with the PLC through the SYSStatus item. Use these tools together with the information in this section to troubleshoot your ABTCP Communication Driver.

Note: In order to determine the version of your Communication Driver, the following steps need to be performed. Search for DASABTCP.dll, right-click on the File Name, select Properties on the shortcut menu, and select the Version tab on the Properties dialog box. The version of your Communication Driver is listed under File Version.

Monitoring Connectivity Status with the PLC

The built-in discrete item, SYSStatus, can be used to monitor the status of communications with the PLC. This item is set to:

- 0 (zero) when communications with the PLC fails.
- 1 (one) when communications is successful.

Note: For DDE/SuiteLink clients, SYSStatus always comes from the leaf level of a Communication Driver hierarchy branch, which is the destination PLC node. For OPC clients, SYSStatus can be accessed at all hierarchy levels. SYSStatus at the root level of the whole hierarchy tree is always good, as it represents the quality status of the local computer itself. Hence, for practical application, OPC clients should reference SYSStatus at any hierarchy levels other than the root.

Enter the following DDE reference formula in the appropriate place in your client:

=DASABTCP|ABPLC!SYSStatus

where:

DASABTCP is the name of the Communication Driver application.

ABPLC is the exact device group defined in the Communication Driver for the PLC.

SYSStatus is the discrete item used to monitor the status of connectivity with the PLC.

Enter the following OPC item reference syntax when adding the item in your OPC client:

YourOPCAccessPath.SYSStatus

where:

YourOPCAccessPath is the assembly of hierarchy node names leading to a specific controller device.

SYSStatus is the discrete item used to monitor the status of connectivity with the controller device.

Note: In the case of a PLC disconnect, the Communication Driver will retry three times before entering into slow poll mode. In the case of reply time-out, the Communication Driver will go into slow poll mode immediately.

Monitoring the Status of Communication Driver Conversations

The InTouch WindowViewer supports built-in topic names, called DDEStatus and IOStatus, that can be used to monitor the status of specific Communication Driver conversations. For example, let us assume that WindowViewer (VIEW) is communicating with the ABTCP Communication Driver to a PLC that has been defined in the Communication Driver with the topic name ABPLC.

The discrete items, DDEStatus and IOStatus, are set to 0 (zero) when this Communication Driver conversation failed, and to 1 (one) when this Communication Driver conversation is successful.

Using DDEStatus and IOStatus in Excel

The status of communications between the PLC and InTouch can be read into Excel by entering the following DDE reference formula in a cell on a spreadsheet:

=view|DDEStatus!ABPLC

or

=view|IOStatus!ABPLC

where:

viewis the name of the InTouch application.

[DDE][IO] Status is the built-in topic name used to monitor the status of communications between the Communication Driver and InTouch.

ABPLC is the exact topic name defined in the Communication Driver for the PLC.

Reading Values from the Communication Driver into Excel

Values may be read directly into Excel spreadsheets from the Communication Driver by entering a DDE formula into a cell using the following format:

=applicationname|topicname!itemname

Example formula:

=DASABTCP|ABPLC!'N7:0'

where:

DASABTCP is the name of the Communication Driver application.

ABPLCis the exact topic name defined in the Communication Driver for the PLC.

N7:0Is the actual location in the PLC that contains the data value. This is the item name.

In this example, each time the value of N7:0 changes in the PLC, the Communication Driver will automatically send the new value to the cell containing the formula in Excel.

Note: Refer to the Microsoft Excel manual for complete details on entering Remote Reference formulas for cells.

Writing Values to the Communication Driver from Excel

Values may be written to the Communication Driver from Microsoft Excel by creating an Excel macro that uses the POKE command. The proper command is entered in Excel as follows:

channel=INITIATE("applicationname","topicname")

=POKE(channel,"itemname", Data_Reference)

=TERMINATE (channel)

=RETURN()

The following describes each of the above **POKE** macro statements:

channel=INITIATE("applicationname","topicname")

- Opens a channel to a specific topic name (defined in the Communication Driver) in a particular application name (the executable name less the .exe).
- Assigns the number of that opened channel to **channel**.

