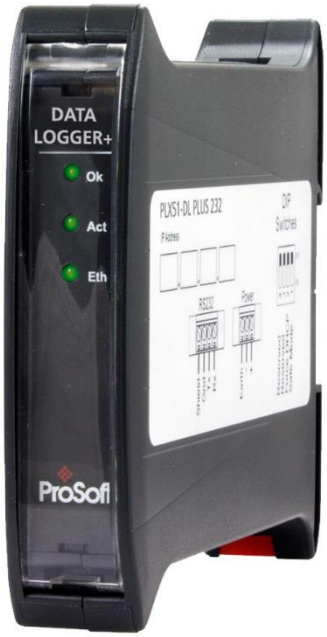
ProSoft TECHNOLOGY

Where Automation Connects.



PLX51-DLplus-232

Data Logger Plus

Data Storage Module

May 26, 2023

USER MANUAL

Your Feedback Please

We always want you to feel that you made the right decision to use our products. If you have suggestions, comments, compliments or complaints about our products, documentation, or support, please write or call us.

ProSoft Technology, Inc.
+1 (661) 716-5100
+1 (661) 716-5101 (Fax)
www.prosoft-technology.com
support@prosoft-technology.com

PLX51-DLPlus-232 User Manual
For Public Use.

May 26, 2023

Content Disclaimer

This documentation is not intended as a substitute for and is not to be used for determining suitability or reliability of these products for specific user applications. It is the duty of any such user or integrator to perform the appropriate and complete risk analysis, evaluation and testing of the products with respect to the relevant specific application or use thereof. Neither ProSoft Technology nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information contained herein. Information in this document including illustrations, specifications and dimensions may contain technical inaccuracies or typographical errors. ProSoft Technology makes no warranty or representation as to its accuracy and assumes no liability for and reserves the right to correct such inaccuracies or errors at any time without notice. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.
No part of this document may be reproduced in any form or by any means, electronic or mechanical, including photocopying, without express written permission of ProSoft Technology. All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components. When devices are used for applications with technical safety requirements, the relevant instructions must be followed. Failure to use ProSoft Technology software or approved software with our hardware products may result in injury, harm, or improper operating results. Failure to observe this information can result in injury or equipment damage.

For professional users in the European Union
If you wish to discard electrical and electronic equipment (EEE), please contact your dealer or supplier for further information.

Warning – Cancer and Reproductive Harm – www.P65Warnings.ca.gov

Agency Approvals and Certifications

Please visit our website: www.prosoft-technology.com

Contents

1 Preface

1.1 Introduction to the PLX51-DLplus-232

This manual describes the installation, configuration, operation, and diagnostics of the PLX51-DLplus-232. The PLX51-DLplus-232 can read and store data from Logix Controllers, DF1 Serial Interfaces, or Modbus devices. The PLX51-DLplus-232 has the capacity to store over 16 million records in its solid-state non-volatile memory. Each stored record includes a Date Time stamp with a 50 ms resolution, Tag Name, Data Type, and Value.

The PLX51-DLplus-232 can be used to log data at a remote site with limited communication with its base. The PLX51-DLplus-232 is also used to store records on mobile equipment such as trucks, drilling rigs, or snow plows. Once the equipment returns back to its base, the historical data can be uploaded and transferred to a more permanent storage device. The PLX51-DLplus-232 can also be configured to collect data which is only downloaded and examined if a fault occurs, otherwise the data is overwritten.

Compared to the **PLX51-DL-232**, the **PLX51-DLplus-232** has the following additional features:

- Trend up to five variables dynamically via the integrated webserver.
- Upload logged data directly from the integrated webserver.
- REST API support retrieves data automatically via JSON-encoded messages.

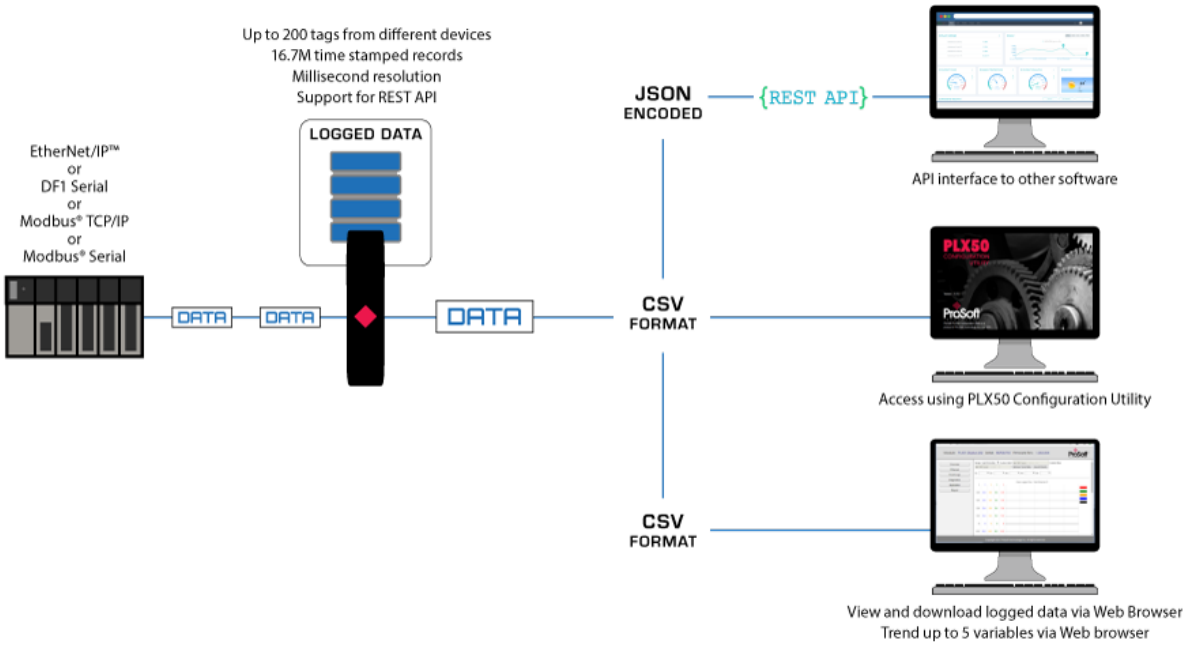


Figure 1.1 – Non-Historian Option

1.2 Features

The PLX51-DLplus-232 provides temporary extensive on-board storage capability for storing process tags. A total of 16,777,216 records can be stored in its non-volatile memory.

Each record consists of the following data:

Table 1.1 - Components of a Record

Parameter	Link
Date Time	UTC Time includes: Year, Month, Day, Hour, Minute, Second, Milliseconds. Time has a resolution of 50 milliseconds.
Tag Name	As defined in Controller or in the PLX50 Configuration Utility for other sources
Data Type	BOOL, SINT, INT, DINT, or REAL
Value	Logix Tag / DF1 File / Modbus Register value

The Log Index is managed by the PLX51-DLplus-232 and incremented each time a new record is stored. The Unload Index is managed externally by the unload service. It is only incremented after a record has been logged successfully to a text file. The records can be unloaded in Logix with the Example Code. Both the Log Index and Unload Indices loop around, eventually reaching the end of the cache. The cache becomes 100% full when the Log Index loops around and equals the Unload Index. In this situation, either older records are overwritten (Log Mode = Overwrite) or newer records are not logged (Log Mode = Hold).

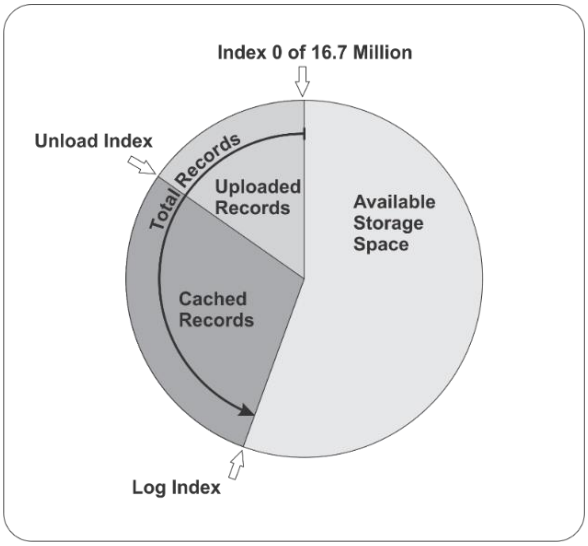


Figure 1.2 - Memory Schematic

The PLX51-DLplus-232 is configured using the ProSoft PLX50 Configuration Utility. This program can be downloaded from www.prosoft-technology.com, free of charge. The PLX50 Configuration Utility offers various configuration methods, including a controller tag browser. The PLX50 Configuration Utility can also be used to monitor the status and download historical data to a local file.

The PLX51-DLplus-232 can operate in both a Logix “owned” and standalone mode. With a Logix connection, the input and output assemblies provide additional diagnostics information. This information is available in the Logix controller environment.

The PLX51-DLplus-232 uses isolated RS232 for DF1 communication. The RS232 port also uses a terminal block for convenient installation.

A built-in webserver provides detailed diagnostics of system configuration and operation.

1.3 Additional Information

The following documents contain additional information that can assist you with installation and operation.

Table 1.2 - Additional Information

Resource	Link
PLX50 Configuration Utility Installation	www.prosoft-technology.com
User Manual Datasheet Example Code & UDTs	www.prosoft-technology.com
Ethernet wiring standard	www.cisco.com/c/en/us/td/docs/video/cds/cde/cde205_220_420/installation/guide/cde205_220_420_hig/Connectors.html
CIP Routing	The CIP Networks Library, Volume 1, Appendix C:Data Management

2 Installation

2.1 Module Layout

The PLX51-DLplus-232 has three ports at the bottom of the enclosure, as shown in the figure below. The ports are used for Ethernet, RS232 serial, and power.

The DC power port uses a three-way connector (**+** positive, **-** negative, and **Earth**).

The RS232 port uses a four-way connector (**Tx** Transmit, **Rx** Receive, **Gnd** Ground, and **Shield** earth connection).

The Ethernet cable must be wired according to industry standards which can be found in the additional information section of this document.

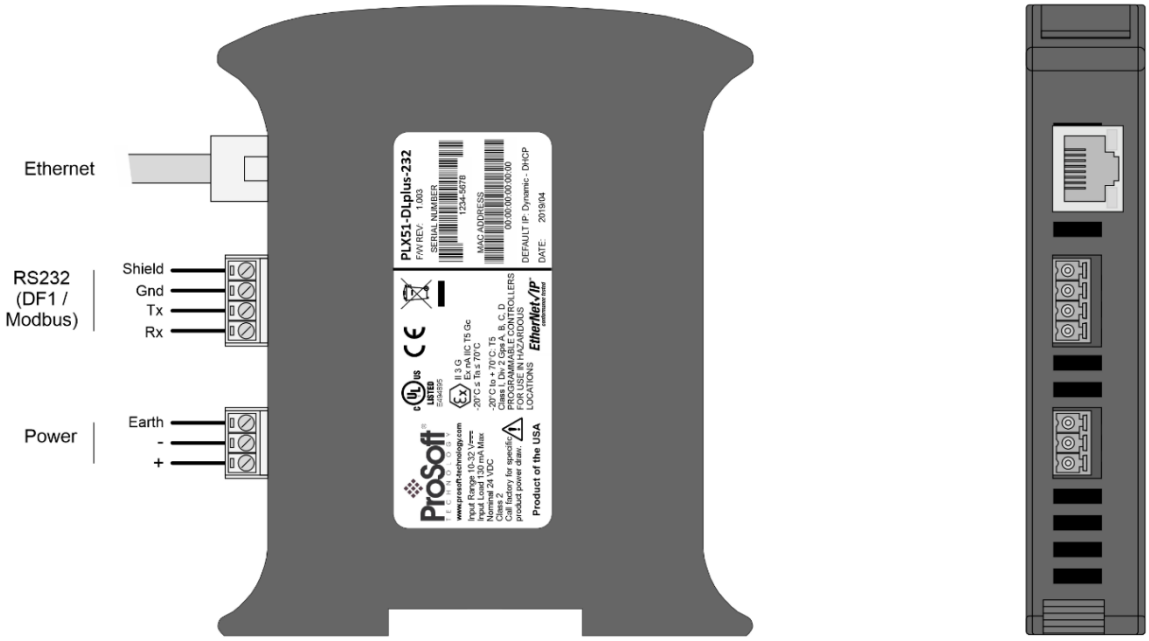


Figure 2.1 –Side and bottom view

The PLX51-DLplus-232 provides three diagnostic LEDs (**Ok**, **Act**, and **Eth**). These LEDs provide information on system operation, the Ethernet interface, and the auxiliary communication interface (RS232).

Figure 2.2 – Front and top view

The PLX51-DLplus-232 has four DIP switches at the top of the enclosure as shown above.

Table 2.1 - DIP Switch Settings

DIP Switch	Description
DIP 1	Used to force the PLX51-DLplus-232 into “Safe Mode”. When in “Safe Mode”, the PLX51-DLplus-232 does not load the application firmware. It waits for new firmware to be downloaded. This should only be used when a firmware update was interrupted at a critical stage.
DIP 2	Used to force the PLX51-DLplus-232 into DHCP mode, useful when the user has forgotten the IP address of the PLX51-DLplus-232. **Note:** If multiple network cards are running on your PC, the DHCP will be unreliable. Only one DHCP server should be used.
DIP 3	Used to lock the configuration from being overwritten by the PLX50 Configuration Utility. When set to ‘On’, the PLX50 Configuration Utility will not be able to download to the module.
DIP 4	When set to ‘On’, upon bootup the Ethernet IP address will be set to **192.168.1.100** and network mask **255.255.255.0**. The DIP switch can then be set to ‘Off’ to allow the assignment of a static IP address, if needed.

2.2 Module Mounting

The PLX51-DLplus-232 provides a DIN rail clip to mount onto a 35mm DIN rail.

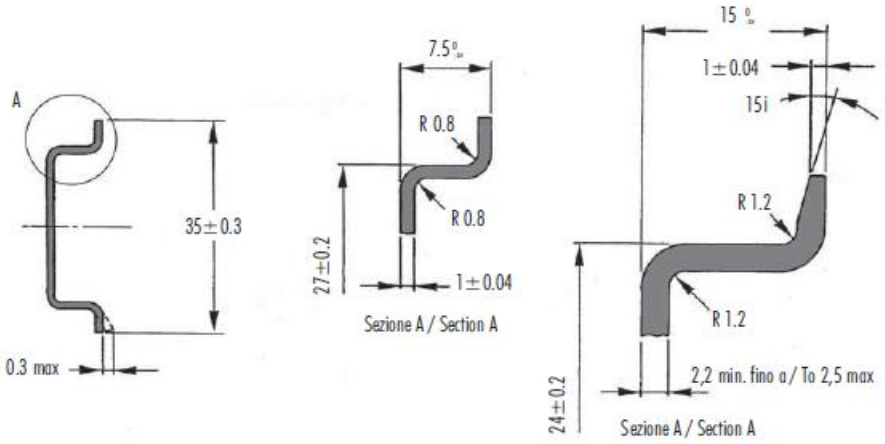


Figure 2.3 - DIN rail specification

The DIN rail clip is mounted on the bottom of the PLX51-DLplus-232. Use a flat screw driver to pull the clip downward. Once the PLX51-DLplus-232 is mounted onto the DIN rail, the clip must be pushed upward to lock the PLX51-DLplus-232 in place.

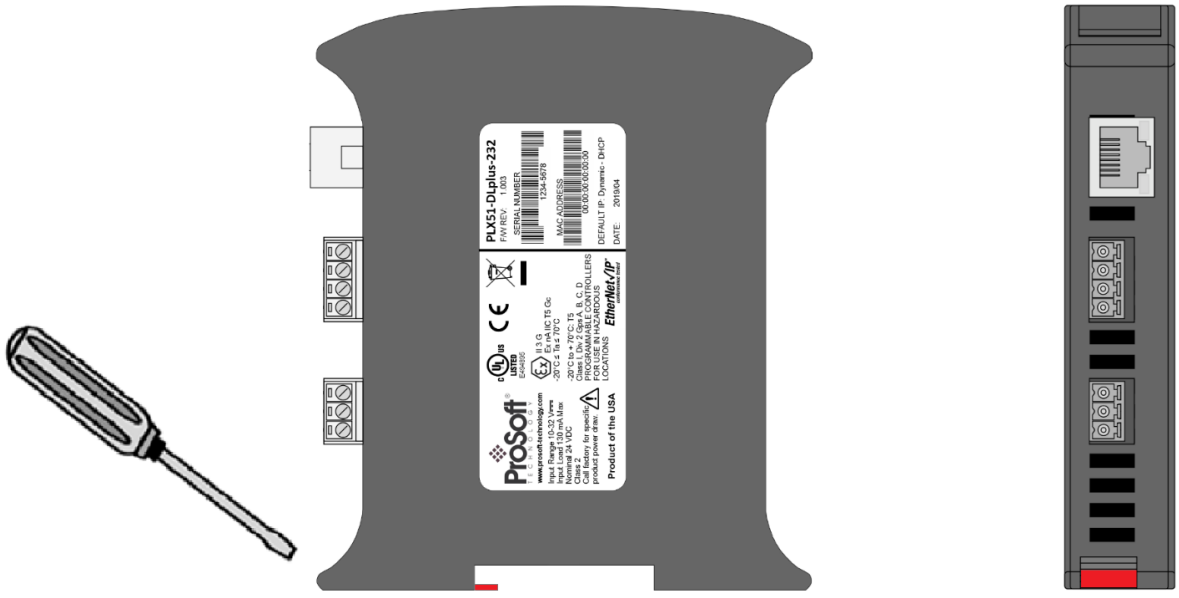


Figure 2.4 - DIN rail mouting

2.3 Power

A three-way power connector is used to connect + positive, - negative, and Earth. The PLX51-DLplus-232 requires an input voltage of 10 to 28 Vdc.

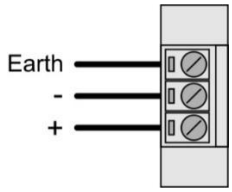


Figure 2.5 - Power connector

2.4 RS232 Port

The RS232 connector is used to connect the Transmit (Tx), Receive (Rx), and Ground conductors for serial communication. The shield terminal can be used for shielded cable in high noise environments.

Important: The shield of the RS232 port is internally connected to the power connector earth. When using a shield it is important to connect the Earth terminal on the power connector to a clean earth. Failing to do this can lower the signal quality of the RS232 communication.

Important: When using a shielded cable, it is important that only one end of the shield is connected to earth to avoid current loops. It is recommended to connect the shield to the PLX51-DLplus-232, and not to the other Serial device.

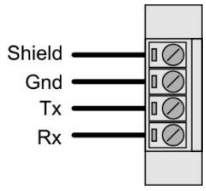


Figure 2.6 - RS232 connector

2.5 Ethernet Port

The Ethernet connector should be wired according to industry standards. Refer to the additional information section in this document for further details.

3 Setup

3.1 Install Configuration Software

The PLX51-DLplus-232 is configured using the PLX50 Configuration Utility environment. This software can be downloaded from www.prosoft-technology.com.

Figure 3.1 - PLX50 Configuration Utility Environment

3.2 Network Parameters

3.2.1 DHCP Server Settings

By default, the PLX51-DLplus-232 has DHCP (Dynamic Host Configuration Protocol) enabled. Thus, a DHCP server must be used to provide the PLX51-DLplus-232 with the required network parameters (IP address, subnet mask, etc.). There are a number of DHCP utilities available. However, it is recommended to use the DHCP server in the PLX50 Configuration Utility.

Note: If multiple network cards are running on your PC, the DHCP will be unreliable. Only one DHCP server should be used.

1 Within the PLX50 Configuration Utility, click on **TOOLS** > **DHCP SERVER**.

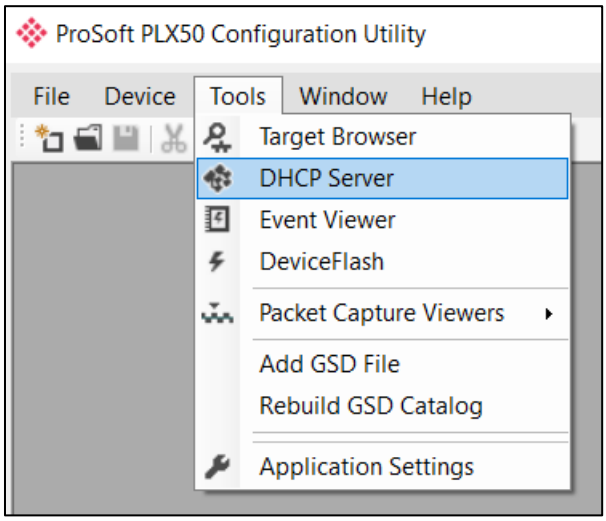


Figure 3.2 - Selecting DHCP Server

2 Once opened, the DHCP server listens on all available network adapters for DHCP requests and displays their corresponding MAC addresses.

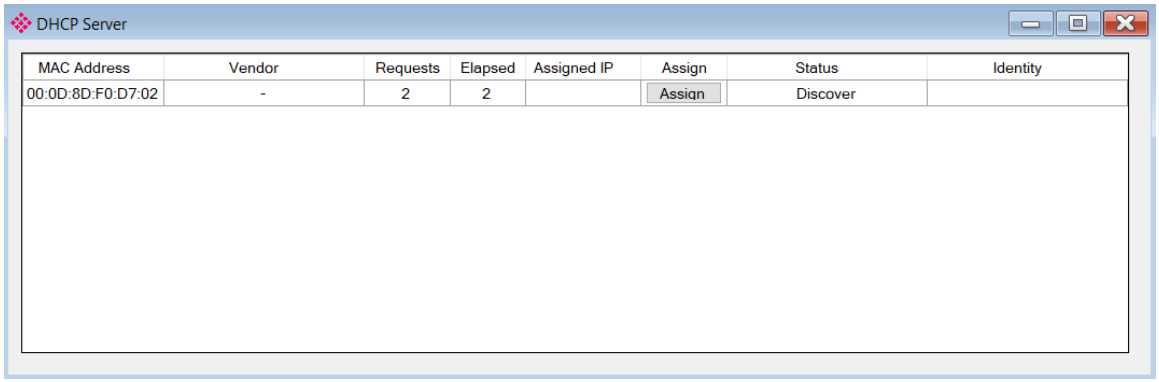


Figure 3.3 - DHCP Server

Note: If the DHCP requests are not displayed in the DHCP Server, it may be due to the local PC's firewall. During installation, the necessary firewall rules are automatically created for the Windows firewall. Another possibility is that another DHCP Server is operational on the network and it has assigned the IP address.

3 To assign an IP address, click on the corresponding **ASSIGN** button. The *Assign IP Address* dialog box opens.

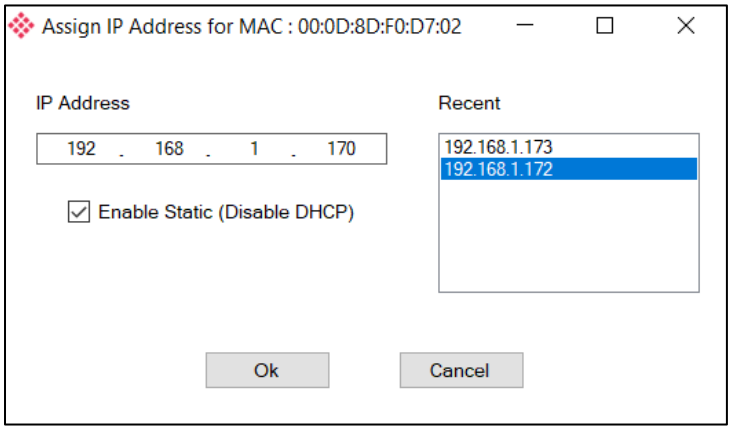


Figure 3.4 - Assigning IP Address

The required IP address can then be either entered, or a recently used IP address can be selected by clicking on an item in the *Recent* list.

If the *Enable Static* checkbox is checked, the IP address will be set to static after the IP assignment, thereby disabling future DHCP requests.

4 Click **OK** when complete.

5 Once the *Assign IP Address* dialog box has been accepted, the DHCP server automatically assigns the IP address to the PLX51-DLplus-232 and reads the *Identity Object Product* name from the device. The device indicates a green background upon successful assignment of the IP address.

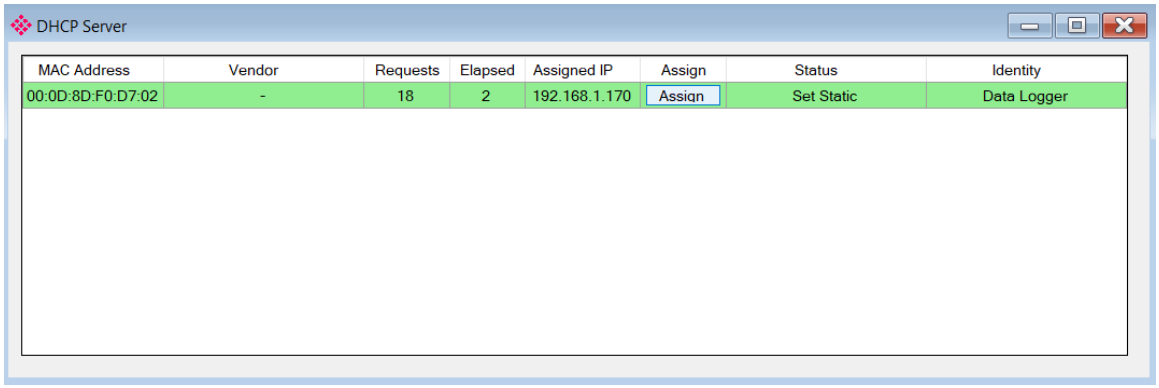


Figure 3.5 - Successful IP address assignment

It is possible to force the PLX51-DLplus-232 into DHCP mode by powering up the device with DIP switch 2 in the **On** position. A new IP address can be assigned by repeating the previous steps.

Note: It is important to return DIP switch 2 back to Off position, to avoid the PLX51-DLplus-232 returning to a DHCP mode after the power is cycled again.

In addition to the setting the IP address, other network parameters can be set during the DHCP process. These settings can be viewed and edited by clicking on **TOOLS** > **APPLICATION SETTINGS**.

3.2.2 Network Settings

Once the DHCP process has been completed, the network settings can be set using the *Ethernet Port Configuration* via the *Target Browser*.

1 Click on Tools > Target Browser

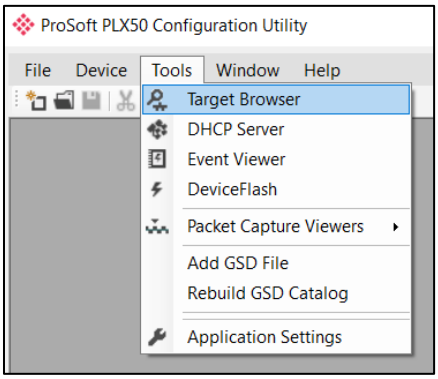


Figure 3.6 - Selecting the Target Browser

2 The *Target Browser* automatically scans the Ethernet network for EtherNet/IP devices.

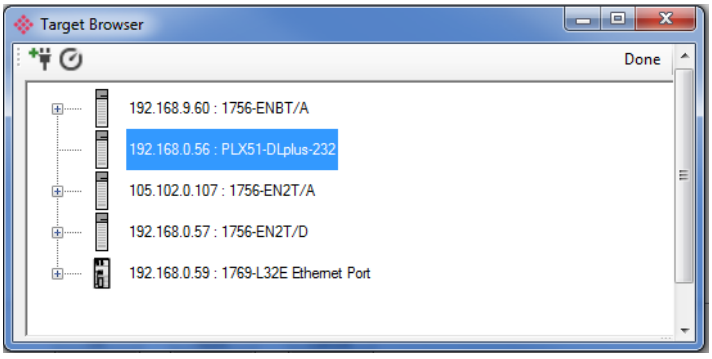


Figure 3.7 - Target Browser

3 Right-clicking on a device reveals the context menu, including the *Port Configuration* option.

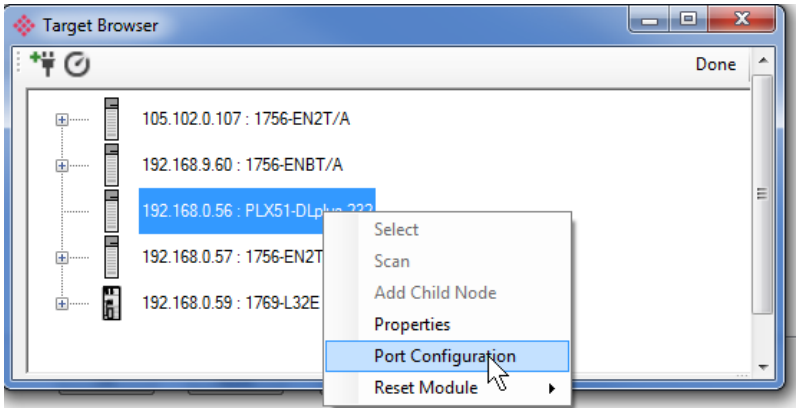


Figure 3.8 - Selecting Port Configuration

4 In the *Port Configuration* dialog box, enter the Ethernet port configuration parameters.

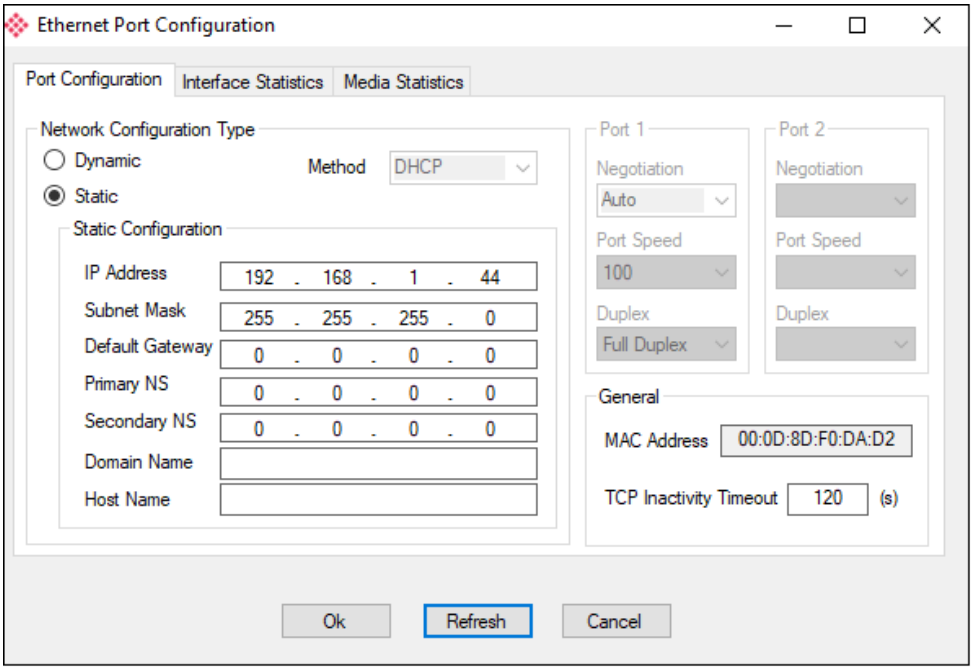


Figure 3.9 – Ethernet Port Configuration

3.3 Creating a New Project

Before you configure the PLX51-DLplus-232, a new PLX50 Configuration Utility project must be created.

1 Click on **File** > **New**.

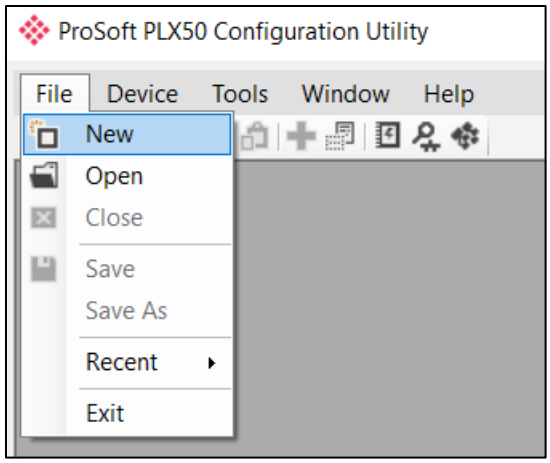


Figure 3.10 - Creating a new project

2 A new project is created and displayed in the Project Explorer tree view.

3 Add a new device by clicking on **DEVICE** > **ADD**.

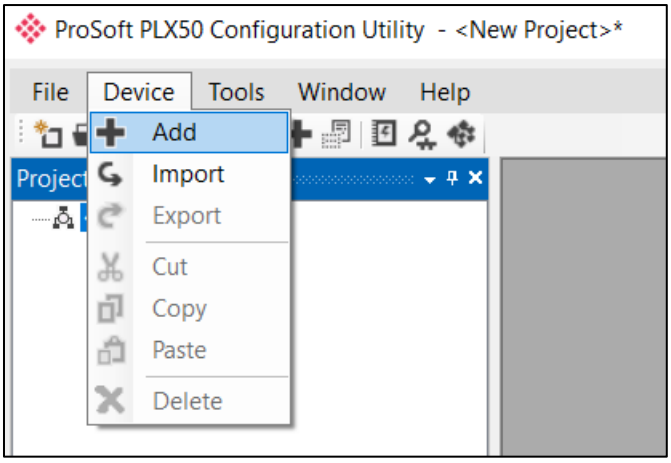


Figure 3.11 - Adding a new device

4 In the *Add New Device* dialog box, select the PLX51-DLplus-232 and click the **Ok** button.

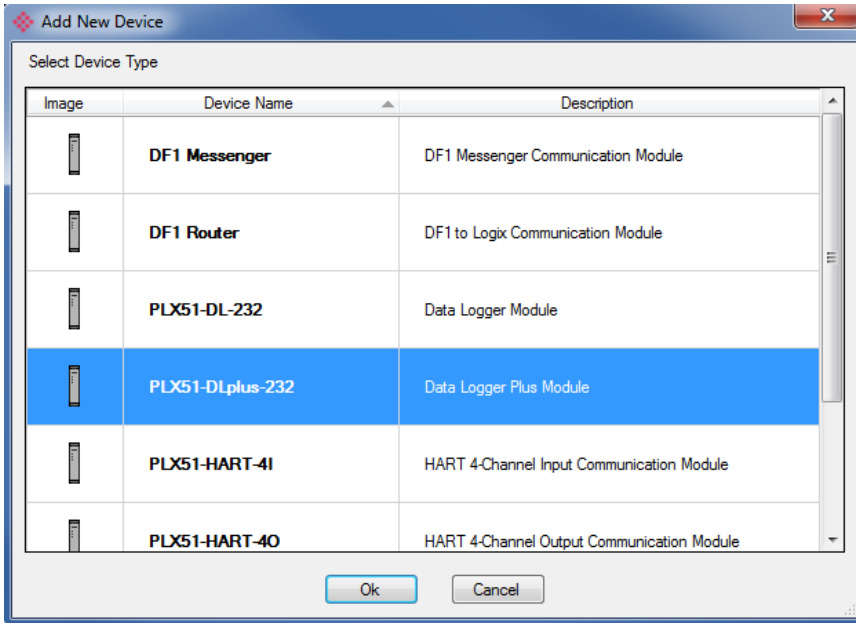


Figure 3.12 – Selecting the PLX51-DLplus-232

5 The device appears in the Project Explorer tree, and its configuration window is opened. The device configuration window can also be opened by double-clicking the PLX51-DLplus-232 icon in the Project Explorer tree, or right-clicking the PLX51-DLplus-232 icon and selecting *Configuration*.

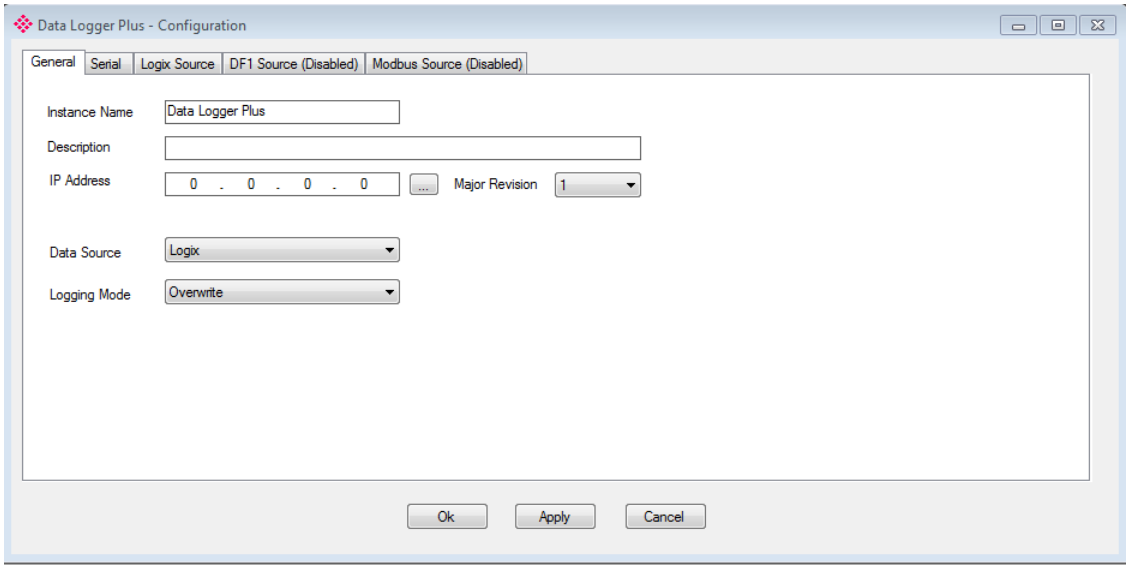


Figure 3.13 – PLX51-DLplus-232 configuration

3.4 Configuring the PLX51-DLplus-232

The PLX51-DLplus-232 is configured by the PLX50 Configuration Utility. The configuration consists of a general configuration, serial configuration for DF1 or Modbus RTU, data source configuration, and tag selections. The PLX51-DLplus-232 configuration is saved in non-volatile memory that persists when the PLX51-DLplus-232 is powered down.

Important: When a firmware upgrade is performed, the PLX51-DLplus-232 will clear all configuration and cached records.

3.4.1 General Tab

The general configuration is shown in the figure below. The general configuration window is opened by either double-clicking on the PLX51-DLplus-232 icon in the tree, or right-clicking the PLX51-DLplus-232 icon and selecting *Configuration*.

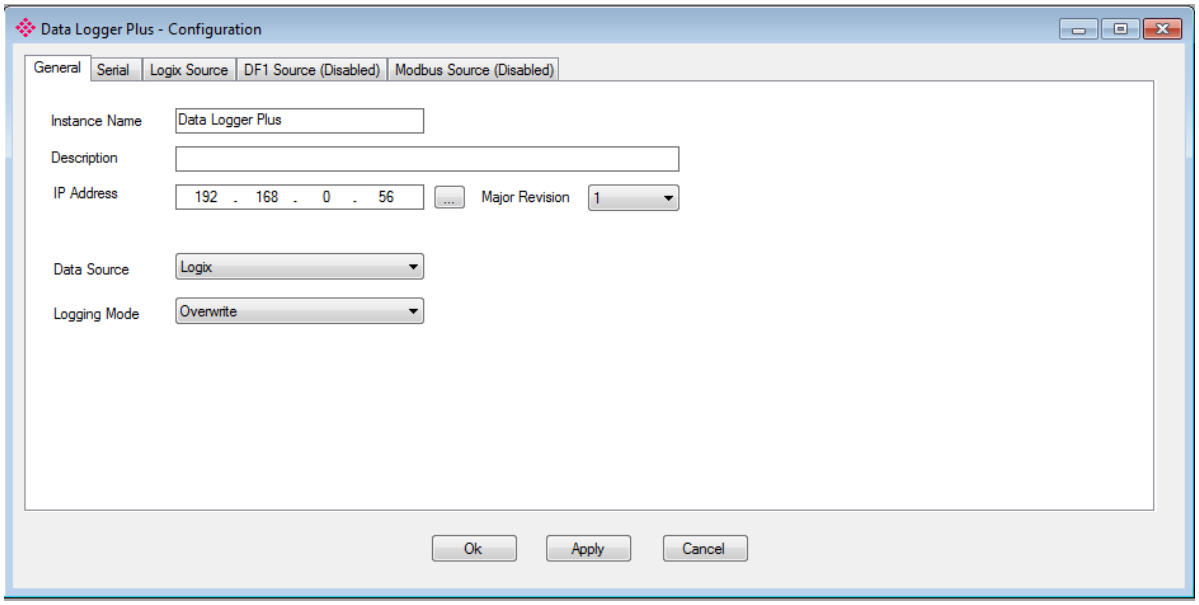


Figure 3.14 - General Configuration

The general configuration consists of the following parameters:

Table 3.1 - General configuration parameters

Parameter	Description
Instance Name	This parameter is a user defined name to identify between various PLX51-DLplus-232's.
Description	This parameter is used to provide a more detailed description of the PLX51-DLplus-232.
Major Revision	The major revision of the PLX51-DLplus-232.
IP Address	The PLX51-DLplus-232's IP address used by the PLX50 Configuration Utility to communicate with the PLX51-DLplus-232.
Data Source	This parameter selects the source of the data. Logix – Rockwell Automation ControlLogix or Compact Logix controller

Parameter	Description
	DF1 – Serial DF1 ModbusRTU – Serial Modbus ModbusTCP – Modbus over Ethernet
Logging Mode	This parameter determines if records are overwritten once the memory is filled. Overwrite = Old records are overwritten, giving priority to newer data. Hold = Old records are preserved while new records are not stored.

3.4.2 Serial Tab

The *Serial* tab is shown in the figure below. The Serial configuration is opened by either double-clicking on the PLX51-DLplus-232 icon in the tree, or right-clicking the PLX51-DLplus-232 icon and selecting *Configuration.* Select the *Serial* tab.

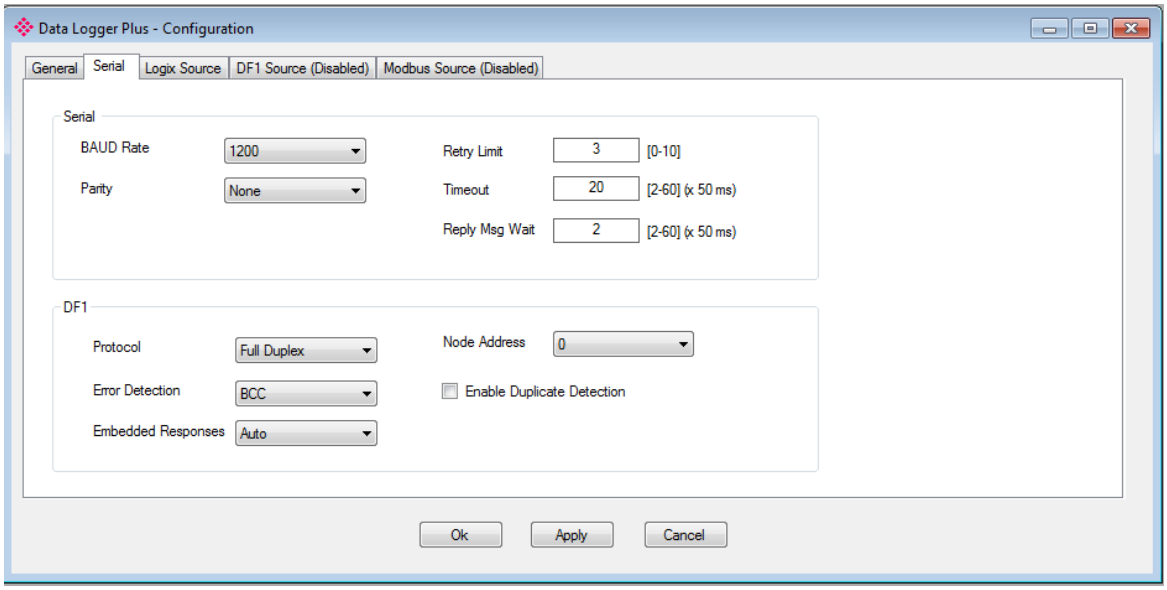


Figure 3.15 - Serial configuration

The Serial configuration (*Serial* tab) consists of general Serial and DF1-specific parameters. For Modbus RTU, only the *Baud Rate* and *Parity* need be configured.

Table 3.2 - Serial Modbus RTU and DF1 configuration parameters

Parameter	Description
Baud Rate	This configures the speed of the data that is sent across the RS232 serial network. The PLX51-DLplus-232 provides the following speeds: 1200, 2400, 4800, 9600, 19200, 38400, 57600, and 115200
Parity	This configures the parity of the RS232 serial port. The PLX51-DLplus-232 allows for Even, Odd, or None.
Protocol	This configures the PLX51-DLplus-232 to operate in full duplex or half duplex mode on the DF1 network.
Error Detection	The PLX51-DLplus-232 can be configured to perform either BCC or CRC checksum validation on incoming and outgoing packets. CRC checksums is a much stronger validation method, but is more processor-intensive to perform.
Embedded Response	This configures the PLX51-DLplus-232 to add the acknowledge responses in the data payload. It can be configured to be Auto Detect or On.

Parameter	Description
	This function is only available in Full Duplex mode.
Node Address	The node address is the local node address of the PLX51-DLplus-232.
Retry Limit	This determines how many times the PLX51-DLplus-232 must retry and message exchange before failing it.
Timeout	This determines the interval between retries for a failed message exchange.
Reply Msg Wait	This is the minimum delay before the DF1 reply is transmitted to the DF1 device.
Duplicate Detection	This configures the PLX51-DLplus-232 to check for duplicate packets and flagging them.

3.5 Data Source Configuration

The Data Source tabs determine the PLX51-DLplus-232 communication mode used to acquire data. The Data Source options include:

- *Logix Source* – ControlLogix and Compact Logix controllers
- *DF1 Source* – For collecting data over DF1 Serial communications
- *Modbus Source* – for Modbus RTU (Serial) and ModbusTCP (Ethernet) communications

Once the data source is selected, each tab allows the configuration of up to three source devices and a total of 200 tags.

Tags can be logged as a result of their individual log criteria **or** via a group trigger. There are eight trigger groups (A thru H), and a tag can be a member of any trigger group. Groups are in turn triggered by one or more tags. The triggering of a group ensures all its member's values are logged at the same instance.

3.5.1 Group and Tag Triggers

Three parameters determine when a tag is triggered:

- Delta Y (Δy) – A change in the value of the tag by this amount or more, AND
- Min ΔT – The minimum time in seconds between each consecutive trigger, OR
- Max ΔT – The maximum time between each consecutive trigger. Setting the Max ΔT to '**0**' disables the "heartbeat" and allows you to log on trigger.

The first two parameters work together to ensure tags are not logged too frequently, and the Max ΔT is set at a minimum logging frequency.