Note: By using the **channel=INITIATE** statement, the word channel must be used in the **=POKE** statement instead of the actual cell reference. The **"application name"** and **"topic name"** portions of the formula must be enclosed in quotation marks.

=POKE(channel,"itemname", Data_Reference)

- **POKEs** the value contained in the **Data_Reference** to the specified item name (actual location in the PLC), via the **channel** number returned by the previously executed **INITIATE** function.
- **Data_Reference** is the row/column ID of the cell containing the data value.

=TERMINATE(channel)

- Closes the channel at the end of the macro.
- Some applications have a limited number of channels; therefore, they should be closed when finished.
- **Channel** is the channel number returned by the previously executed **INITIATE** function.

=RETURN()

- Marks the end of the macro.

Note: Refer to the **.XLM** sample Excel poke macro provided on the Communication Driver installation media. Also refer to the Microsoft Excel manual for complete details on entering Remote Reference formulas for cells.

Error Messages

Generic Communication Driver error messages, ABTCP-Communication Driver-specific error messages, and controller-specific error messages are supported. These error messages are sent to the Log Viewer. Use the Log Flag data to customize the type of messages logged to the Log Viewer, and see the Log Viewer documentation for more information about using log flags.

To troubleshoot Communication Driver problems, use the following error messages together with the OI Server Manager Diagnostics root data.

ABTCP Communication Driver Error Messages

The following table lists all the generic-Communication Driver and ABTCP-Communication Driver-specific error messages.

Error Message	Explanation	Possible Cause	Solution
PLC connection attempt timed out, closing socket at <Time unit in Msec>	The TCPIP socket was closed due to time-out when attempted to connect to the PLC.	1. The network has not been set up correctly or is having a problem. 2. The PLC is not set up or is having a problem.	1. Make sure the network is set up and running correctly. 2. Make sure the PLC is set up and running correctly.
PLC message timed out, revoking message <Message ID> at <Time unit in Msec>	The message was revoked due to time-out while waiting for a response from the PLC.	1. The network communications is having a problem. 2. The PLC is having a problem communicating.	1. Make sure the network is functioning. 2. Make sure the PLC is functioning.
runt Unsolicited msg packet received on <Host IP Address>	Invalid unsolicited message packet was received.	The PLC is not set up correctly.	Set up the PLC for unsolicited message communications.
Maximum number of socket <MAX> exceeded	Exceeded the maximum number of TCPIP sockets allowed.	The maximum number of sockets allowed is exceeded.	Free socket usage.
Error encountered initializing Unsolicited Data Port. No direct (i.e.: peer-to-peer) unsolicited data will be accepted.	Failed to create a TCPIP socket for the "peer-to-peer" unsolicited data used. Result: no unsolicited data can be accepted.	1. The network communications is having a problem. 2. The PLC is having a problem communicating.	1. Make sure the network is functioning. 2. Make sure the PLC is functioning.
WSAStartup() returned <Error Code>	Error code returned by the failing WinSocket Initialization routine.	Failed to initialize a Window Socket in the system, due to an internal system error.	Check the returned error code against the Microsoft error code list

Error Message	Explanation	Possible Cause	Solution
			to fix the WinSock initialization problem.
Winsock description: <Description string>	When the verbose flag is on, this message is displayed after the preceding WinSocket initialization failing message. This message contains a brief description for the error code returned.	More description for the preceding failing WinSocket initialization routine.	Check the description and fix the problem according to Microsoft's error list.
Winsock sys status: <Status string>	When the verbose flag is on, this message is displayed after the preceding WinSocket initialization failing message. This message contains a brief status string for the error code returned.	More description for the preceding failing WinSocket initialization routine.	Check the description and fix the problem according to Microsoft's error list.
Rejected <PLC Type> ITEM = <Item Name> on plc <PLC Name with hierarchy path>	The item requested by the client was rejected.	1. The requested item syntax was incorrect. 2. The hierarchy PLC name was incorrect.	1. Correct the syntax of the item. 2. Correct the hierarchy path for the PLC name.
A floating point value read for Item: [ITEM] on Topic: [TOPIC] was NOT A VALID NUMBER (Negative Infinity), therefore, it was converted to -3.4e38	A bad negative floating value was read from the PLC. The value was changed by the Communication Driver to a known good value before sending it to the client.	Bad data was read.	None.