Logix Tag (max. of 200 items.)

Target Name	Target Tag	Group Trigger	Group Member	Data Type	Digital Set	Δy	Min ΔT	Max ΔT
Truck6	OutputRate			SINT		1	10	300
Truck6	TankLevel			INT		10	60	300
Truck6	Speed	A	AB	SINT		10	5	300
Truck6	Direction		AB	REAL		10	30	300
Truck6	Temp	B	B	REAL		3	60	300
Truck6	Mix		B	SINT		1	20	300
Truck6	Pressure	B	B	REAL		1	20	300

Figure 3.16 – Group and Tag Triggers

3.5.2 Logix Source

The *Logix Source* tab is used to configure tags from Rockwell Automation Logix controllers over EtherNet/IP. The PLX51-DLplus-232 can read tags from up to three separate controllers. A *Target Name* must be provided. This acts as a reference to the Logix CIP path. The *Target Name* does not have to match the actual controller name set in RSLogix. The Controller's CIP Path can either be typed in or selected from a list in the *Target Browser*.

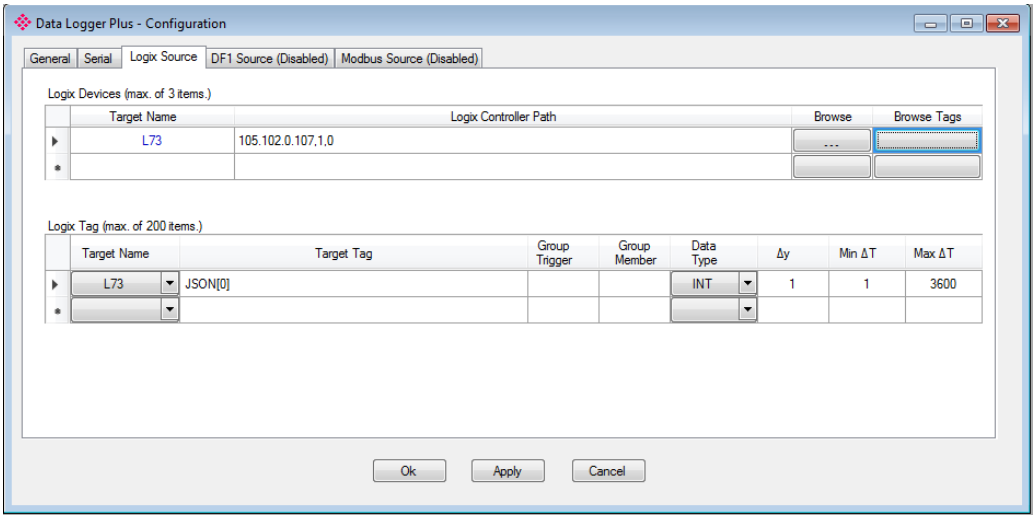


Figure 3.17 – Logix Source Configuration Tab

1 Click the **BROWSE** button in the browse column to launch the *Target Browser*. The *Target Browser* opens and automatically scans for all available EtherNet/IP devices.

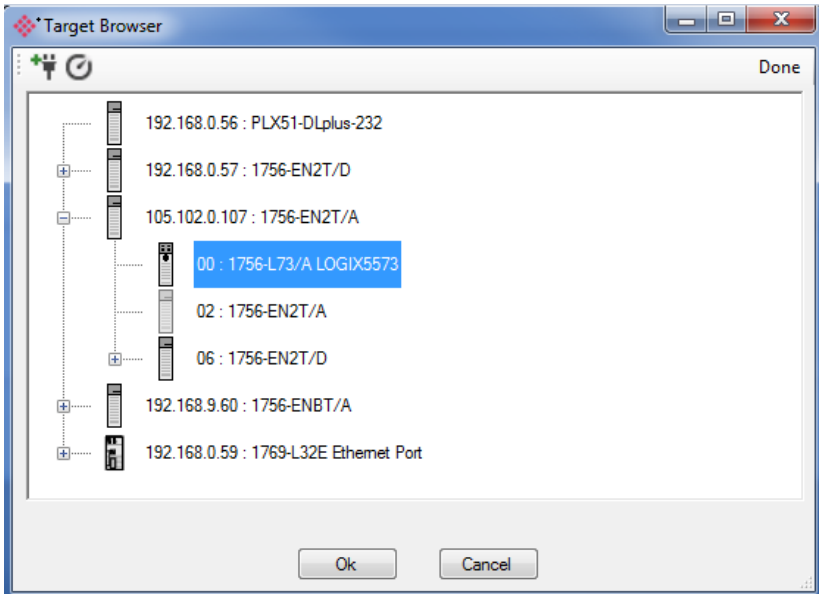


Figure 3.18 – Target Browser Window

2 If the Ethernet/IP module is a bridge module, it can be expanded by right-clicking on the PLX51-DLplus-232 icon and selecting the *Scan* option.

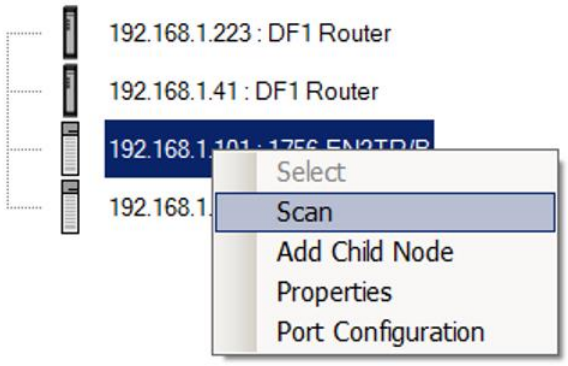


Figure 3.19 - Scanning node in the Target Browser

3 The Logix controller can be selected by clicking the **Ok** button, or by double-clicking on the controller module.

4 Once the controller references have been configured, the individual Logix tags can be added. Tags can either be entered manually or selected by using the *Tag Browser* associated with each controller.

Logix Tag (max. of 200 items.)

	Target Name	Target Tag	Group Trigger	Group Member	Data Type	Digital Set	Δy	Min ΔT	Max ΔT
▶	Truck6	DoorLock			BOOL	NO_YES	1	30	3600
	Truck6	RunTime			DINT		600	600	3600
	Truck6	Direction		A	REAL		5	30	1800
	Truck6	Speed	A	A	SINT		5	30	1800
*									

Figure 3.20 - Logix Tag configuration

Important: Tag names need to match in order for the PLX51-DLplus-232 to correctly identify the tag. Full tag names are needed to be located in program scopes.

5 To launch the *Tag Browser*, click the **BROWSE TAGS** button associated with the controller. Tags that are were already selected and identified are highlighted in green.

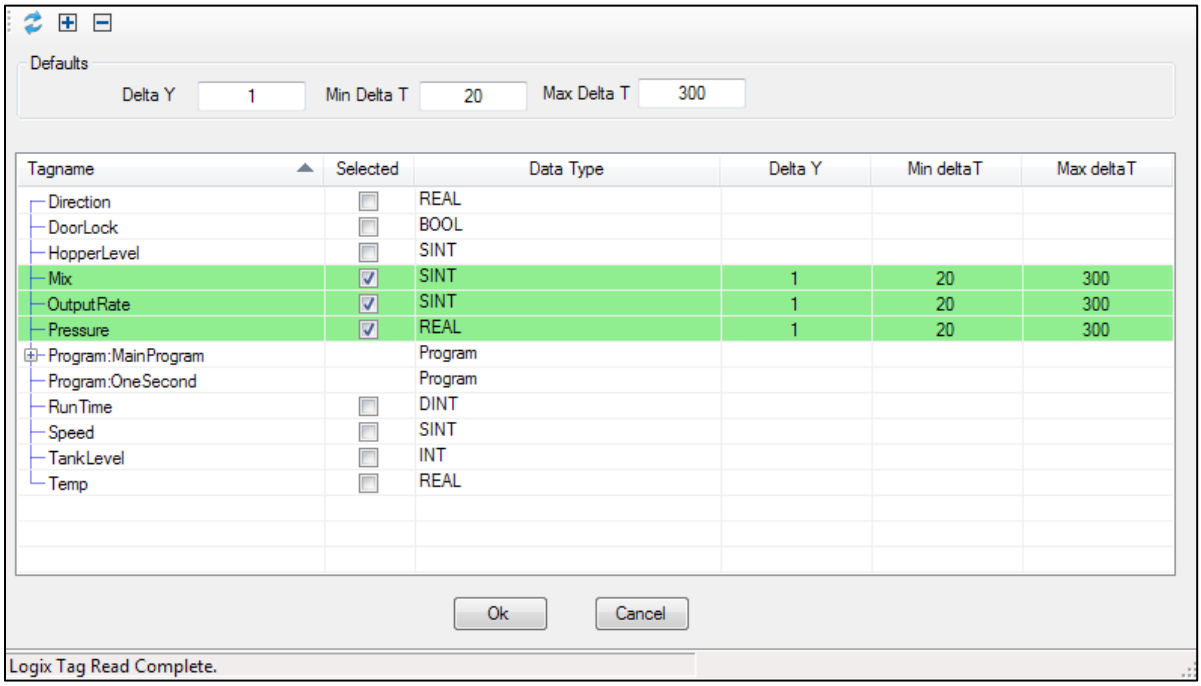


Figure 3.21 – Tag Browser Selection

6 Tags can be removed by selecting the rows in the left margin, and right-clicking to display the **DELETE** option.

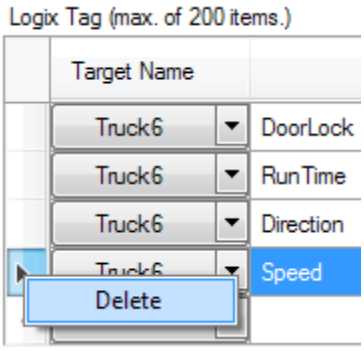


Figure 3.22 – Deleting Tags

3.5.3 DF1 Source

A maximum of three DF1 Sources can be configured. The configuration of each source requires a *Device Name* (used as a reference for tag data sources), the *Device Type* (either PLC5 or SLC), and a *Node Address*.

Each DF1 Tag requires a unique *Tagname* and *Data Address*.

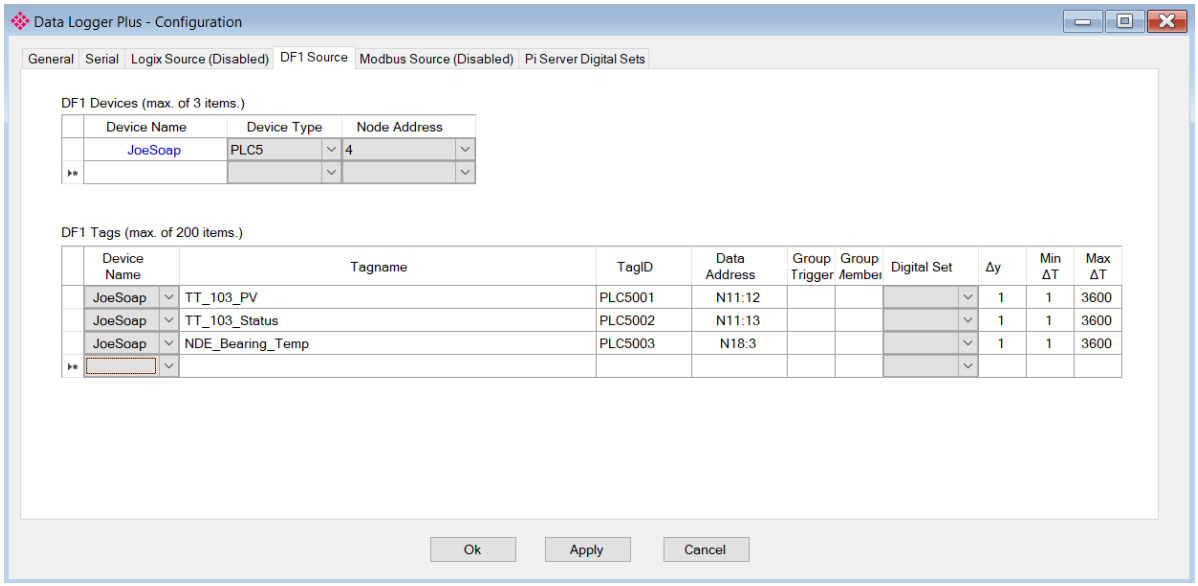


Figure 3.23 – DF1 Source configuration

3.5.4 Modbus Source

Both Modbus RTU and Modbus TCP/IP are configured using the *Modbus Source* tab. For Modbus RTU, a maximum of three Modbus sources can be configured. The configuration of each source requires a *Device Name* (used as a reference for tag data sources), the *IP Address* (Modbus TCP/IP only), and a *Node Address*.

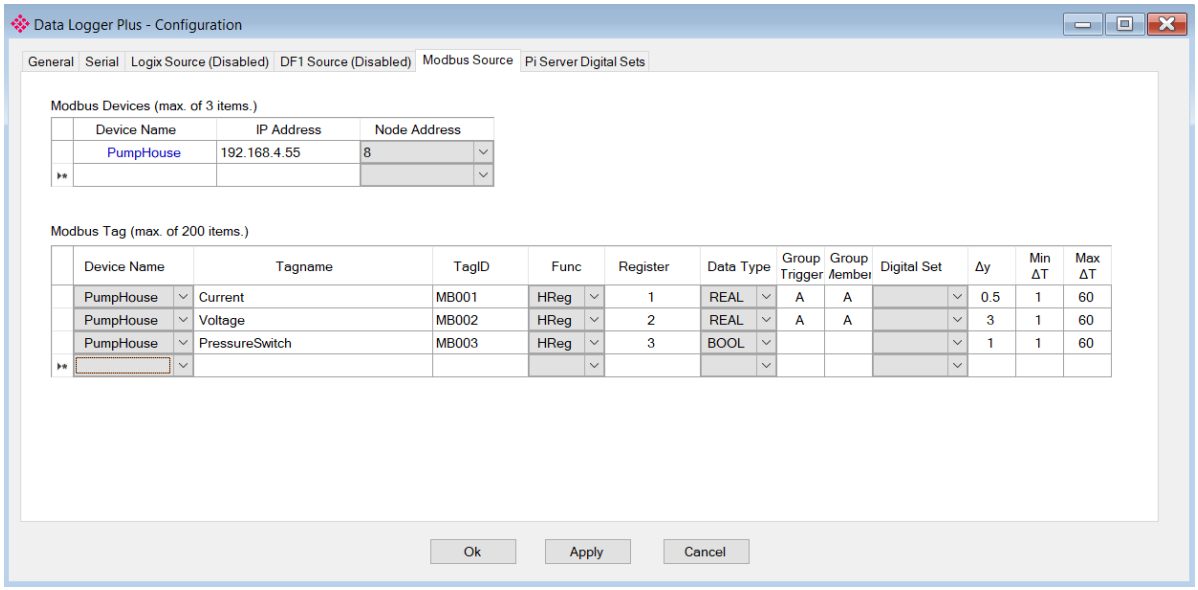


Fig 3.24 – Modbus Source Configuration

3.6 Module Download

Once the configuration is complete, it must be downloaded to the PLX51-DLplus-232.

Before downloading, the connection path of the PLX51-DLplus-232 should be set. This path automatically defaults to the PLX51-DLplus-232 IP address, as set in the PLX51-DLplus-232 configuration. It can be modified if the PLX51-DLplus-232 is not on a local network.

1 The connection path can be set by right-clicking on the PLX51-DLplus-232 icon and selecting the *Connection Path* option.

2 The new connection path can be entered manually or selected by means of the *Target Browser*.

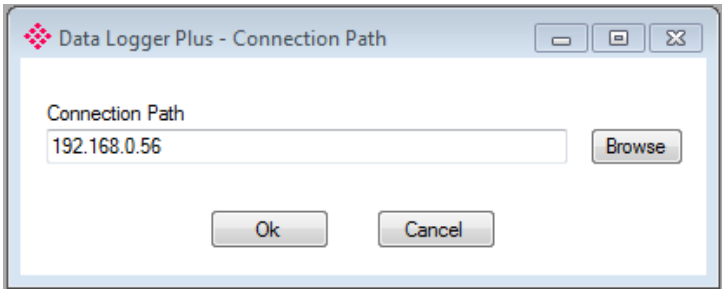


Figure 3.25 - Connection Path

3 To initiate the download, right-click on the PLX51-DLplus-232 icon and select *Download*.

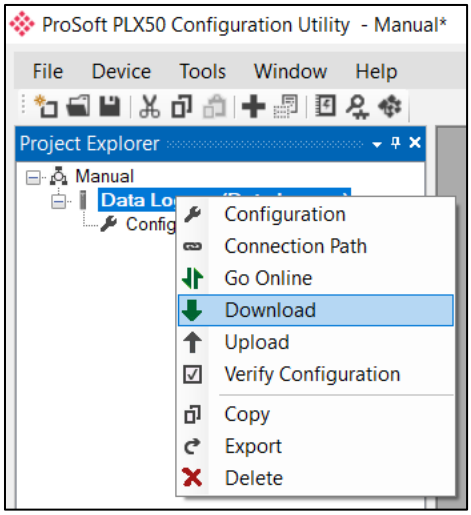


Figure 3.26 - Selecting Download

4 Once complete, you will be notified that the download was successful.

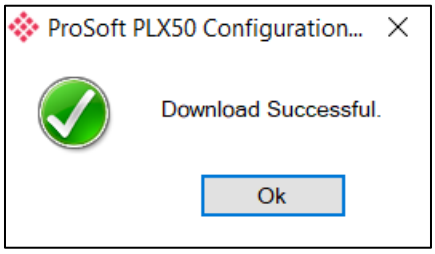


Figure 3.27 - Successful download

5 During the download process, the PLX51-DLplus-232's time will be compared to that of the PC's time. Should the difference be greater than 30 seconds, you will be prompted to set the PLX51-DLplus-232 time to that of the PC time.

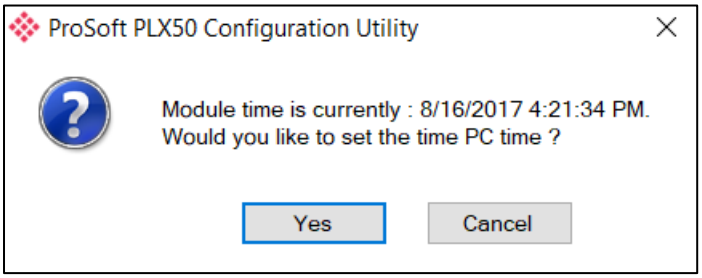


Figure 3.28 – Setting module time

6 The PLX51-DLplus-232 time is used only for the event log. Within the PLX50 Configuration Utility environment, the PLX51-DLplus-232 will be in the *Online* state, indicated by the green circle around the PLX51-DLplus-232 icon.

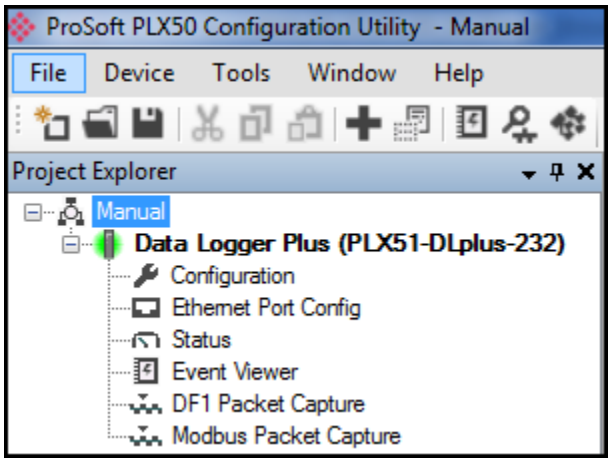


Figure 3.29 - Module Online

7 The PLX51-DLplus-232 is now configured.

4 RSLogix 5000 Configuration

4.1 Add Module to I/O Configuration

The PLX51-DLplus-232 can operate in both a Logix “owned” and standalone mode. When the PLX51-DLplus-232 operates in a Logix “owned” mode, the PLX51-DLplus-232 needs to be added to the RSLogix 5000 / Studio5000 IO tree, as a generic Ethernet module.

1 Right-click on the Ethernet Bridge in RSLogix 5000 and select *New Module*. Then select *ETHERNET-MODULE* and click **Ok**.

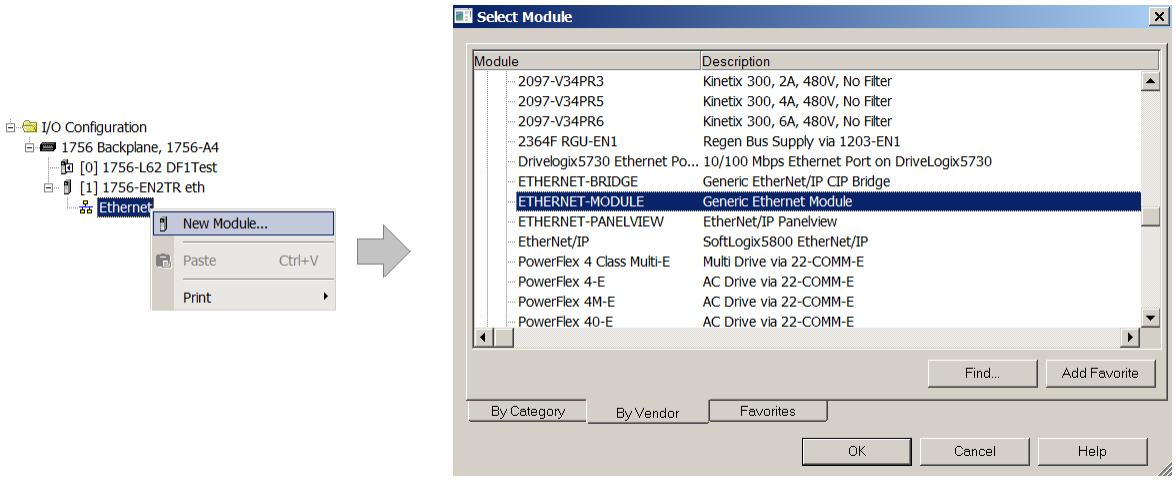


Figure 4.1 - Add a Generic Ethernet Module in RSLogix 5000

2 Enter the IP address of the PLX51-DLplus-232. The *Assembly Instance* and *Size* must also be added for the input, output, and configuration in the *Connection Parameters* section. Below are the required connection parameters.

Table 4.1 - RSLogix class 1 connection parameters

Connection Parameter	Assembly Instance	Size
Input	103	29 (32-bit)
Output	104	1 (32-bit)
Configuration	102	0 (8-bit)

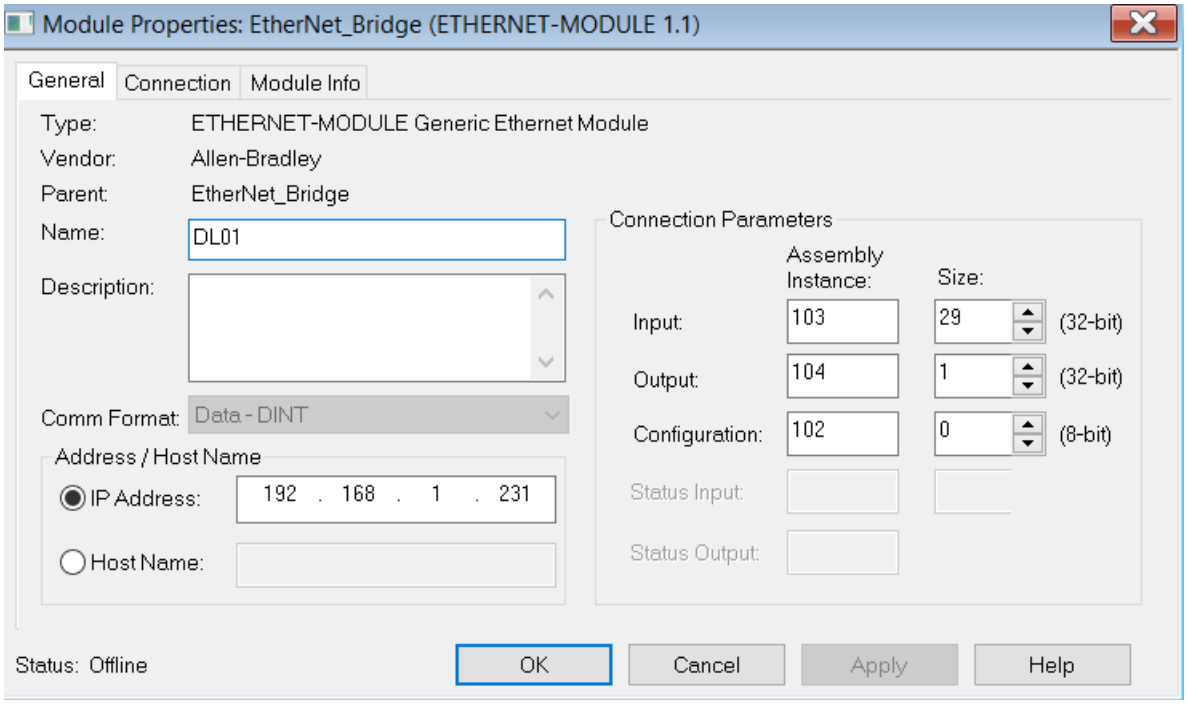


Figure 4.2 - RSLogix General module properties in RSLogix 5000

3 Add the connection requested packet interval (RPI). This is the rate at which the input and output assemblies are exchanged. The recommended value is 500 ms. Refer to the technical specification section in this document for further details on the limits of the RPI.

Important: Although the PLX51-DLplus-232 is capable of running an RPI of 10 ms, it is recommended to set the RPI to 500 ms to avoid unnecessary overloading of the PLX51-DLplus-232 processor.

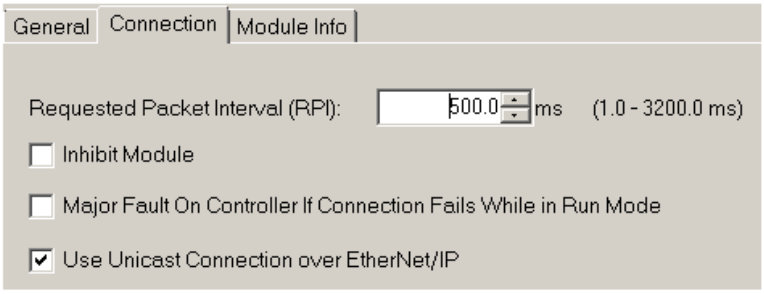


Figure 4.3 - Connection module properties in RSLogix 5000

4 Once the PLX51-DLplus-232 has been added to the RSLogix 5000 IO tree, assign the User Defined Types (UDTs) to the input and output assemblies. You can import the required UDTs by right-clicking on the *User-Defined* sub-folder in the *Data Types* folder and selecting *Import Data Type*. The assemblies are then assigned to the UDTs with a ladder copy instruction (COP).

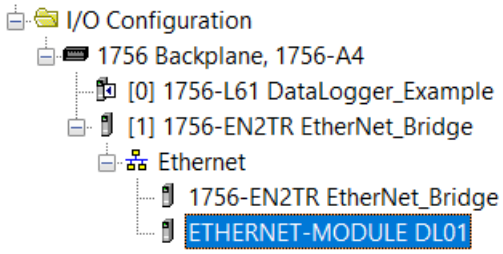


Figure 4.4 – RSLogix 5000 I/O module tree

4.2 Importing UDT's and Mapping Routines

To simplify the mapping of the input image, an RSLogix 5000 Routine Partial Import (.L5X) file is provided.

1 Right-click on the required Program and select the *Import Routine* option.

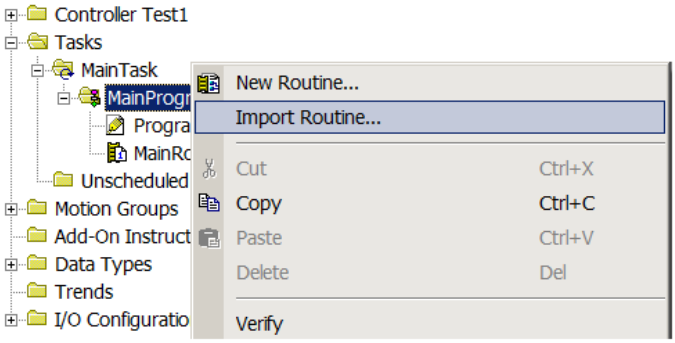


Figure 4.5 – RSLogix 5000 Importing PLX51-DLplus-232 routine and UDTs

2 Select the proper .L5X file.

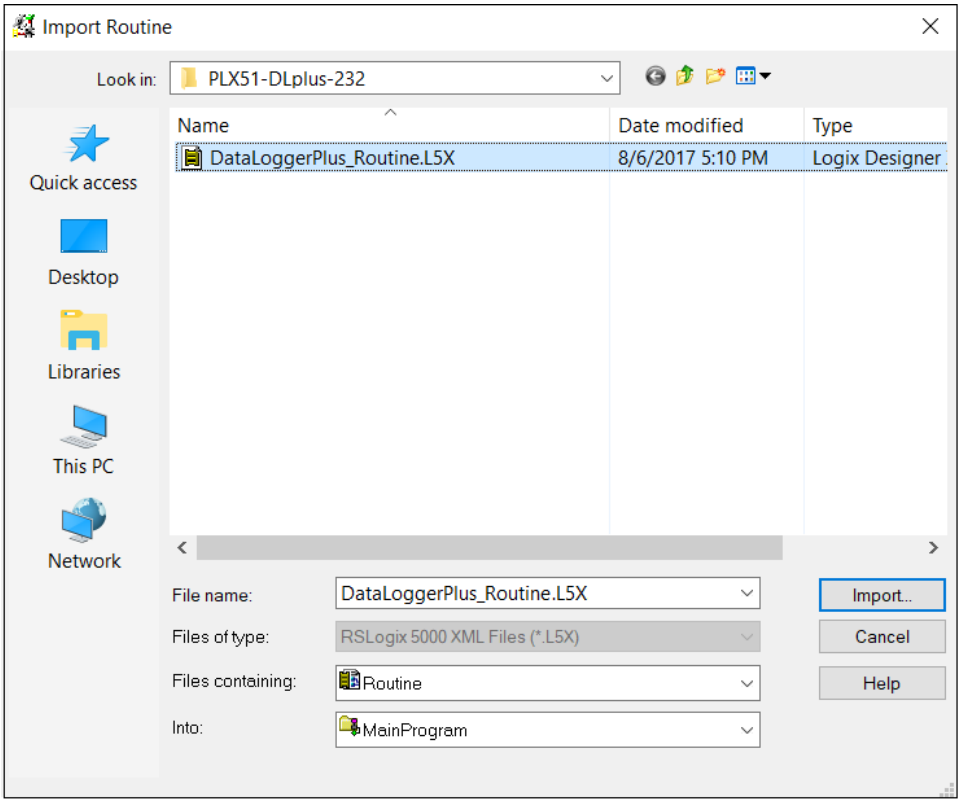


Figure 4.6 - Selecting import file

The import creates the following:

- The required UDTs (user defined data types)
- Controller tags representing the Input Assembly.
- A routine mapping the PLX51-DLplus-232 to the aforementioned tag.

3 You may need to change the routine to map to the correct PLX51-DLplus-232 instance name. Make sure that the mapping routine is called by the Program's Main Routine.

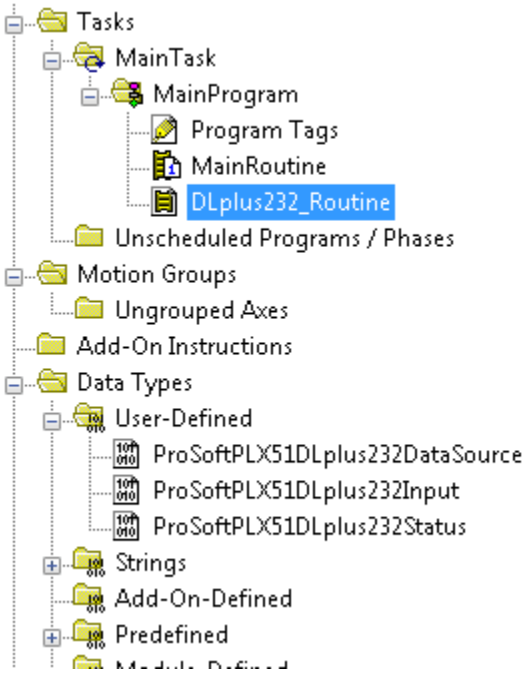


Figure 4.7 - Imported RSLogix 5000 objects

Refer to the Additional Information section of this document for an RSLogix 5000 project example, as well as the required UDTs.

4.3 RSLogix 5000 Assemblies

When the PLX51-DLplus-232 operates in a Logix “owned” mode, the Logix controller establishes a class 1 cyclic communication connection with the PLX51-DLplus-232. An input assembly is exchanged at a fixed interval. The provided UDTs convert the input arrays into tag-based assemblies. Refer to the Additional Information section in this document for more information on the input UDTs. There are no Output or Configuration assemblies.

⊟ PLX51DLplus232Input	{...}
⊞ PLX51DLplus232Input.Instance	'Data Logger Plus'
⊟ PLX51DLplus232Input.Status	{...}
PLX51DLplus232Input.Status.Running	1
PLX51DLplus232Input.Status.ConfigurationValid	0
PLX51DLplus232Input.Status.ContinuousLogging	0
PLX51DLplus232Input.Status.LogRollover	0
PLX51DLplus232Input.Status.LoggingInhibited	0
PLX51DLplus232Input.Status.LoggingStopped	0
PLX51DLplus232Input.CachePercentageUsed	0.0
⊞ PLX51DLplus232Input.CacheRecordCount	0
⊞ PLX51DLplus232Input.TotalRecordCount	0
⊞ PLX51DLplus232Input.ActiveTagCount	0
⊟ PLX51DLplus232Input.DataSource	{...}
PLX51DLplus232Input.DataSource.EtherNetIP	1
PLX51DLplus232Input.DataSource.DF1	1
PLX51DLplus232Input.DataSource.ModbusRTU	1
PLX51DLplus232Input.DataSource.ModbusTCP	1
⊞ PLX51DLplus232Input.DataSourceReadCount	0

Figure 4.8 - Input assembly UDT structure

4.3.1 Input Assembly

The following parameters are used in the input assembly of the PLX51-DLplus-232.

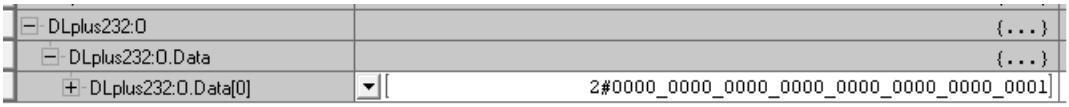
Table 4.2 - RSLogix 5000 input assembly parameters

Parameter	Datatype	Description
Instance	STRING	Instance name of the PLX51-DLplus-232 that was configured under the *General Configuration* tab in the PLX50 Configuration Utility.
Status.Running	BOOL	Set if the PLX51-DLplus-232 has a valid configuration and is reading tags.
Status.ConfigValid	BOOL	Set if a valid configuration is executing in the PLX51-DLplus-232.
Status.ContinuousLogging	BOOL	Set if *Logging Mode* is set to Overwrite, clear for Hold.
Status.ConfigurationValid	BOOL	Set if a valid configuration is executing in the PLX51-DLplus-232.
Status.LoggingInhibited	BOOL	Not Used.
Status.LoggingStopped	BOOL	Not Used.
CachePercentage	REAL	The number of cached records not yet uploaded, as a percentage of the total record capacity of 16,777,216.
CacheRecordCount	DINT	The number of cached records not yet uploaded.
TotalRecordCount	DINT	The total number of cached records uploaded or not.
ActiveTagCount	DINT	The number of individual tags configured to be read.
DataSource .EtherNetIP	BOOL	Set if the data source is set to Logix.
DataSource .DF1	BOOL	Set if the data source is set to DF1.

Parameter	Datatype	Description
DataSource .ModbusRTU	BOOL	Set if the data source is set to Modbus RTU.
DataSource .ModbusTCP	BOOL	Set if the data source is set to Modbus TCP/IP.
DataSourceReadCount	DINT	The number of tag reads from the configured data source.

4.4 Inhibiting the Module

The PLX51-DLplus-232 can be inhibited by setting the first bit of the DLplus232:O.Data[0] output image to '1', as shown below:



5 Diagnostics

5.1 LEDs

The PLX51-DLplus-232 provides three LEDs for diagnostics purposes as shown below.

Figure 5.1 - Front view

Table 5.1 - Module LED operation

LED	Description
Ok	The **Ok** LED provides information of the system-level operation of the PLX51-DLplus-232. If the LED is green, then the PLX51-DLplus-232 has booted and is running correctly. If the LED is red, then the PLX51-DLplus-232 is not operating correctly. For example, if the PLX51-DLplus-232 application firmware has been corrupted or there is a hardware fault, **Ok** LED will be red.
Act	The **Act** LED is used for the RS232 serial port. For every successful received DF1 or Modbus-RTU packet, the **Act** LED toggles green. The LED toggles red if a corrupted packet is received (eg. failed checksum).
Eth	The **Eth** LED illuminates when an Ethernet link is detected (by plugging in a connected Ethernet cable). The LED flashes when traffic is detected.

5.2 Module Status Monitoring

The PLX51-DLplus-232 provides a range of statistics that can assist with module operation, maintenance, and troubleshooting. The statistics can be accessed by the PLX50 Configuration Utility or using the PLX51-DLplus-232 web server.

1 To view the PLX51-DLplus-232's status in the PLX50 Configuration Utility, the PLX51-DLplus-232 must be online. If the PLX51-DLplus-232 is not Online (following a recent configuration download), right-click on the PLX51-DLplus-232 icon and select the *Go Online* option.

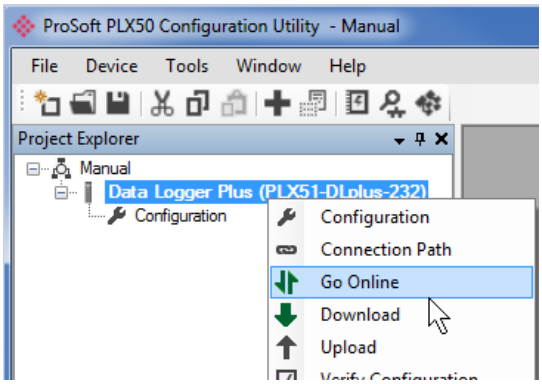


Figure 5.2 - Selecting to Go Online

2 The *Online* mode is indicated by the green circle behind the PLX51-DLplus-232 in the Project Explorer tree.

3 The *Status* window is opened by either double-clicking on the *Status* option in the Project Explorer tree, or by right-clicking on the PLX51-DLplus-232 icon and selecting *Status*.

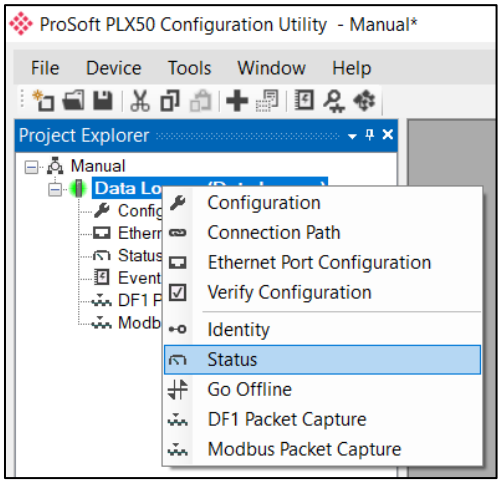


Figure 5.3 - Selecting online Status

4 The *Status* window contains multiple tabs to display the current status of the PLX51-DLplus-232.

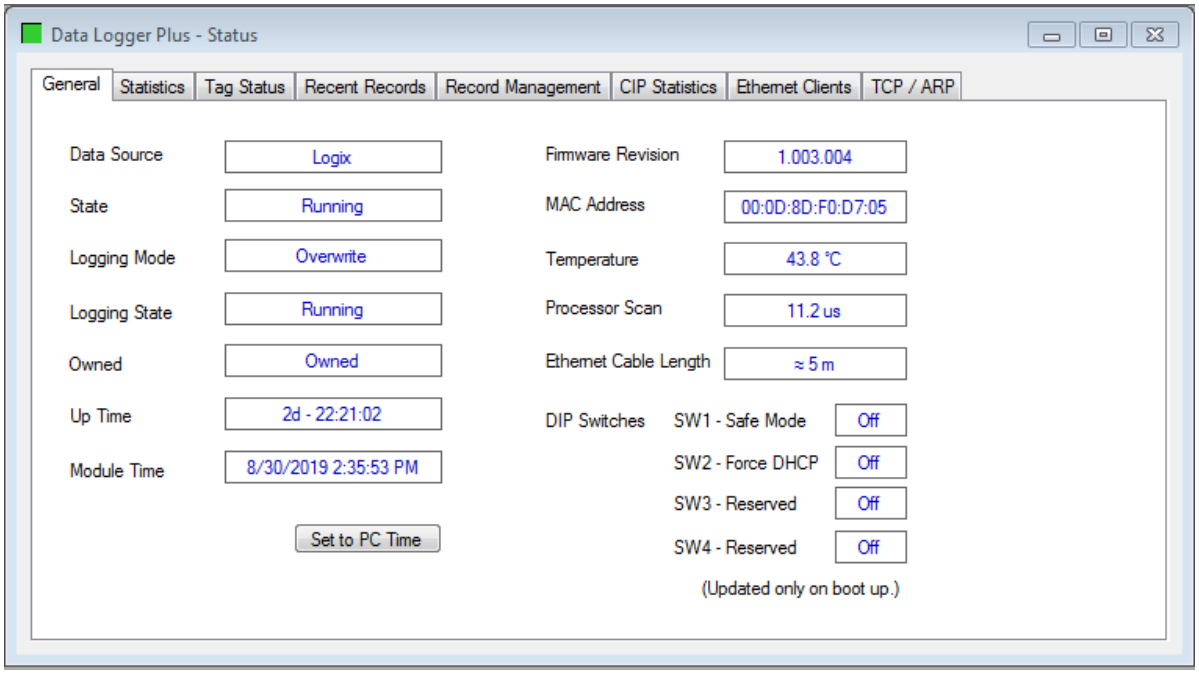


Figure 5.4 - Status monitoring - General

5.2.1 General Tab

The *General* tab displays the following general parameters as well as setting the PLX51-DLplus-232 time to the PC time:

Table 5.2 - Parameters displayed in the Status Monitoring – General Tab

Parameter	Description
Data Source	Logix, DF1, Modbus RTU, Modbus TCP/IP
State	This is the current state of the module. **Running** The module is reading tags and logging to the log. **Stopped** The module is idle and not reading tags or logging data.
Logging Mode	Hold or Overwrite – determines if records are overwritten when the cache is full.
Logging State	This is the current state of the logging in the module. **Running** Data is being read from the source and logged if the criteria is met. **Running Rollover** Data is being read from the source and logged if the criteria is met. In this state the event index has rolled over at least once. **Inhibited** The module has stopped reading and logging data, because the user has inhibited it from Logix. **Stopped** The module has stopped logging data, because it has reached maximum events and the module is set to not overwrite.
Owned	Indicates whether or not the PLX51-DLplus-232 is currently owned (Class 1) by a Logix controller.
Up Time	Indicates the elapsed time since the PLX51-DLplus-232 was powered up.
Module Time	Indicates the PLX51-DLplus-232's internal time. The PLX51-DLplus-232 time is stored in UTC (Universal Coordinate Time) but displayed on this page according to the local PC Time Zone settings.
MAC Address	Displays the PLX51-DLplus-232's unique Ethernet MAC address.
Temperature	Internal temperature of the PLX51-DLplus-232.
Processor Scan	Amount of time (microseconds) taken by the PLX51-DLplus-232's processor in the last scan.
DIP Switch Position	Status of the DIP switches when the PLX51-DLplus-232 booted. Note that this status will not change if the DIP switches are altered when the PLX51-DLplus-232 is running.

5.2.2 Statistics tab

The *Statistics* tab displays the statistics of the record cache and data source.

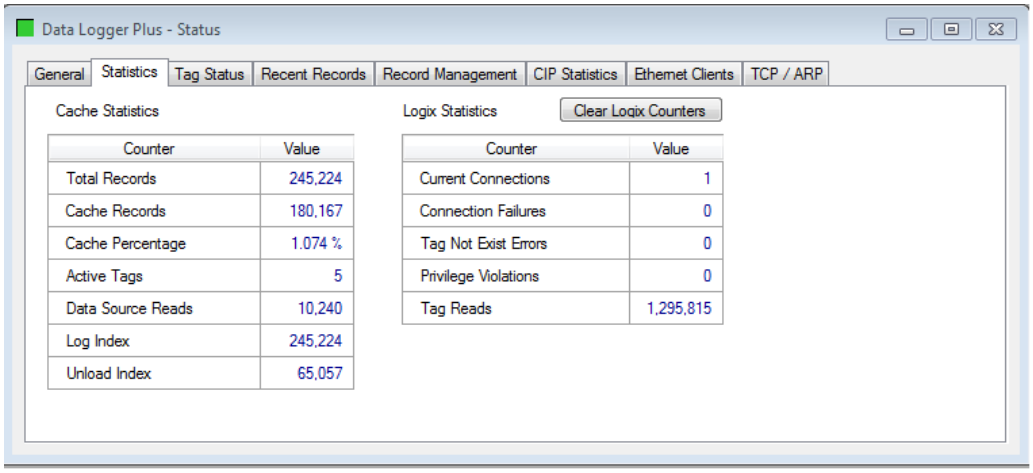


Figure 5.5 – Transactions

Table 5.3 – Cache Statistics

Statistic	Description
Total Records	The total number of cached records, uploaded or not.
Cache Records	The number of cached records not yet uploaded.
Cache Percentage	The number of cached records not yet uploaded, as a percentage of the total record capacity of 16,777,216.
Active Tags	The number of individual tags configured to be read.
Data Source Reads	The number of tag reads from the configured data source.
Log Index	The current record index being written to.
Unload Index	The upload record index. Managed by the Unload Service.

Table 5.4 – Logix statistics

Statistic	Description
Current Connections	The number of current open class 3 connections.
Connection Failures	The number of failed attempts at establishing a class 3 connection with a Logix controller.
Tag Not Exist Errors	The number of failed tag read/write transactions due to a non-existent destination tag.
Privilege Violations	The number of failed tag read/write transactions due to a privilege violation error. This may be caused by the *External Access* property of the Logix tag being set to either **None** or **Read Only**.
Tag Reads	The number of tag read transactions executed by the PLX51-DLplus-232.

5.2.3 Tag Status Tab

The *Tag Status* tab provides current values for all tags, along with their trigger settings. The following tag mapping statistics are only relevant when the PLX51-DLplus-232 is running in either *Reactive Tag* or *Scheduled Tag* mode.