Error Message	Explanation	Possible Cause	Solution
A floating point value read for Item: [ITEM] on Topic: [TOPIC] was NOT A VALID NUMBER (Positive Infinity), therefore, it was converted to +3.4e38	A bad positive floating value was read from the PLC. The Communication Driver changed the value to a known good value before sending it to the client.	Bad data was read.	None.
A floating point value read for Item: [ITEM] on Topic: [TOPIC] was NOT A VALID NUMBER (! NaN!), therefore, it was converted to +3.4e38	A bad positive floating value was read from the PLC. The Communication Driver changed the value to a known good value before sending it to the client.	Bad data was read.	None.
Could not find PLC node when creating message	Failed to find the PLC node from the hierarchy from the message.	When creating the message for the item, the Communication Driver cannot locate the PLC node. Possible corrupted or bad CFG file.	Check and correct the CFG file.
Response code [CmdNum# from reply] different from command code [CmdNum# from original request]	The response message's command code does not match the original command code.	This error indicates something is very wrong. Either the PLC is bad or the communications between the PLC and the computer is bad – in such a way that wrong data is passing between the Communication Driver and the PLC.	Make sure the Communication Driver is running correctly. Make sure the PLC is running correctly. Restart both the PLC and the Communication Driver if needed.

Error Message	Explanation	Possible Cause	Solution
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: An address field has an illegal value - check the Topic Configuration dialog, the block sizes may be too large or the PLC type may be incorrect.	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message was correctly formatted and sent. Make sure the PLC is configured correctly.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Not enough address fields specified - check the Topic Configuration dialog, the block sizes may be too large or the PLC type may be incorrect.	When doing this read or write operation to the item, the operation failed because an incorrect format in the message was found. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly formatted and sent. Make sure the PLC is configured correctly.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Too many address fields specified - check the Topic Configuration dialog, the block sizes may be too large or the PLC type may be incorrect	When doing this read or write operation to the item, the operation failed because of an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly formatted and sent. Make sure the PLC is configured correctly.

Error Message	Explanation	Possible Cause	Solution
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Addressed symbol not found - check the Topic Configuration dialog, the block sizes may be too large or the PLC type may be incorrect.	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly constructed and sent. Make sure the PLC is configured correctly.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Addressed symbol is > 8 or <= 0 length - check the Topic Configuration dialog, the block sizes may be too large or the PLC type may be incorrect.	When doing this read or write operation to the item, the operation failed because an incorrect format in the message was found. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly formatted and sent. Make sure the PLC is configured correctly.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: This address does not exist in PLC - check Data Table definition inside PLC and make sure ALL requested points are defined.	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly constructed and sent. Make sure the PLC is configured correctly.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: One or more points specified within this message do not exist in the PLC's Data Table - Please define, inside	When doing this read or write operation to the item, the operation failed because an incorrect format in the message was found. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly formatted and sent. Make sure the PLC is configured correctly.

Error Message	Explanation	Possible Cause	Solution
the PLC, the missing point(s).			
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Cannot complete request; the situation inside the PLC and/or bridge devices has changed since the command started.	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly constructed and sent. Make sure the PLC is configured correctly.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Data or File is too large - the packet the PLC is trying to send back is too large - check the Topic Configuration dialog.	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly formatted and sent. Make sure the PLC is configured correctly.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Transaction size too large for protocol - check the Topic Configuration dialog, the block sizes may be too large or the PLC type may be incorrect.	When doing this read or write operation to the item, the operation failed because of an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly formatted and sent. Make sure the PLC is configured correctly.