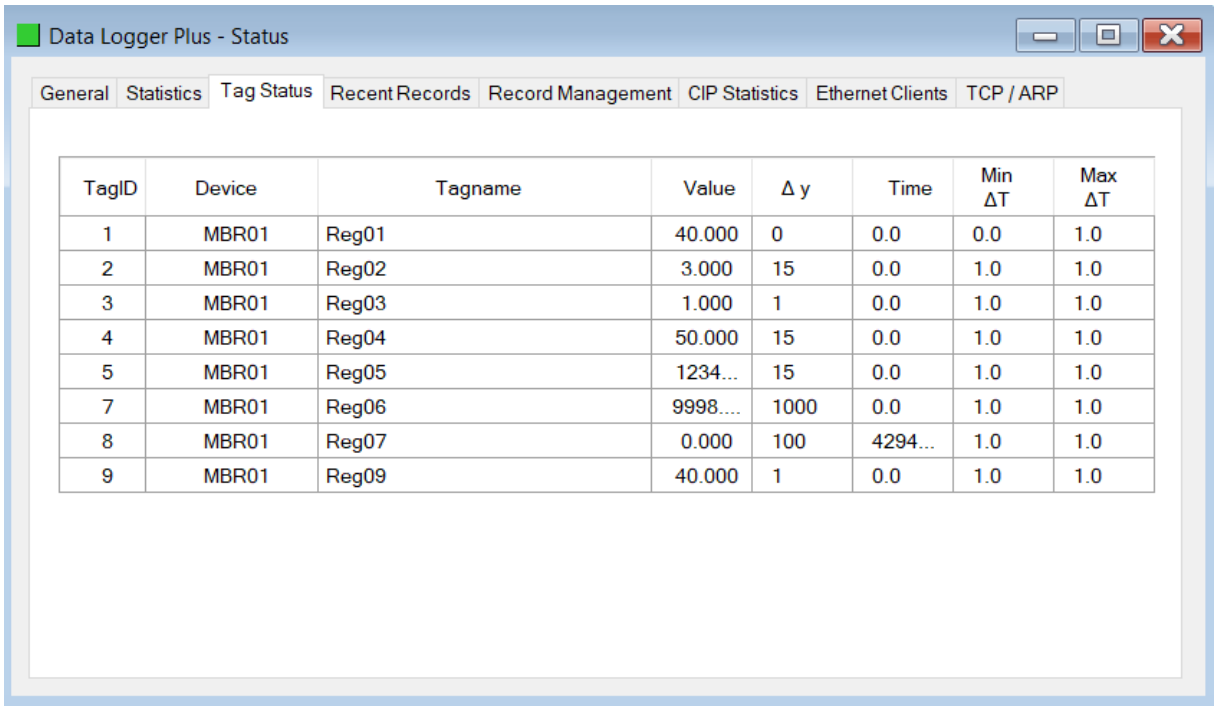


Figure 5.6 – Tag Status

Table 5.5 – Tag Status

Statistic	Description
TagID	Configured Tag ID for the specific Tag/Register/File.
Device	The configured source device where the tag is extracted.
Tagname	The Tagname configured for the specific tag.
Value	The last read value from the specific tag.
Delta Y	Configured deadband for the tag value.
Time	The time in seconds since the last data was logged.
Min Delta Time	Minimum time in seconds between consecutive data logging.
Max Delta Time	Maximum time in seconds between consecutive data logging.

5.2.4 Recent Records Tab

The *Recent Records* tab provides a list of the last records recorded with their time stamp and value.

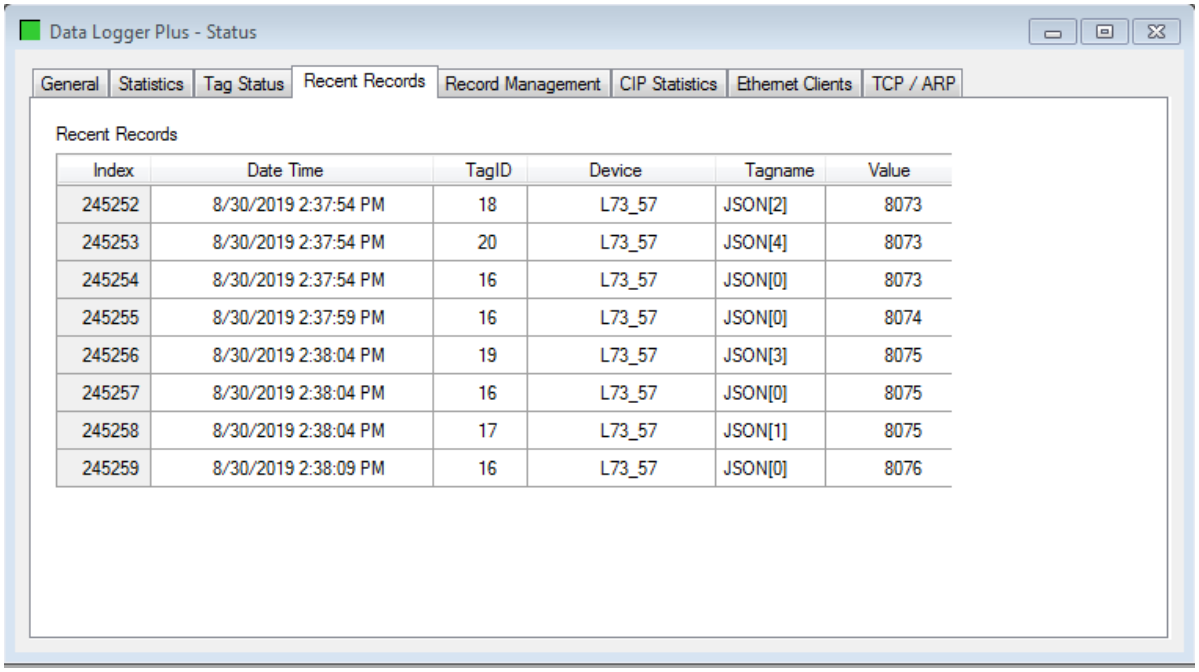


Figure 5.7 – Recent Records

Table 5.6 – Recent Records

Statistic	Description
Index	Logged data index.
Date Time	The time stamp when the data was logged.
Tag ID	Configured Tag ID for the specific Tag/Register/File.
Device	The configured source device where the tag is extracted.
Tagname	The Tagname configured for the specific tag.
Value	The last logged value from the specific tag.

5.2.5 Record Management Tab

The *Record Management* tab manages the PLX51-DLplus-232 records. Records can be downloaded to a (.csv) file format. Options to reset the log indices and erasing the cache are also available.

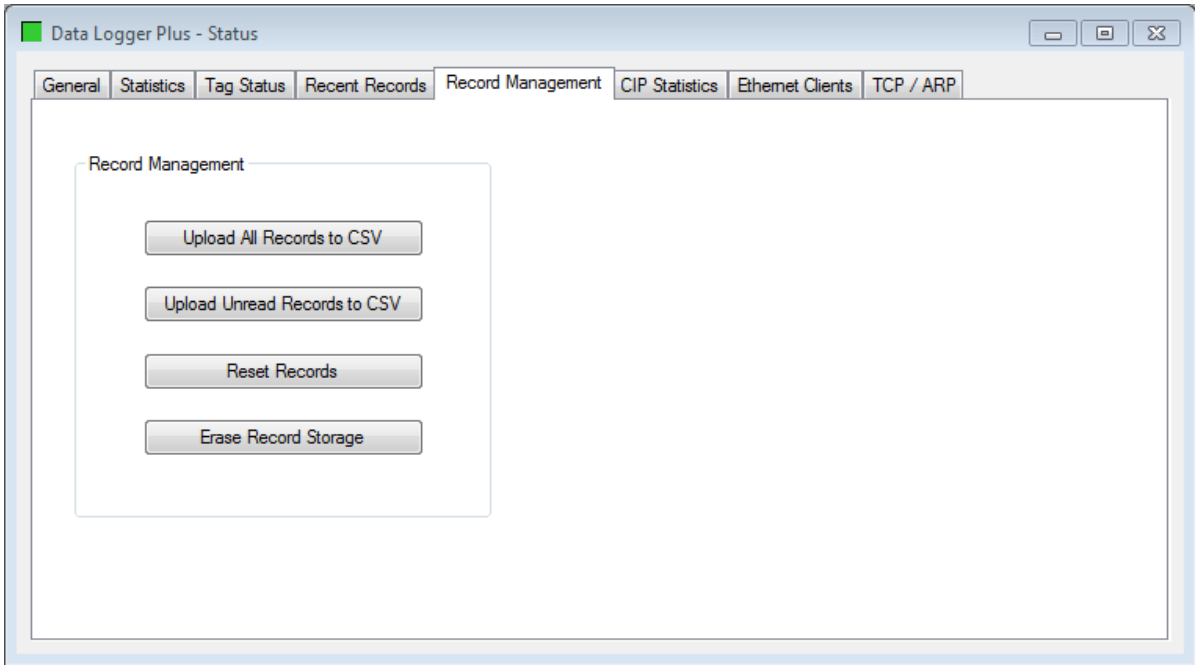


Figure 5.8 – Record Management

Table 5.7 – Record Management

Parameter	Description
Upload All Records to CSV	Upload all records currently in the module.
Upload Unread Records to CSV	Upload unread records currently in the module.
Reset Records	Resets the log indices
Erase Record Storage	Erases the cache of all records

5.2.6 CIP Statistics Tab

The *CIP Statistics* tab provides a set of Common Industrial Protocol (CIP) communication statistics.

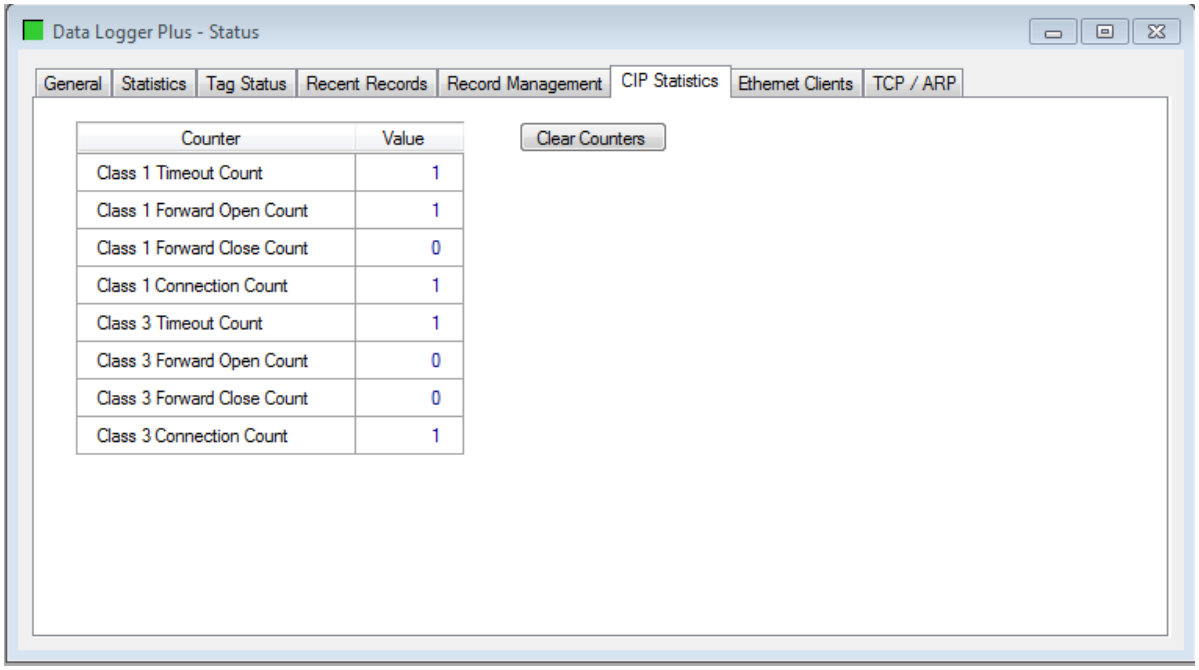


Figure 5.9 – CIP Statistics

Table 5.8 – CIP Statistics

Statistic	Description
Class 1 Timeout Count	Number of times a Class 1 connection has timed out
Class 1 Forward Open Count	Number of Class 1 Connection establish attempts
Class 1 Forward Close Count	Number of Class 1 Connection close attempts
Class 1 Connection Count	Number of Class 1 Connections currently active
Class 3 Timeout Count	Number of times a Class 3 connection has timed out
Class 3 Forward Open Count	Number of Class 3 Connection establish attempts
Class 3 Forward Close Count	Number of Class 3 Connection close attempts
Class 3 Connection Count	Number of Class 3 Connections currently active

5.2.7 Ethernet Clients Tab

The *Ethernet Clients* tab provides a count of EtherNet Client and EtherNet/IP connections.

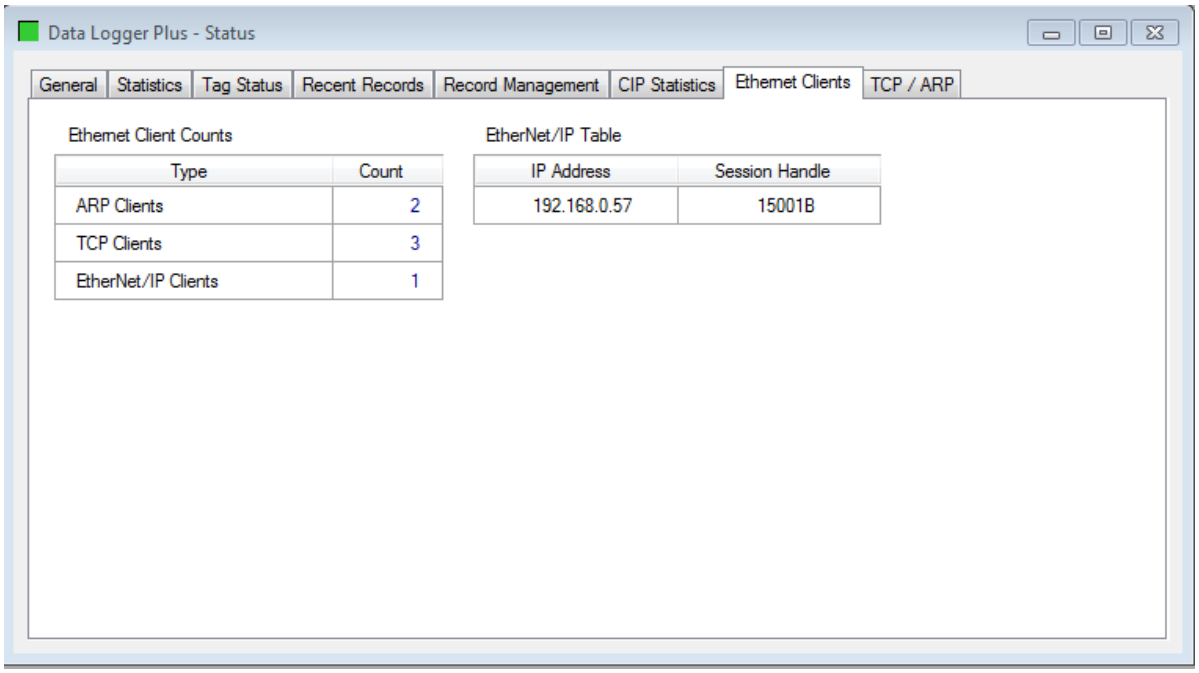


Figure 5.10 – Ethernet Connection Counts

Table 5.9 – Ethernet Client Counts

Statistic	Description
ARP Clients	Number of active clients in the ARP table
TCP Clients	Number of active connections in the TCP client table
EtherNet/IP Clients	Number of active connections in the ENIP client table

Table 5.10 – EtherNet/IP Table

Statistic	Description
IP Address	IP address of the client in the ENIP client table
Session Handle	Session handle in the ENIP client table

5.2.8 TCP / ARP Tab

The *TCP / ARP* tab lists the ARP and TCP/IP information associated with the known MAC addresses in the network. Parameters include the *IP Address*, *Remote Port* number, and *Local Port* number.

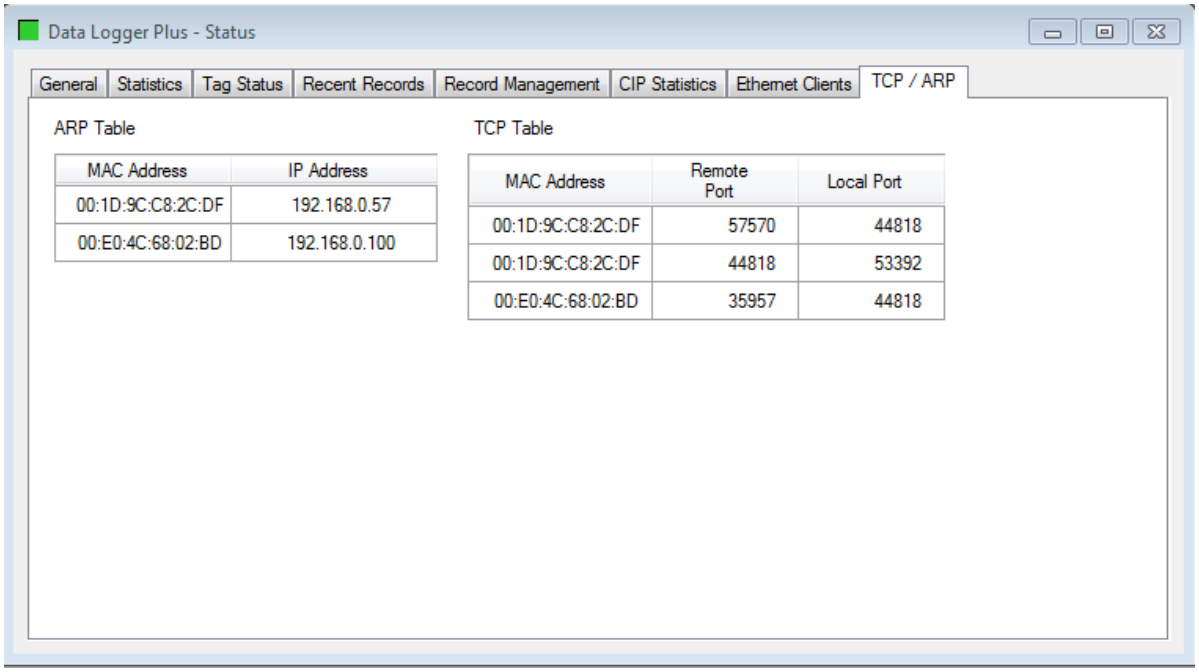


Figure 5.11 – TCP and ARP Table Entries

Table 5.11 – ARP Table

Statistic	Description
MAC Address	MAC address of the client in the ARP Table
IP Address	IP address of the client in the ARP Table

Table 5.12 – TCP Table

Statistic	Description
MAC Address	MAC address of the client in the TCP Table
Remote Port	Remote TCP port of the client in the TCP Table
Local Port	Local TCP port of the client in the TCP Table

6 Retrieving Logged Data from the PLX51-DLplus-232

6.1 DF1 Packet Capture

The PLX51-DLplus-232 provides the capability to capture the DF1 traffic for analysis.

1 To begin the packet capture of the PLX51-DLplus-232, double-click on the *DF1 Packet Capture* selection in the Project Explorer tree.

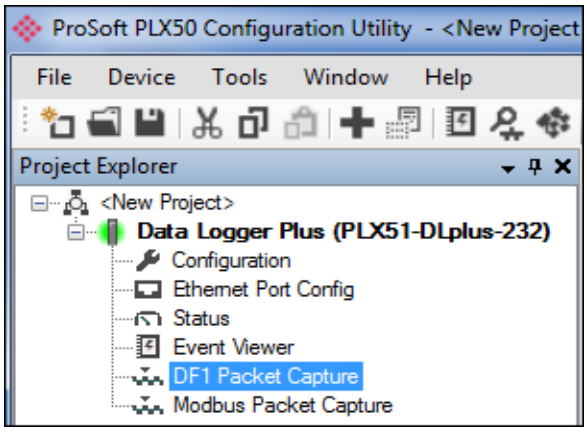


Figure 6.1 - Selecting DF1 Packet Capture

2 The *DF1 Packet Capture* window opens and automatically starts capturing all DF1 packets.

Note: The PLX51-DLplus-232 keeps a circular buffer of the last twenty DF1 packets. Thus, there may be up to 20 packets in the capture that were received / sent before the capture was initiated.

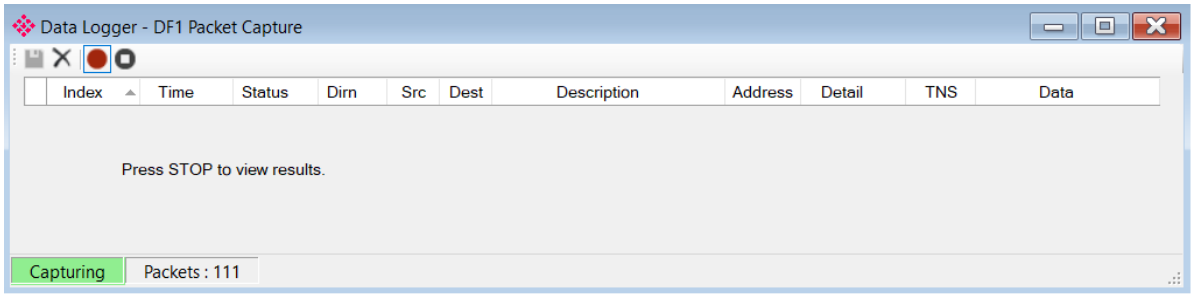
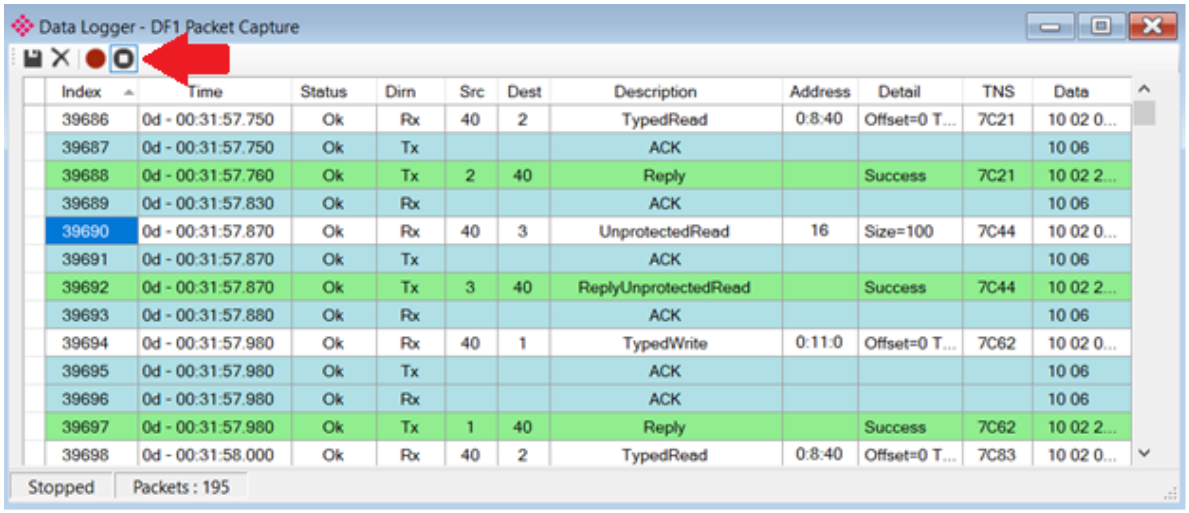


Figure 6.2 - DF1 packet capture

3 To display the captured DF1 packets, the capture process must first be stopped by pressing the **STOP** button.



Index	Time	Status	Dirn	Src	Dest	Description	Address	Detail	TNS	Data
39686	0d - 00:31:57.750	Ok	Rx	40	2	TypedRead	0:8:40	Offset=0 T...	7C21	10 02 0...
39687	0d - 00:31:57.750	Ok	Tx			ACK				10 06
39688	0d - 00:31:57.760	Ok	Tx	2	40	Reply		Success	7C21	10 02 2...
39689	0d - 00:31:57.830	Ok	Rx			ACK				10 06
39690	0d - 00:31:57.870	Ok	Rx	40	3	UnprotectedRead	16	Size=100	7C44	10 02 0...
39691	0d - 00:31:57.870	Ok	Tx			ACK				10 06
39692	0d - 00:31:57.870	Ok	Tx	3	40	ReplyUnprotectedRead		Success	7C44	10 02 2...
39693	0d - 00:31:57.880	Ok	Rx			ACK				10 06
39694	0d - 00:31:57.980	Ok	Rx	40	1	TypedWrite	0:11:0	Offset=0 T...	7C62	10 02 0...
39695	0d - 00:31:57.980	Ok	Tx			ACK				10 06
39696	0d - 00:31:57.980	Ok	Rx			ACK				10 06
39697	0d - 00:31:57.980	Ok	Tx	1	40	Reply		Success	7C62	10 02 2...
39698	0d - 00:31:58.000	Ok	Rx	40	2	TypedRead	0:8:40	Offset=0 T...	7C83	10 02 0...