Error Message	Explanation	Possible Cause	Solution
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Access Denied; improper privilege - the PLC has refused us access to one or more of the data points in this message - check PRIVILEGE settings inside PLC.	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly formatted and sent. Make sure the PLC is configured correctly.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: A resource is not available - A programming terminal may be connected to (communicating with) this PLC - please disconnect the programming terminal.	When doing this read or write operation to the item, the operation failed because of an incorrect format in the message. The reason for failure is included in the message.	Write to read/write to a PLC that is already connected to a programming terminal.	Disconnect the programming terminal to the PLC and repeat the read/write operation.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Condition already exists; resource is already available - a loss of communications synchronization has occurred - if it continues, restart the Communication Driver.	When doing this read or write operation to the item, the operation failed because of an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly constructed and sent. If error continues, restart the Communication Driver and repeat the read/write operation.

Error Message	Explanation	Possible Cause	Solution
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Command cannot be executed - the PLC has refused to execute the command we have sent to it - check the PLC type in the Topic Configuration dialog.	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly formatted and sent. Make sure the PLC is configured correctly.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: No Access - the PLC has refused us access to one or more of the data points in this message - check PRIVILEGE settings inside the PLC.	When doing this read or write operation to the item, the operation failed because an incorrect format in the message was found. The reason for failure is included in the message.	Cannot access the PLC.	Make sure the PLC is configured correctly. Repeat the read/write operation after correcting the configuration.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: The data type requested (see FileType parameter) may not match the data type for the specified file number inside the PLC.	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly formatted and sent. Make sure the PLC is configured correctly.

Error Message	Explanation	Possible Cause	Solution
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: The PLC has rejected the format of our command - check the Topic Configuration dialog, the block sizes may be too large or the PLC type may be incorrect.	When doing this read or write operation to the item, the operation failed because of an incorrect format in message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly formatted and sent. Make sure the PLC is configured correctly.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: At least one word of the PLC's Data Table we are requesting with this message has been marked for deletion by a programming terminal - this is illegal.	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly constructed and sent. Make sure the PLC is configured correctly.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Data conversion error - check PLC configuration and PLC type selection inside the Topic Configuration dialog.	When doing this read or write operation to the item, the operation failed because of an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly formatted and sent. Make sure the PLC is configured correctly.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Remove I/O Scanner not able to communicate with 1771 rack adapter - check PLC for problems.	When doing this read or write operation to the item, the operation failed because an incorrect format in the message was found. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly constructed and sent. Make sure the PLC is configured correctly.

Error Message	Explanation	Possible Cause	Solution
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Remove I/O Adapter cannot communicate with module - check PLC for problems.	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly formatted and sent. Make sure the PLC is configured correctly.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: PLC's remote I/O scanner is unable to update the requested data because of a communications problem - check PLC for problems.	When doing this read or write operation to the item, the operation failed because of an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly formatted and sent. Make sure the PLC is configured correctly.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Duplicated Label in Message and/or PLC - you should check PLC setup and Topic Configuration for errors.	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly constructed and sent. Make sure the PLC is configured correctly.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: File is open; another node owns it - a programming terminal may connected to (communicating with) this PLC - please disconnect the programming terminal.	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Write to read/write to a PLC that is already connected to a programming terminal.	Disconnect the programming terminal to the PLC and repeat the read/write operation.

Error Message	Explanation	Possible Cause	Solution
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Another node is the PLC program owner - a programming terminal may connected to (communicating with) this PLC - please disconnect the programming terminal.	When doing this read or write operation to the item, the operation failed because an incorrect format in the message was found. The reason for failure is included in the message.	Write to read/write to a PLC that is already connected to a programming terminal.	Disconnect the programming terminal to the PLC and repeat the read/write operation.
ERROR - When topic: [TopicName] tried to Read/Write [DESC of the item] encountered the following error: Unknown Communication Error	When doing this read or write operation to the item, the operation failed because of an incorrect format in the message. The reason for failure is included in the message.	Bad request message was constructed and sent. This is a "catch-all" error message.	Make sure the request message is correctly formatted and sent. Make sure the PLC is configured correctly.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Destination node (PLC) is out of buffer space - Therefore, the PLC is refusing to accept our messages	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Cannot communicate with the PLC correctly.	Make sure the PLC is configured and running correctly and successfully. Make sure the PLC is on line and is communicating with the Communication Driver.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Remote node is not acknowledging (ACK) our requests for data - Is the PLC on-line? Is	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Cannot communicate with the PLC correctly.	Make sure the PLC is configured and running correctly and successfully. Make sure the PLC is on line and is communicating with the Communication Driver.

Error Message	Explanation	Possible Cause	Solution
the bridge device on-line?			
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Duplicate token holder detected - two stations on the highway have the same address (this is illegal)	When doing this read or write operation to the item, the operation failed because an incorrect format in the message was found. The reason for failure is included in the message.	Cannot communicate with the PLC correctly.	Make sure the PLC is configured and running correctly and successfully. Make sure the PLC is on line and is communicating with the Communication Driver.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Local port is disconnected - we may not be connected (physically) to the highway - check the cabling (wired incorrectly?) and _ALL_ connectors (loose?)	When doing this read or write operation to the item, the operation failed because of an incorrect format in the message. The reason for failure is included in the message.	Cannot communicate with the PLC correctly.	Make sure the PLC is configured and running correctly and successfully. Make sure the PLC is on line and is communicating with the Communication Driver. Make sure cabling is correctly connected.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Application	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason	Cannot communicate with the PLC correctly.	Make sure the PLC is configured and running correctly and successfully. Make sure the PLC is on line and is communicating with

Error Message	Explanation	Possible Cause	Solution
layer timed out waiting for a response	for failure is included in the message.		the Communication Driver.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Duplicate node detected - two stations on the highway have the same address (this is illegal)	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Cannot communicate with the PLC correctly.	Make sure the PLC is configured and running correctly and successfully. Make sure the PLC is on line and is communicating with the Communication Driver.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Station is off-line - the PLC we are attempting to contact is not on-line	When doing this read or write operation to the item, the operation failed because of an incorrect format in the message. The reason for failure is included in the message.	Cannot communicate with the PLC correctly.	Make sure all PLCs are configured and running correctly and successfully. Make sure the PLC is on line and is communicating with the Communication Driver.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Hardware fault - a hardware problem has prevented messages from being sent	When doing this read or write operation to the item, the operation failed because the message was incorrectly formatted. The reason for failure is included in the message.	Cannot communicate with the PLC correctly.	Make sure all PLCs are configured and running correctly and successfully. Make sure the PLC is on line and is communicating with the Communication Driver. Make sure all the hardware is running and functioning correctly.

Error Message	Explanation	Possible Cause	Solution
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Unknown Communication Error	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Cannot communicate with the PLC correctly.	Make sure all PLCs are configured and running correctly and successfully. Make sure the PLC is on line and is communicating with the Communication Driver. Make sure all the hardware is running and functioning correctly.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Illegal command or format - check the Topic Configuration dialog, the block sizes may be too large, or the wrong PLC type may be selected	When doing this read or write operation to the item, the operation failed because the message was incorrectly formatted. The reason for failure is included in the message.	Bad request message was constructed and sent.	Make sure the request message is correctly formatted and sent. Make sure the PLC is configured correctly.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Remote host will not communicate - PLC may not be on-line	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Cannot communicate with the remote PLC correctly.	Make sure all PLCs and all remote devices are configured and running correctly and successfully.

Error Message	Explanation	Possible Cause	Solution
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Remote host missing or down - PLC may not be on-line	When doing this read or write operation to the item, the operation failed because of an incorrect format in the message. The reason for failure is included in the message.	Cannot communicate with the remote PLC correctly.	Make sure all PLCs and all remote devices are configured and running correctly and successfully.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Remote host hardware fault - PLC has experienced a hardware fault	When doing this read or write operation to the item, the operation failed because an incorrect format in the message was found. The reason for failure is included in the message.	Cannot communicate with the remote PLC correctly.	Make sure all PLCs and all remote devices are configured and running correctly and successfully. Make sure all the hardware is working correctly.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Addressing problem or memory protection rungs (inside the PLC) have prevented us from retrieving the requested information	When doing this read or write operation to the item, the operation failed because the message was incorrectly formatted. The reason for failure is included in the message.	Cannot communicate with the remote PLC correctly.	Make sure all PLCs and all remote devices are configured and running correctly and successfully. Make sure all the hardware is working correctly.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Function disallowed due to command protection inside the PLC, or the position of a hardware key on the PLC	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Cannot communicate with the remote PLC correctly.	Make sure all PLCs and all remote devices are configured and running correctly and successfully. Make sure all the hardware is working correctly.