Figure 6.3 - DF1 Packet Capture complete

The captured DF1 packets are listed as follows:

Table 6.1 - DF1 Packet Capture fields

Statistic	Description
Index	The packet index, incremented for each packet sent or received.
Time	The elapsed time since the PLX51-DLplus-232 was powered up.
Status	The status of the packet. Received packets are checked for valid DF1 constructs and valid checksums.
Dirn	The direction of the packet, either transmitted (Tx) or received (Rx).
Src	DF1 node address of the message source.
Dest	DF1 node address of the message destination.
Description	Brief description of the packet, usually the command.
Address	The string representing a PLC data address, where applicable.
Detail	Additional details associated with command.
TNS	Transaction number. Used to match request and reply messages.
Data	The packet's raw data displayed in space delimited hex.

4 The packet capture can be saved to a file for further analysis by selecting the **SAVE** button on the toolbar.

5 Previously saved DF1 Packet Capture files can be viewed by selecting the *DF1 Packet Capture Viewer* option in the *Tools* menu.

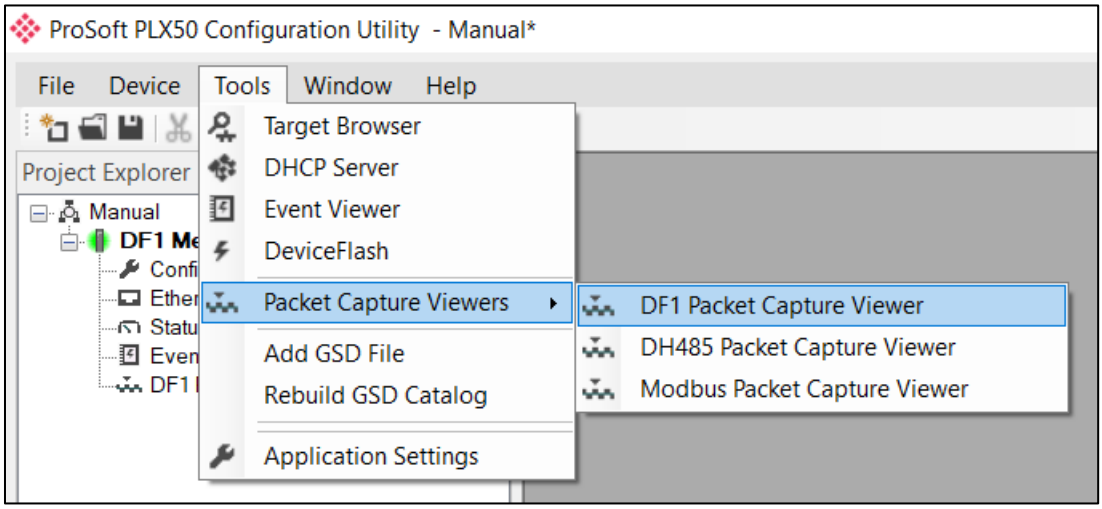


Figure 6.4 - Selecting the DF1 Packet Capture Viewer

6.2 Modbus Packet Capture

The PLX51-DLplus-232 provides the capability to capture the Modbus traffic for analysis.

1 To begin the capture of the PLX51-DLplus-232, double-click on the *Modbus Packet Capture* selection in the Project Explorer tree.

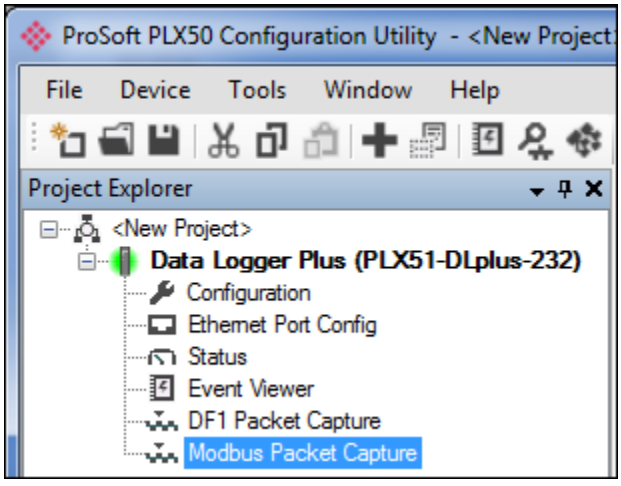


Figure 6.5 - Selecting Modbus Packet Capture

2 The *Modbus Packet Capture* window opens and automatically starts capturing all Modbus packets.

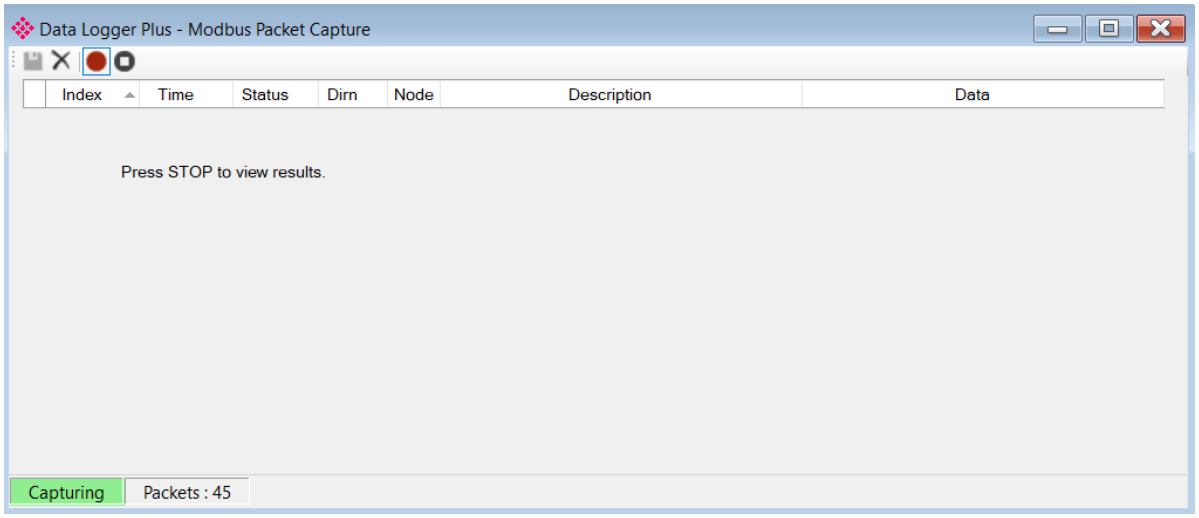


Figure 6.6 – Modbus packet capture

3 To display the captured Modbus packets, the capture process must first be stopped by pressing the **STOP** button.

Figure 6.7 – Modbus Packet Capture complete

The captured Modbus packets are tabulated as follows:

Table 6.2 – Modbus Packet Capture fields

Statistic	Description
Index	The packet index, incremented for each packet sent or received.
Time	The elapsed time since the PLX51-DLplus-232 was powered up.
Status	The status of the packet. Received packets are checked for valid Modbus constructs and valid checksums.
Dirn	The direction of the packet, either transmitted (Tx) or received (Rx).
Node	Modbus node address of the message destination.
Description	A brief description of the packet, showing the function and register range if applicable.
Data	The raw packet data.

4 The packet capture can be saved to a file for further analysis by selecting the **SAVE** button on the toolbar.

5 Previously saved Modbus Packet Capture files can be viewed by selecting the *Modbus Packet Capture Viewer* option in the *Tools* menu.

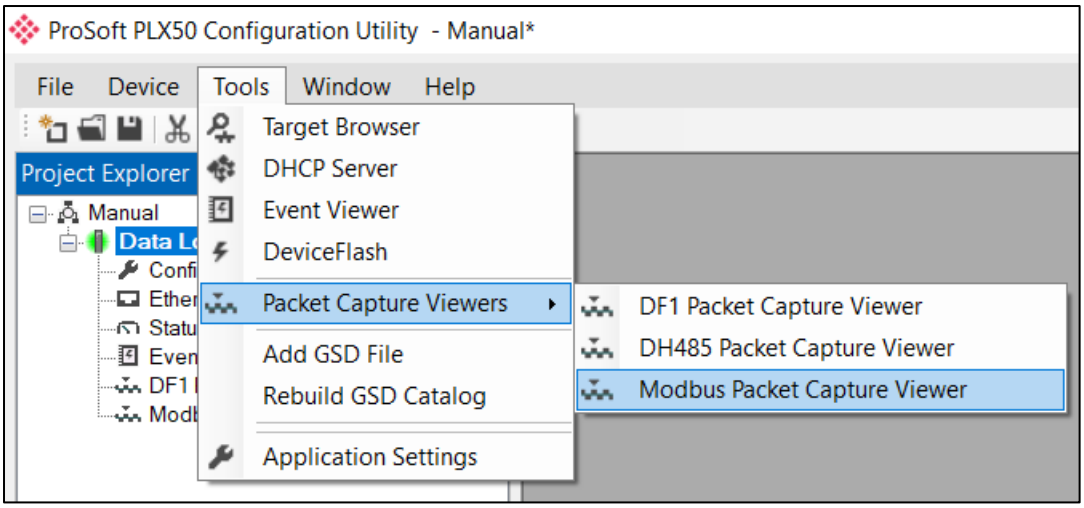


Figure 6.8 - Selecting the Modbus Packet Capture Viewer

6.3 Module Event Log

The PLX51-DLplus-232 logs various diagnostic records to an internal event log. These logs are stored in non-volatile memory and can be displayed in the PLX50 Configuration Utility or the web server.

1 To view them in the PLX50 Configuration Utility, select the *Event Viewer* option in the Project Explorer tree.

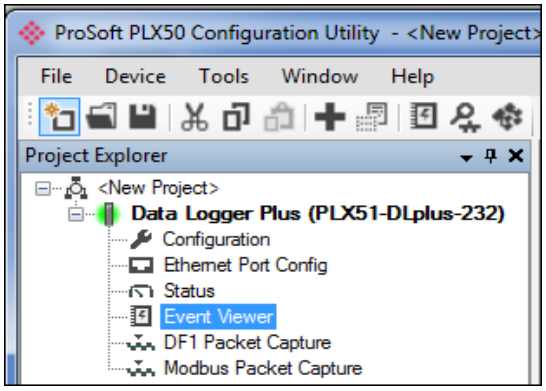
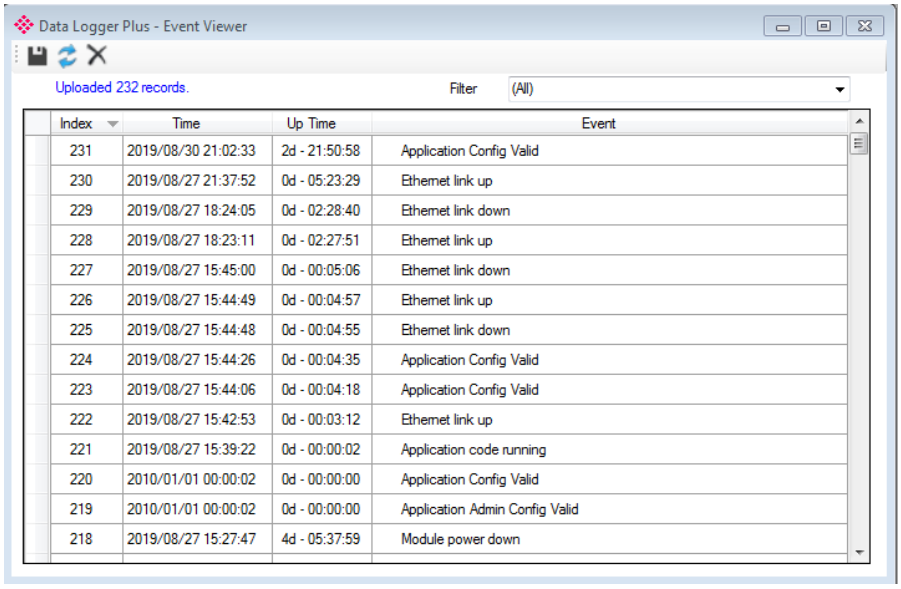


Figure 6.9 - Selecting the PLX51-DLplus-232 Event Log

2 The *Event Log* window opens and automatically reads all the events from the PLX51-DLplus-232.

3 The log entries are listed with the latest record at the top. Custom sorting is achieved by double-clicking on the column headings.



Index	Time	Up Time	Event
231	2019/08/30 21:02:33	2d - 21:50:58	Application Config Valid
230	2019/08/27 21:37:52	0d - 05:23:29	Ethernet link up
229	2019/08/27 18:24:05	0d - 02:28:40	Ethernet link down
228	2019/08/27 18:23:11	0d - 02:27:51	Ethernet link up
227	2019/08/27 15:45:00	0d - 00:05:06	Ethernet link down
226	2019/08/27 15:44:49	0d - 00:04:57	Ethernet link up
225	2019/08/27 15:44:48	0d - 00:04:55	Ethernet link down
224	2019/08/27 15:44:26	0d - 00:04:35	Application Config Valid
223	2019/08/27 15:44:06	0d - 00:04:18	Application Config Valid
222	2019/08/27 15:42:53	0d - 00:03:12	Ethernet link up
221	2019/08/27 15:39:22	0d - 00:00:02	Application code running
220	2010/01/01 00:00:02	0d - 00:00:00	Application Config Valid
219	2010/01/01 00:00:02	0d - 00:00:00	Application Admin Config Valid
218	2019/08/27 15:27:47	4d - 05:37:59	Module power down

Figure 6.10 – Module Event Log

4 The log can also be stored to a file for future analysis by selecting the **SAVE** button in the tool menu. To view previously saved files, use the *Event Log Viewer* option under the *Tools* menu.

6.4 Web Server

The PLX51-DLplus-232 provides a web server allowing you to view various diagnostics of the PLX51-DLplus-232 without the PLX50 Configuration Utility or RSLogix 5000. This includes Ethernet parameters, system event log, advanced diagnostics, and application diagnostics (DF1 diagnostics).

Note: The web server is read-only, no parameters or configuration can be altered from the web interface.

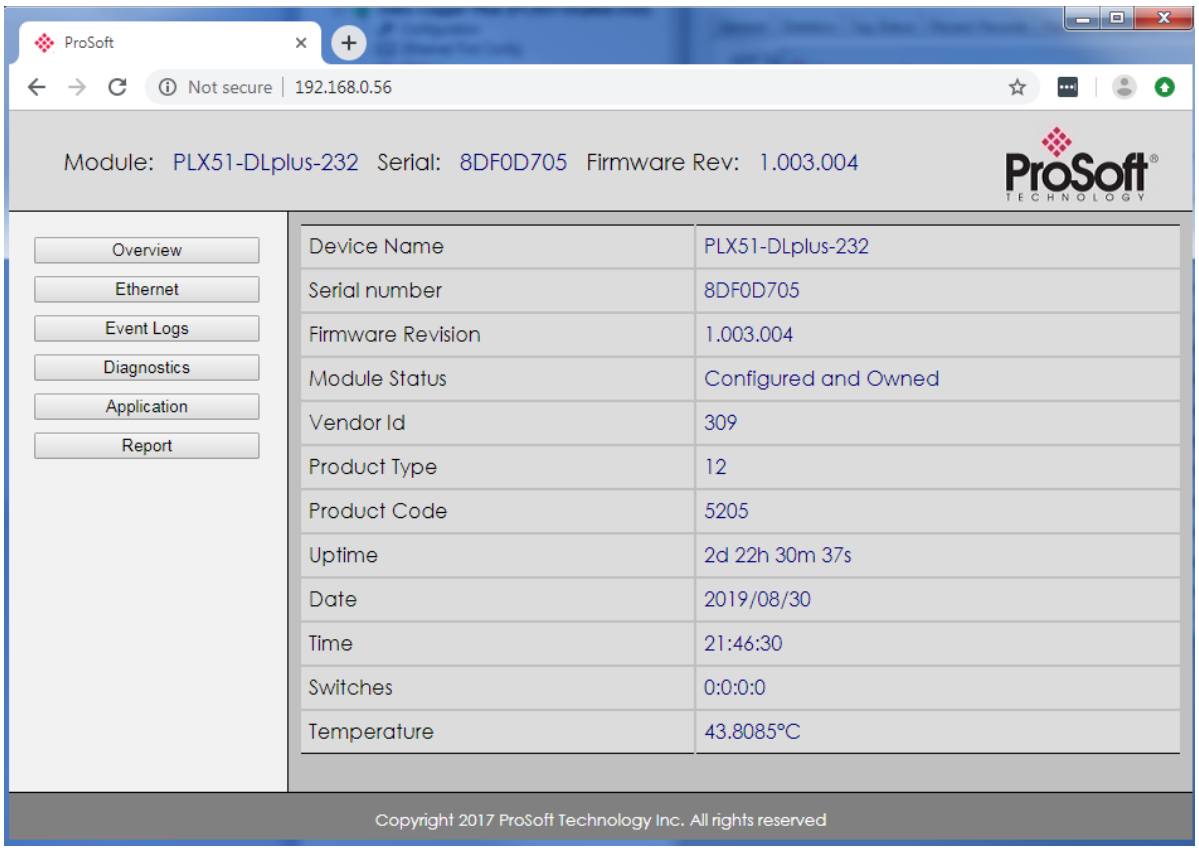


Figure 6.11 - Web interface

6.5 Web Reporting

The PLX51-DLplus-232 allows the user to extract logs and draw trends from the module using a standard web browser as shown below.

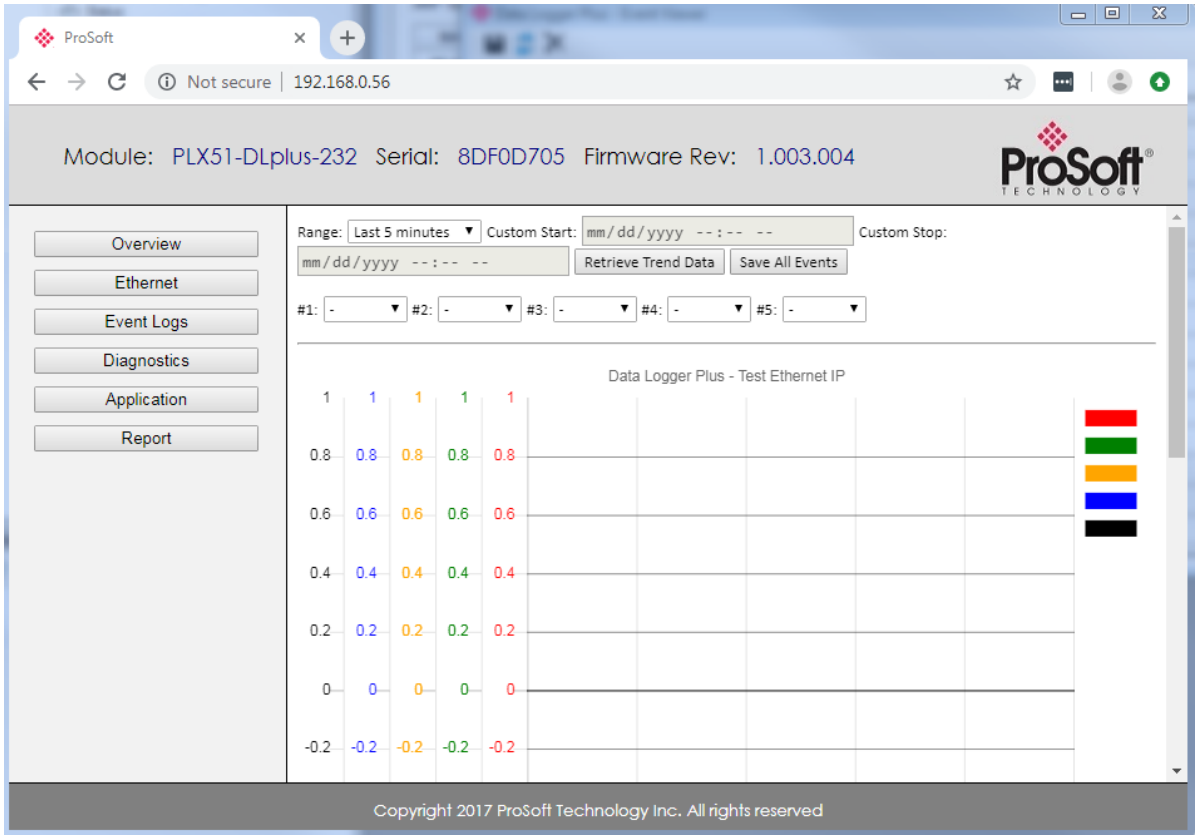


Figure 6.12 – Web reporting

The reporting page can be accessed via the standard web interface of the module by entering the IP address of the module into the browser and clicking on the **REPORT** button as shown below:

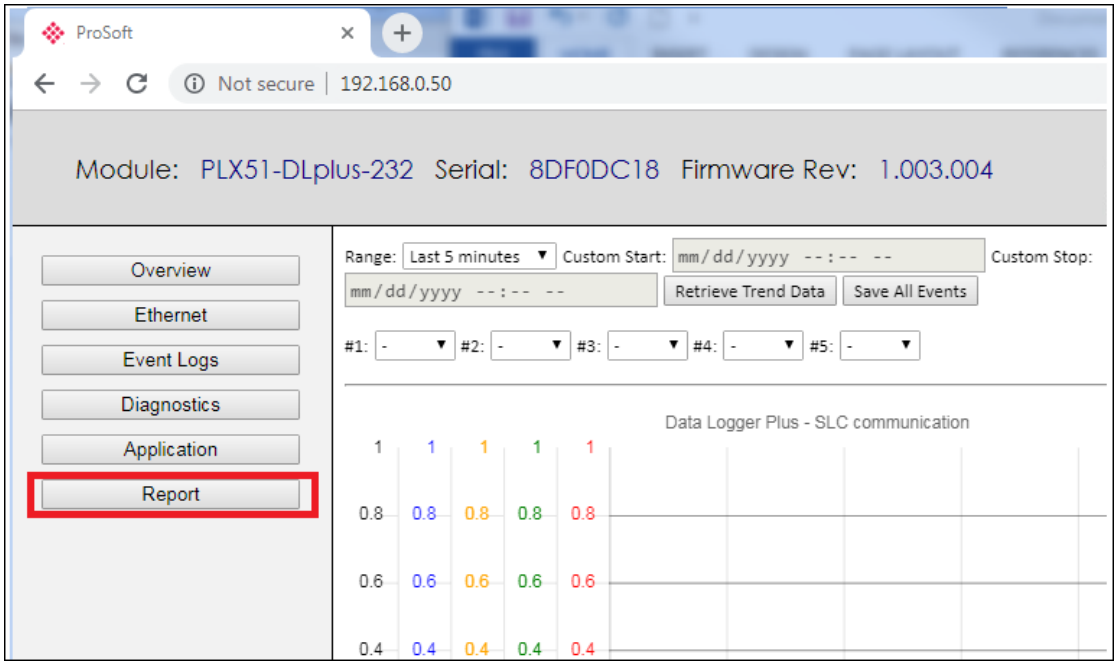


Figure 6.13 – Web report – access via normal webserver

Alternatively, the user can directly access the Report page by entering the IP address of the module into the browser followed by the report page url; *report.html* (eg. **http://192.168.1.xxx/report.html**). This allows the user to have full screen access of the trend, as shown below.