Error Message	Explanation	Possible Cause	Solution
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: PLC processor is in PROGRAM mode - it must be in RUN mode for us to be able to gather data	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Cannot communicate with the remote PLC correctly.	Make sure all the PLCs and remote devices are configured and running correctly and successfully. Make sure all the hardware is working correctly. Make sure the PLC is not in the program mode, but rather is in the RUN mode.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Compatibility mode file missing or communication zone problem inside the PLC	When doing this read or write operation to the item, the operation failed because an incorrect format in the message was found. The reason for failure is included in the message.	Cannot communicate with the remote PLC correctly.	Make sure all the PLCs and remote devices are configured and running correctly and successfully. Make sure all the hardware is working correctly.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Remote node (PLC or Bridge Device) cannot buffer the command we are trying to send it	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Cannot communicate with the remote PLC correctly.	Make sure all the PLCs and remote devices are configured and running correctly and successfully. Make sure all the hardware is working correctly. Make sure the correct message to send to the remote PLC has been constructed.

Error Message	Explanation	Possible Cause	Solution
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Remote node (PLC or Bride Device) problem due to download - a download is most likely in progress, please wait until it is complete	When doing this read or write operation to the item, the operation failed because the message was incorrectly formatted. The reason for failure is included in the message.	Cannot communicate with the remote PLC correctly.	Make sure all PLCs and all remote devices are configured and running correctly and successfully. Make sure all the hardware is working correctly. Repeat the read/write operation after the download operation is completed.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Cannot execute command due to active IPBs inside the PLC	When doing this read or write operation to the item, the operation failed due to an incorrect format in the message. The reason for failure is included in the message.	Cannot communicate with the remote PLC correctly.	Make sure all PLCs and all remote devices are configured and running correctly and successfully. Make sure all the hardware is working correctly.
Error - When topic [TopicName] tried to Read/Write [DESC of the item] it encountered the following error:STS=[sts error code]: Cannot execute command due to active IPBs inside the PLC	When doing this read or write operation to the item, the operation failed because of an incorrect format in the message. The reason for failure is included in the message.	Cannot communicate with the remote PLC correctly.	Make sure all the PLCs and remote devices are configured and running correctly and successfully. Make sure all the hardware is working correctly.
recv() for [HostName] on port [PortNumber] failed	Failed to read from the Window Socket specified.	Failed to read from the Winsock.	Repeat the operation by restarting the Communication Driver.
recd packet from [HostName] too big on port [PortNumber] ([#of bytes received] bytes)	The received packet from the PLC exceeds the maximum packet size allowed for this type of protocol.	Incorrect data packet was read from the Socket.	Repeat the operation by restarting the Communication Driver.

Error Message	Explanation	Possible Cause	Solution
connect() for [HostName] on port [PortNumber] refused	PLC has refused to make the connection via the WinSock.	Failed to make the connection with the PLC.	Reset the PLC and/or the Communication Driver and try again.
connect() for [HostName] on port [PortNumber] failed	Failed to make the connection with the PLC.	Failed to make the connection with the PLC.	Reset the PLC and/or the Communication Driver and try again.
attempt to resolve remote hostname [HostName] failed	Failed to resolve the HostName.	The HostName cannot be translated to a valid IP address internally.	Use a correct HostName.
A PLC (IP: [IPAddress]) attempted to send us an unsolicited data packet. But the maximum number of simultaneous unsolicited data connections [MAX socket] has already been reached. Data packet ignored.	The maximum number of sockets used for unsolicited data communications was reached. No more unsolicited data package will be accepted.	The maximum number of sockets used for unsolicited data communications was reached. No more unsolicited data package will be accepted.	Decrease the number of unsolicited data to communicate to the socket.
ABTCPAcceptedSocket::Initialize unable to associate an event with a handle	Unable to associate the event with a valid handle within the internal state computer.	Software internal error.	Restart the Communication Driver and try again.
Timeout waiting for initialization packet from PLC on an unsolicited data port connected to [HostName]	Time-out occurred while waiting for unsolicited data header from a PLC.	Failed to receive unsolicited data from a PLC.	Make sure the PLC is configured to send out unsolicited data correctly. Make sure the Communication Driver is functioning correctly.
Timeout waiting for data packet from PLC on an unsolicited data port connected to [HostName]	Time-out occurred while waiting for unsolicited data packet from a PLC.	Failed to receive unsolicited data from a PLC.	Make sure the PLC is configured to send out unsolicited data correctly. Make sure the Communication Driver iscorrectly functional.
Timeout waiting for an unknown event from PLC on an unsolicited data port connected to [HostName]	Time-out occurred while waiting for unsolicited data packet from a PLC.	Failed to receive unsolicited data from a PLC.	Make sure the PLC is configured to send out unsolicited data correctly. Make sure the