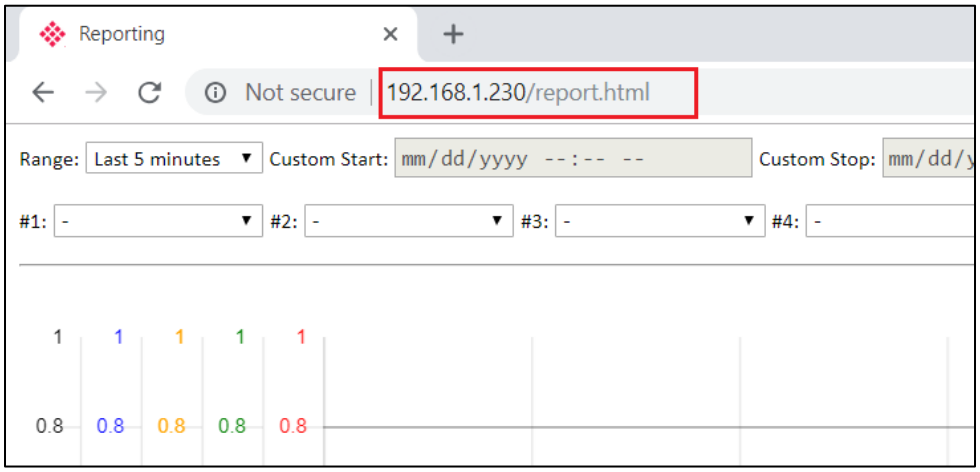


Figure 6.14 – Web report – access directly

The PLX51-DLplus-232 supports five trend objects (or pens) that can be used to trend logged data over a requested period. Each trend can be set to any of the configured tags (in the PLX50 Configuration Utility). The user can also upload all the records for a requested period to a CSV file via the web browser.

Each of the five trends can plot up to 10,000 records. Uploading data to CSV can store up to 1,000,000 records.

Important: If a new configuration has been downloaded to the module, the entire web page will need to be refreshed.

6.5.1 Retrieve Trend Data

When drawing a trend, the user must first select the tags that need to be trended.

1 Click on the drop-down box for each trend and selecting the tag that must be displayed.

Note: Selecting "-" results in no trend being drawn for that specific pen.

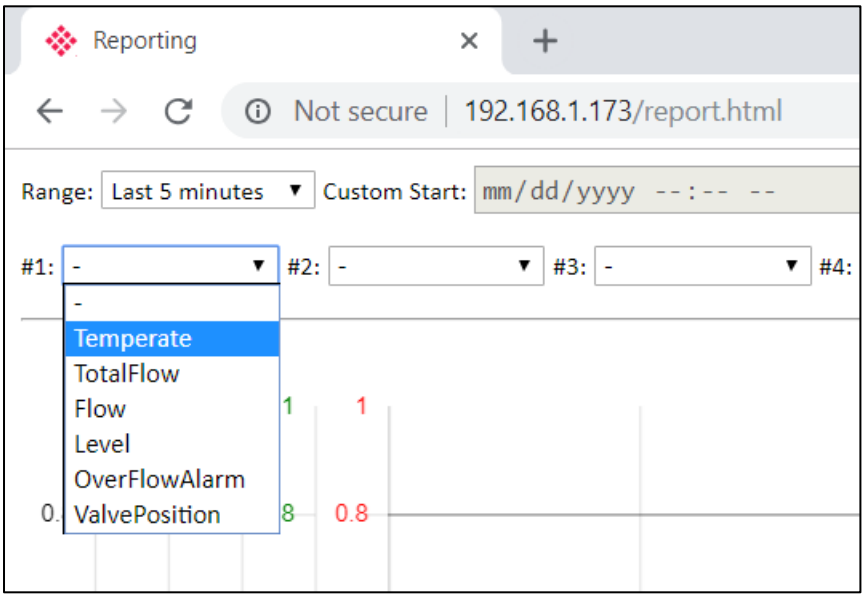


Figure 6.15 – Selecting data to trend

2 Once the required tags have been selected, the user will need to select the time period over which the data must be trended.

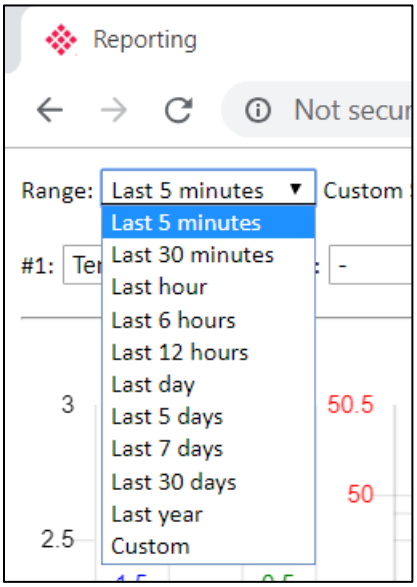


Figure 6.16 – Selecting time period

3 Press the *Retrieve Trend Data* button.

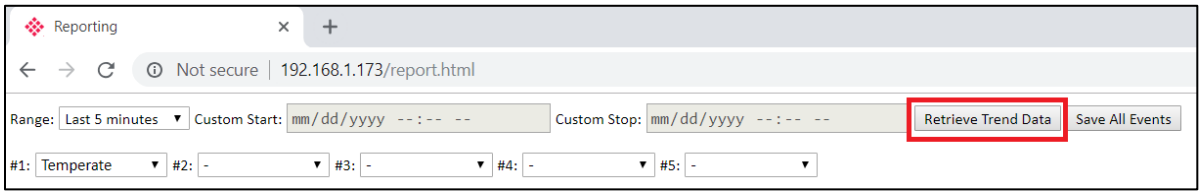


Figure 6.17 – Retrieve Trend Data

This will start collecting the required data from the PLX51-DLplus-232.

Figure 6.18 – Busy indication

4 Once the data has successfully been collected it will be trended on the graph as shown below:

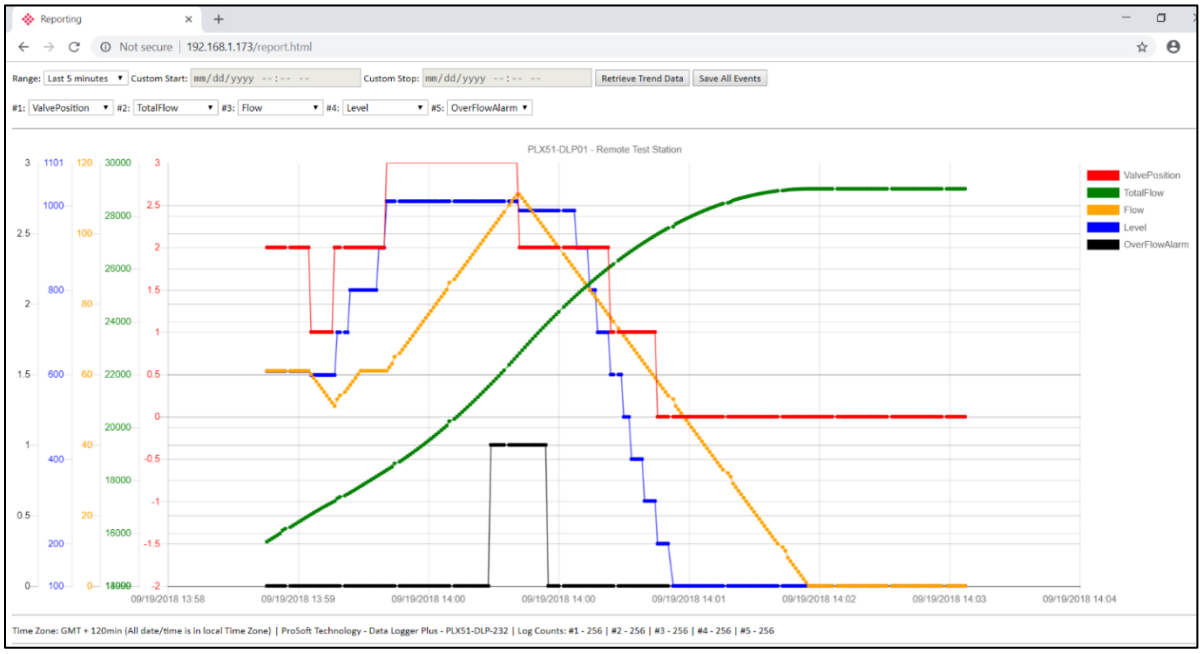


Figure 6.19 – Trends of the selected tags over the requested time period

Note: Depending on the amount of data that needs to be collected, it can take a few minutes to retrieve all the required data (each trend can have up to 10,000 data points).

5 The user can see how many points were plotted for each trend by viewing the *Log Counts* below the graph. In the example below, each of the five trends has 256 data points.

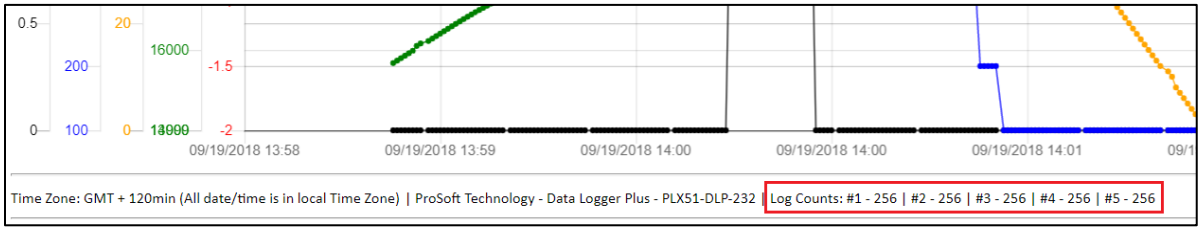


Figure 6.20 – Data point count for each trend

6 The user can also view the local time zone that will be applied to the UTC time retrieved for each record.

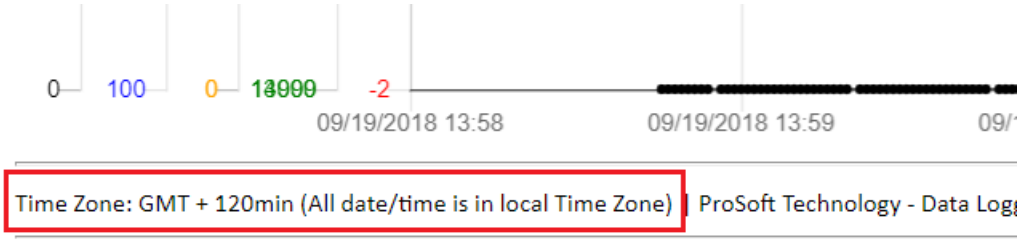


Figure 6.21 – Web browser local time zone

6.5.2 Customize Graph Axis

Each trend will have its own y-axis on the graph. The y-axis can be customized by scrolling down on the report page and selecting the *Manual* option for the specific y-axis.

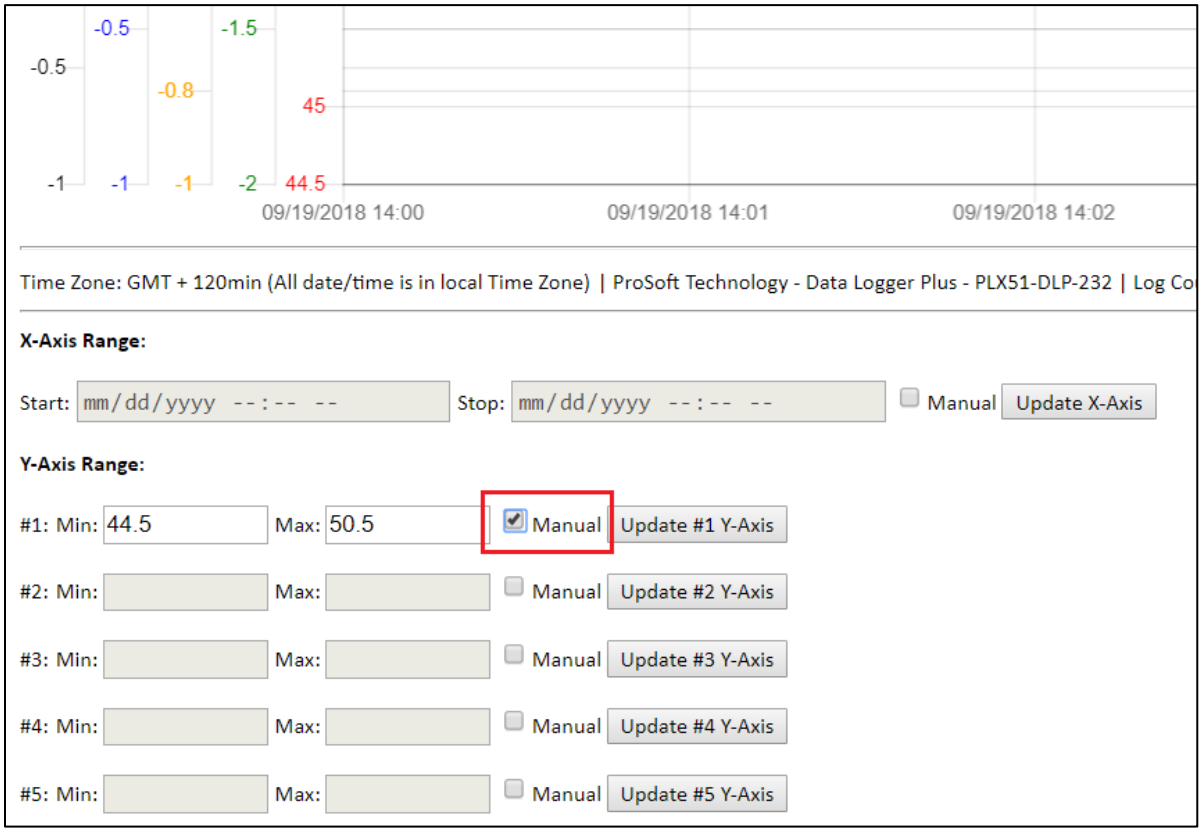


Figure 6.22 – Manual option for y-axis

Once selected, the max and min values of the current trend will be populated in the textboxes. The user can enter the max and min range for the specific trend and press the **UPDATE #X Y-AXIS** button. This will update the selected y-axis with the current loaded data.

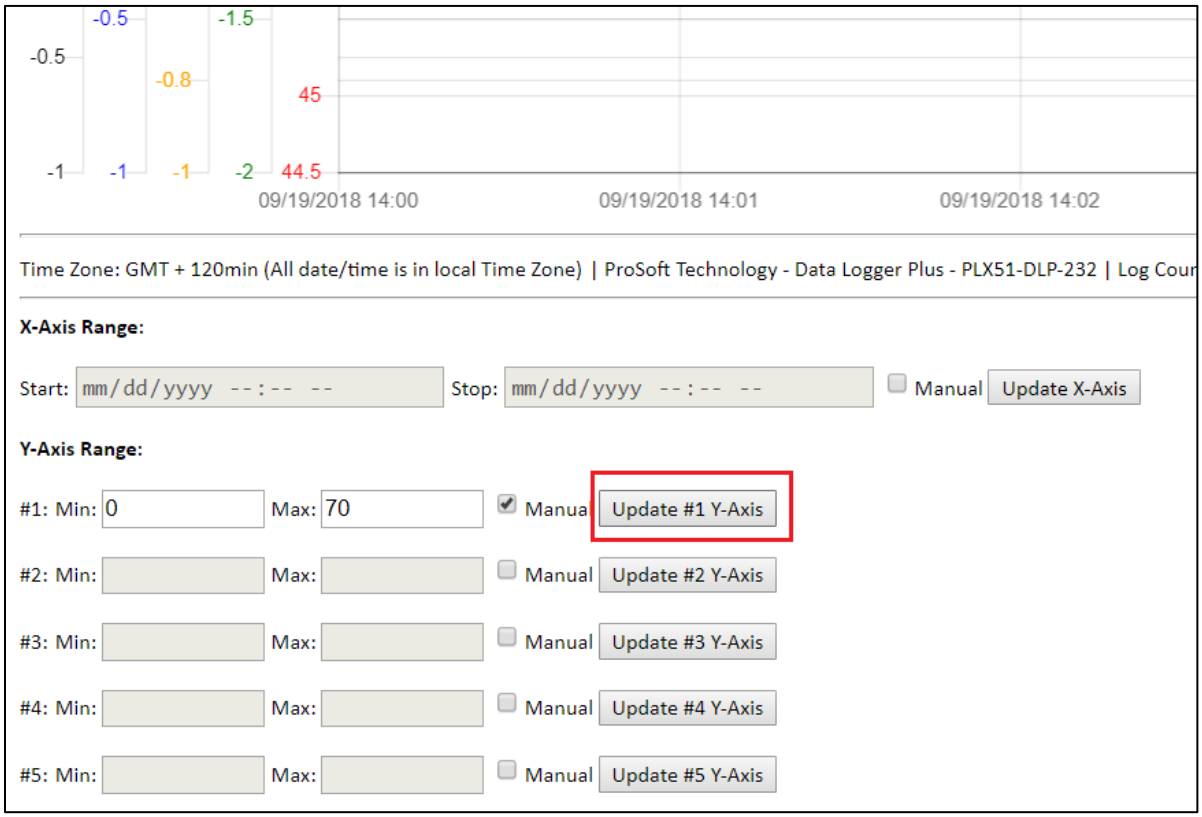


Figure 6.23 – Update Y-Axis

Note: Once the user selects to retrieve new data from the PLX51-DLplus-232, the current selection for the custom y-axis will be removed.

Note: By un-selecting the **Manual** option and pressing the **UPDATE #X Y-AXIS** button, the trend y-axis will return to the original scaled values

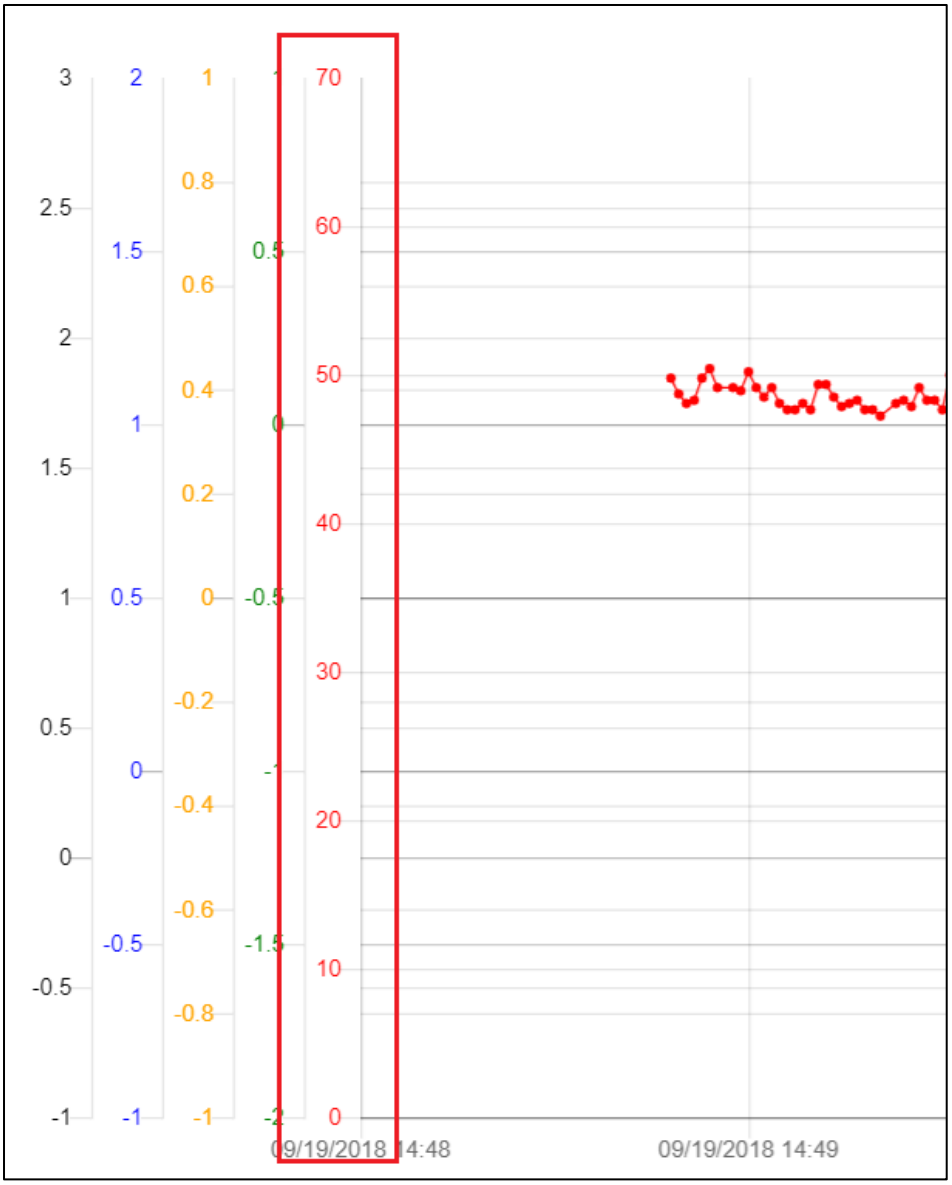


Figure 6.24 – Updated Y-Axis

Similarly, the x-axis of the graph can be updated for a specific time range. The user can select the **Manual** option on the X-Axis range (as shown below):

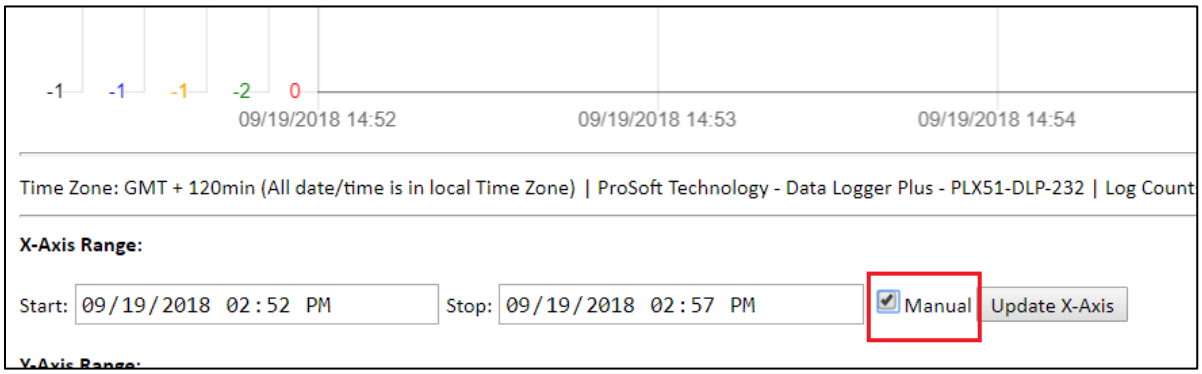


Figure 6.25 – Manual X-Axis selection

Once selected, the start and stop times of the current trend will be populated in the textboxes. The user can now enter the new custom time and press the **UPDATE X-AXIS** button.