Error Message	Explanation	Possible Cause	Solution
			Communication Driver is functioning correctly.
An error occurred attempting to complete the initialization of an unsolicited connection.	When the initialization portion of the unsolicited data was received, a Winsock error occurred.	Failed to receive the initialization portion of the unsolicited data packet from the PLC.	Make sure the PLC is configured to send out unsolicited data correctly. Make sure the Communication Driver is functioning correctly.
An error occurred attempting to receive the data portion of an unsolicited connection.	When the data portion of the unsolicited data was received, a Winsock error occurred.	Failed to receive the data portion of the unsolicited data packet from the PLC.	Make sure the PLC is configured to send out unsolicited data correctly. Make sure the Communication Driver is functioning correctly.
Minor Error: Winsock told us data was waiting for us, but when we checked it, the whole message wasn't there. Ignoring Winsock Message.	Winsock error occurred; ignore the event.	Winsock error occurred; ignore the event.	Ignore the event and the message.
State engine error. Received Unsol DATA packet, but socket was not waiting for one. Ignoring packet	Winsock error occurred; ignore the event.	Winsock error occurred; ignore the event.	Ignore the event and the message.

PLC-5 Error Messages

The error messages generated specifically for the PLC-5 family controllers are listed in the following table.

Error Message	Explanation	Possible Cause	Solution
item <ItemName> not valid, PLC does not have PID feature	PID feature is not supported for this PLC-5 configuration.	The PLC-5 configuration indicates that the PID feature is not supported.	Select the "Support PID" feature option for PLC-5, if the PLC supports the feature.
BINARY file number must be greater than 2	Incorrect format for this item. The Binary Item's File Number	The Binary Item's File Number must be 3 or larger.	Only access the Binary Item with File Number equal to 3 or larger.

Error Message	Explanation	Possible Cause	Solution
	was smaller than 3 for PLC-5.		
TIMER file number must be greater than 2	Incorrect format for the item. The Timer Item's File Number was smaller than 3 for PLC-5.	The Timer Item's File Number must be 3 or larger.	Only access the Timer Item with File Number equal to 3 or larger.
CONTROL file number must be greater than 2	Incorrect format for the item. The Control Item's File Number was smaller than 3 for PLC-5.	The Control Item's File Number must be 3 or larger.	Only access the Control Item with File Number equal to 3 or larger.
INTEGER file number must be greater than 2	Incorrect format for the item. The Integer Item's File Number was smaller than 3 for PLC-5.	The Integer Item's File Number must be 3 or larger.	Only access the Integer Item with File Number equal to 3 or larger.
BCD file number must be greater than 2	Incorrect format for the item. The BCD Item's File Number was smaller than 3 for PLC-5.	The BCD Item's File Number must be 3 or larger.	Only access the BCD Item with File Number equal to 3 or larger.
PD file number must be > 8	Incorrect format for the item. The PID Item's File Number was 8 or smaller for PLC-5.	The PID Item's File Number must be 9 or larger.	Only access the PID Item with File Number equal to 9 or larger.
ST file number must be > 8	Incorrect format for the Item. The ST item's File Number was 8 or smaller for PLC-5.	The ST Item's File Number must be 9 or larger.	Only access the ST Item with File Number equal to 9 or larger.
BT file number must be > 8	Incorrect format for the Item. The BT item's File Number was 8 or smaller for PLC-5.	The BT Item's File Number must be 9 or larger.	Only access the BT Item with File Number equal to 9 or larger.

Error Message	Explanation	Possible Cause	Solution
SC file number must be > 4	Incorrect format for the Item. The SC item's File Number was 4 or smaller for PLC-5.	The ST Item's File Number must be 5 or larger.	Only access the ST Item with File Number equal to 5 or larger.
Cannot write to file BT[FileNumber]	Failed to write to a BT item for PLC-5.	For PLC-5, write operation to a BT item is not permitted.	Do not attempt to write to a BT item for PLC-5.

SLC 500 Error Messages

The following table lists all the SLC-500-controller-specific error messages.

Error Message	Explanation	Possible Cause	Solution
BINARY file number must be 3 or 9-255	Incorrect format for this item. The Binary Item's File Number was not 3 or 9-255 for SLC 500.	Binary Item's File Number must be 3 or 9-255.	Only access Binary Item with valid File Number.
TIMER file number must be 4 or 9-255	Incorrect format for the item. The Timer Item's File Number was not 4 or 9-255 for SLC 500.	Timer Item's File Number must be 4 or 9-255.	Only access Timer Item with valid File Number.
COUNTER file number must be greater than 5 or 9-255	Incorrect format for the item. The Counter Item's File Number was not 5 or 9-255 for SLC 500.	Counter Item's File Number must be 5 or 9-255.	Only access Counter Item with valid File Number.
CONTROL file number must be greater than 6 or 9-255	Incorrect format for the item. The Control Item's File Number was not 6 or 9-255 for SLC 500.	The Control Item's File Number must be 6 or 9-255.	Only access the Control Item with the valid File Number.
INTEGER file number must be 7 or 9-255	Incorrect format for the item. The Integer Item's File Number was not 7 or 9-255 for SLC 500.	The Integer Item's File Number must be 7 or 9-255.	Only access the Integer Item with the valid File Number.
FLOAT file number must be 8 or 8-255	Incorrect format for the item. The Float Item's File Number was not 8 or 8-255 for SLC 500.	The Float Item's File Number must be 8 or 8-255.	Only access the Float Item with the valid File Number.

PLC-5 and SLC 500 Error Messages

The error messages listed in the following table pertain to both the PLC-5 and SLC 500 controllers.

Error Message	Explanation	Possible Cause	Solution
File numbers must be between 0 and 999	For PLC-5 and SLC 500: Incorrect format for the Item. The Item's File Number was out of range.	A bad item File Number was used.	Use a valid range for the Item's File Number.
OUTPUT file number must be 0	For PLC-5 and SLC 500: Incorrect format for the item. The Output Item's File Number was not 0.	The Output Item's File Number must be 0.	Only access the Output Item with a File Number equaling to 0.
INPUT file number must be 1	For PLC-5 and SLC 500: Incorrect format for the item. The Input Item's File Number was not 1.	The Input Item's File Number must be 1.	Only access the Input Item with a File Number that equals to 1.
STATUS file number must be 2	For PLC-5 and SLC 500: Incorrect format for the item. The Status Item's File Number was not 2.	The Status Item's File Number must be 2.	Only access the Status Item with a File Number equaling to 2.
FLOATING POINT file number must be greater than 2	For PLC-5 and SLC 500: Incorrect format for the item. The Floating Point Item's File Number was smaller than 3.	The Floating Point Item's File Number must be 3 or larger.	Only access the Floating Point Item with a File Number that equals to 3 or larger.
FLOATING POINT file cannot have bit number	For PLC-5 and SLC 500: Incorrect format for the item. The Floating Point Item contained a bit number field.	The Floating Point Item must not contain a bit number field.	Only access the Floating Point Item without a bit number field.
ASCII file number must be greater than 2	For PLC-5 and SLC 500: Incorrect format for the item. The ASCII Item's File Number was smaller than 3.	The ASCII Item's File Number must be 3 or larger.	Only access the ASCII Item with a File Number equaling to 3 or larger.