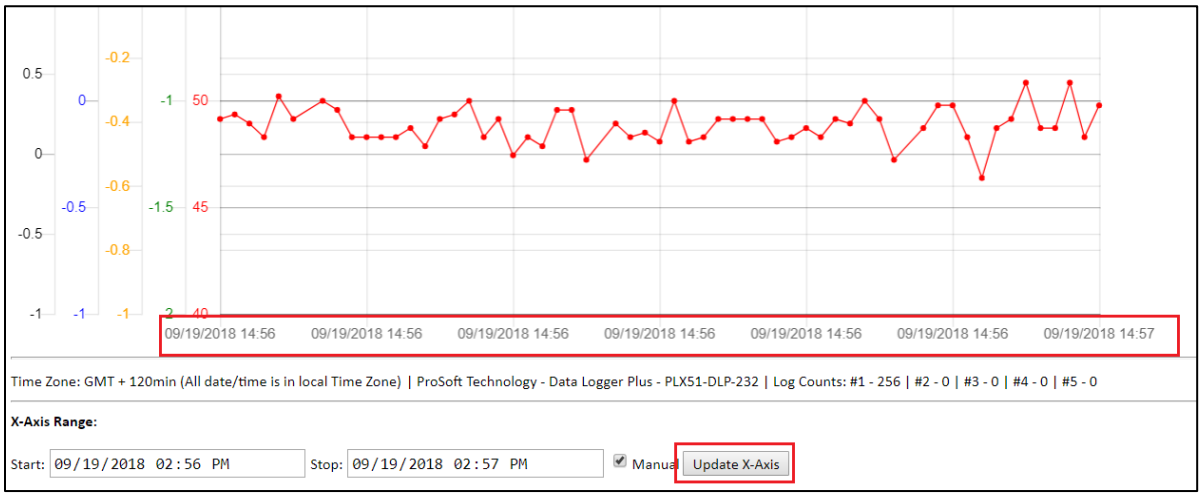


Figure 6.26 – Updated X-Axis

Note: Once the user selects to retrieve new data from the PLX51-DLplus-232, the current selection for the custom x-axis will be removed.

Note: By un-selecting the **Manual** option and pressing the **UPDATE X-AXIS** button, the trend y-axis will return to the original scaled values.

6.5.3 Upload Records to CSV File

The user can also upload all records for a requested time period to a CSV file. The user can then save and open this file in MS Excel. To create the CSV file, the user must select the required time range, followed by pressing the **SAVE ALL EVENTS** button.

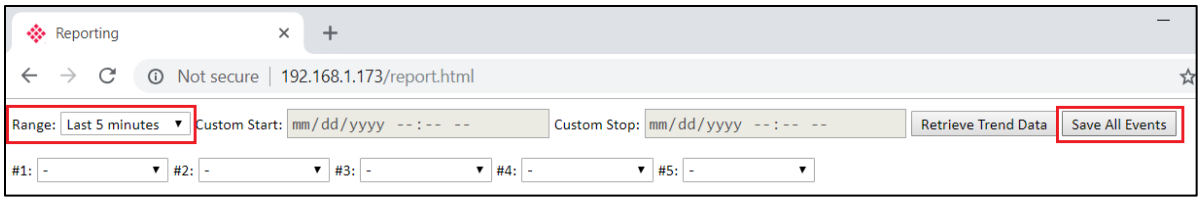


Figure 6.27 – Upload records to CSV file

Note: All tag records will be uploaded to the CSV file, and **NOT** the tags selected in the trend selections.

Note: Up to 1,000,000 records can be uploaded to a CSV file at a time. This can take several minutes to collect from the PLX51-DLplus-232.

Depending on the web browser, the created CSV file will be displayed in the web browser window:

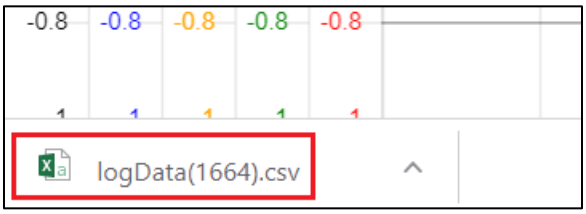


Figure 6.28 – Created CSV file

Note: The value in brackets () of the file name is the number of records stored in the CSV file. In the above example, there are 1664 records stored.

The file can be opened in MS Excel and will have the following Columns:

C2 9/19/2018 3:22:58 PM

	A	B	C	D	E	F	G	H
1	LogIndex	UTC DateTime	Local DateTime	MilliSecond	TagID	Tagname	DataType	DataValue
2	0	9/19/2018 13:22	9/19/2018 15:22	800	13	Flow	REAL	0
3	1	9/19/2018 13:22	9/19/2018 15:22	800	14	Level	REAL	100
4	2	9/19/2018 13:22	9/19/2018 15:22	800	15	OverFlowAlarm	SINT	0
5	3	9/19/2018 13:22	9/19/2018 15:22	800	16	ValvePosition	DINT	0
6	4	9/19/2018 13:22	9/19/2018 15:22	800	11	Temperate	REAL	48.0638
7	5	9/19/2018 13:22	9/19/2018 15:22	800	12	TotalFlow	REAL	29017
8	6	9/19/2018 13:22	9/19/2018 15:22	850	11	Temperate	REAL	48.2766
9	7	9/19/2018 13:22	9/19/2018 15:22	850	12	TotalFlow	REAL	29017
10	8	9/19/2018 13:22	9/19/2018 15:22	850	13	Flow	REAL	0

Figure 6.29 – CSV file opened in MS Excel

Table 6.3 – CSV file Parameters

Parameter	Description
Log Index	The number of the log index in this specific file.
UTC Date Time	The UTC (GMT + 0) time when the record was logged.
Local Date Time	The local time (GMT + local time zone) when the record was logged.
Millisecond	Millisecond when record was logged (50ms resolution).
Tag ID	The Tag ID of the specific tag.
Tagname	The tagname of the specific tag.
Data Type	The data type of the logged record.
Data Value	The data value of the logged record.

7 JSON Client

The PLX51-DLplus-232 module supports a number of status- and record-based JSON queries that can be used to extract logged records from the module. The module supports REST API to collect JSON encoded data from the module that can be utilized in other systems. The JSON requests and responses are encapsulated in an HTTP GET or POST command.

You can download a sample Python JSON Client script from www.prosoft-technology.com.

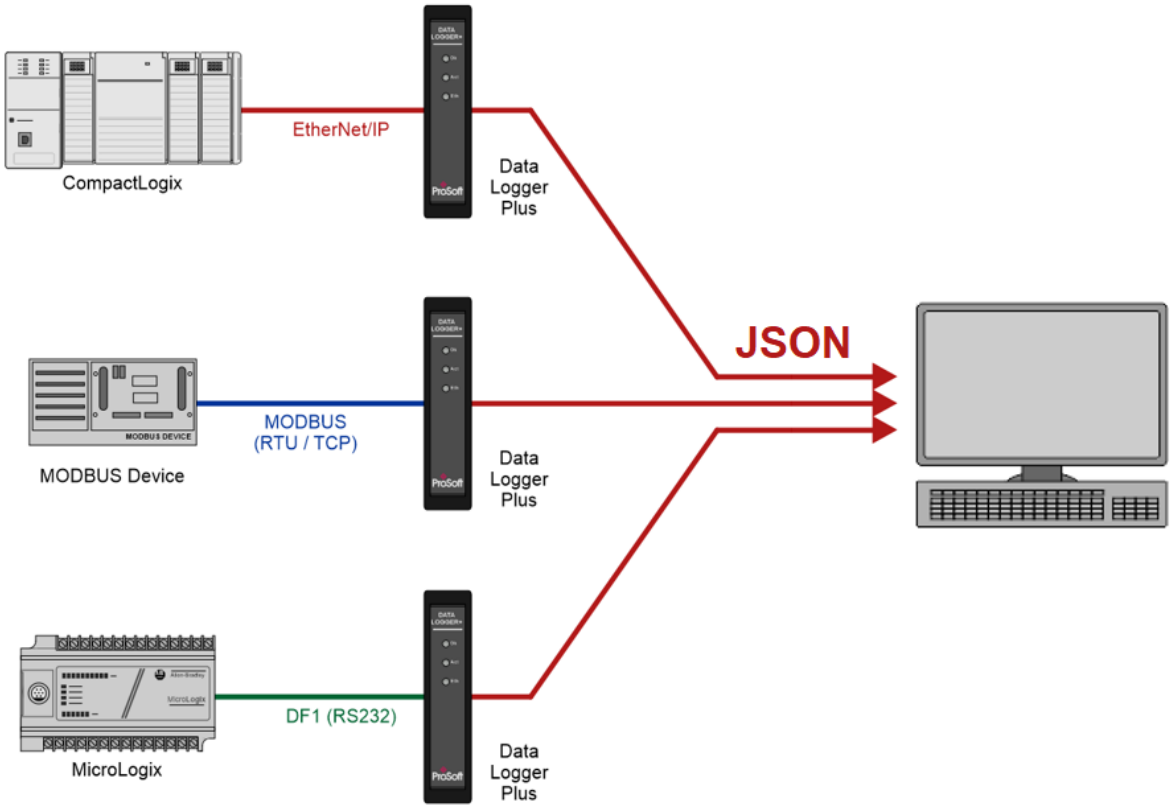


Figure 7.1 – Typical data source options for JSON interfacing.

7.1 General Status

There are two methods of extracting saved records from the PLX51-DLplus-232 using the JSON interface:

- Cached Record Method
- Trend Data Method

7.1.1 Cached Record Method

When using the *Cached Record* extraction, the user will need to track the load and unload log index and manually request records at a specific log index.

The log index is managed by the module and incremented each time a new record is stored. The unload index is managed externally by the unload service and should only incremented after a record has been logged successfully to a Database or text file. Both the Log Index and Unload Indices will loop around reaching the end of the cache. The cache becomes 100% full when the log index loops around and catches up with the unload index. In this situation, either older records are overwritten (Log Mode = Overwrite) or newer records are not logged (Log Mode = Hold).

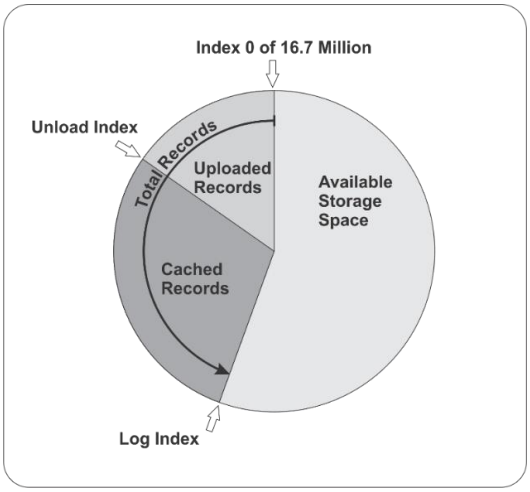


Figure 7.2 – Load and Unload Index Record Management

The user needs to first read the *General Status* to obtain the load and unload log indexes. Once the user has these values, the user can start reading records, using the *Cached Records*, at the unload log index. The *Unload Log Index Update* must be used to mark those records as “unloaded”.

7.1.2 Trend Data Method

When using the *Trend Data* extraction, the user will need to provide the timeline which to extract data from, and select either up to five specific tags, or all tags. The PLX51-DLplus-232 module will then provide the user with the specific requested data using a poll strategy.

It is recommended that the user first read the *General Status* to obtain the configuration checksum (parameter *ConfigCRC*) which is used to validate the current tag name list. This approach ensures the detection of new configuration being downloaded whilst the trend data is being extracted.

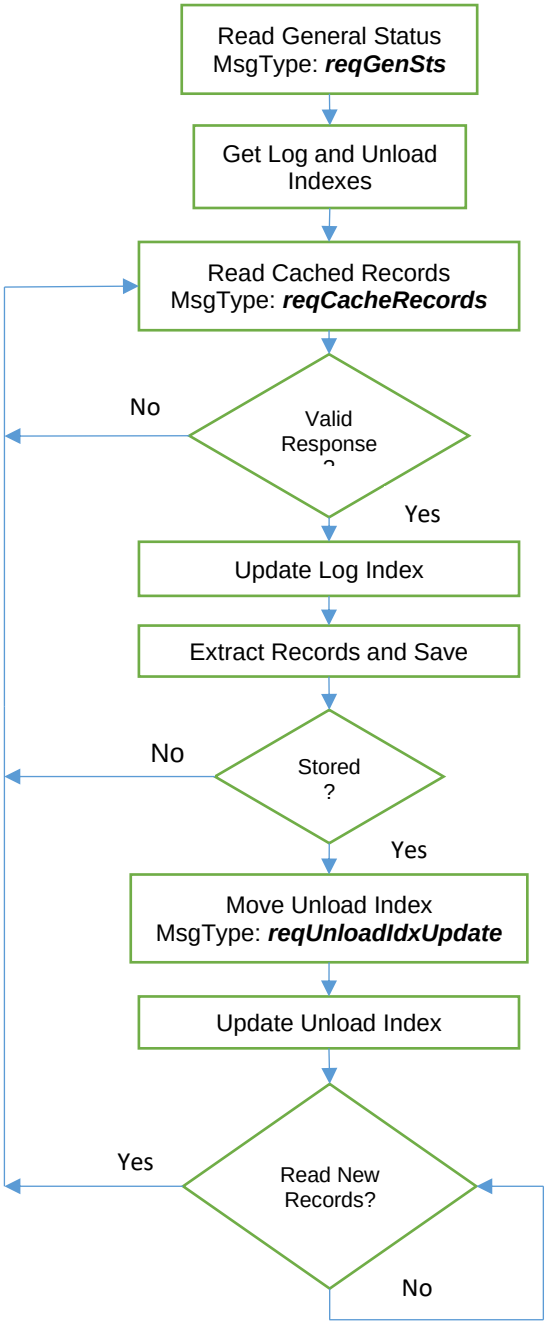
Once this *ConfigCRC* has been requested and stored, the user can then use the *Get Tag Names* function to extract the configured data points for the specific source, which will include the tag Id used to extract and match the records in the trend extraction. Once the user can then use the *Trend Data* to extract either specific or all records for a specific timeline.

7.2 Workflow

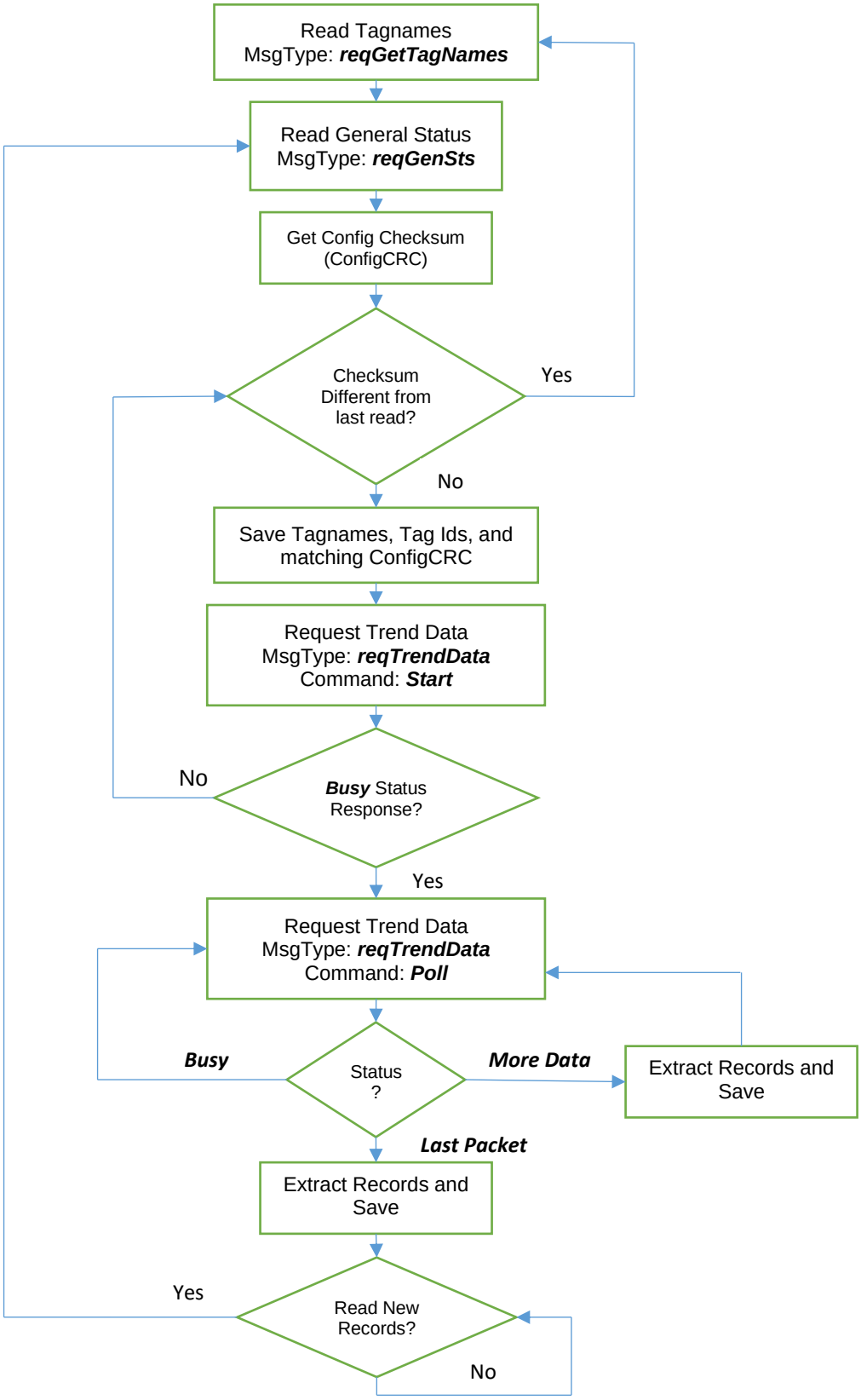
The workflow for extracting records using the Cached Record Method and Trend Data Method are illustrated below.

See the JSON Message Description for more details regarding the specific data in each request and response.

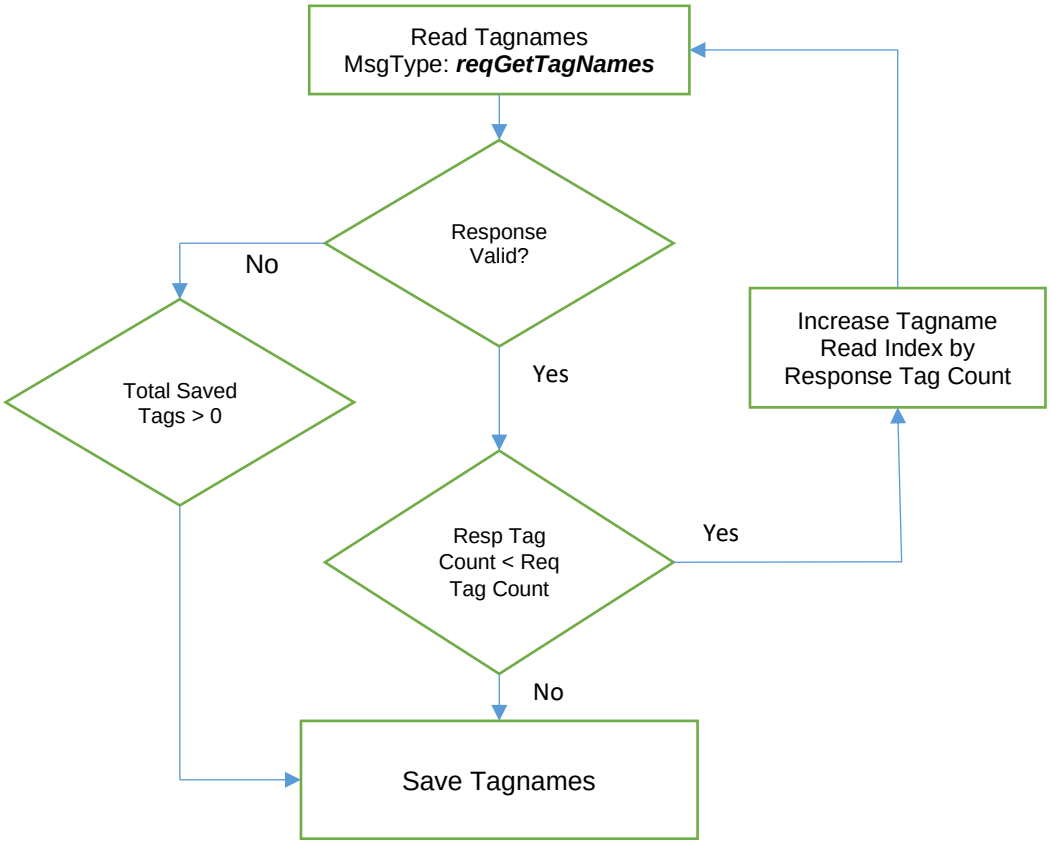
7.2.1 Cached Record Method



7.2.2 Trend Data Method



7.2.3 Tag Name Extraction



7.3 Operation

There are multiple ways to extract data from the PLX51-DLplus-232. The following operations are described in the form of pseudocode examples.

7.3.1 Get Tag Database

Before reading the tags, it is recommended to first read the configuration CRC. If

the tag configuration changes during the tag upload, the upload process can be repeated.

```
crcStart = reqGenSts().ConfigCRC
tagIndex = 0
Loop
      reqGetTagNames(tagIndex,7)
      tagIndex = tagIndex +7
crcEnd = reqGenSts().ConfigCRC
if(crcEnd != crcStart) then repeat
```

7.3.2 Get Trend Data

The *Get Trend Data* request extracts up to 5 specific tags, or all of them, between a start and stop time. The interpolation and record index searched are managed by the PLX51-DLplus-232.

```
      status = reqTrendData("start", duration, startDT, stopDT).Status
while(status != LastPacket)
      status = reqTrendData("poll", duration, startDT, stopDT).Status
```

7.3.3 Get Cache Records

The *Get Cache Records* request can extract subsets of records by using the record index. This is a low-level approach where the application is expected to manage the record indices.

The PLX51-DLplus-232 stores records in a 16 million record circular buffer, using a LogIndex and

UnloadIndex. Each time a record is logged, the LogIndex is incremented.

The UnloadIndex is typically controlled by the PLX50 Configuration Utility, and is used to determine the records that have not yet been uploaded.

To unload records, the application should first read the GeneralStatus to determine the LogIndex, UnloadIndex and RollOver flag. The records are then read out, 16 at a time.

If the application requires these records to be flagged as “Read”, the application should use the UnloadLogIndexUpdate command at the end of the read cycle.

If the requirement is to unload all the events, then it is not necessary to consider the UnloadIndex.

If the RollOver flag is set, then the Unload can start at the LogIndex+1. Otherwise, it starts at index 0. The unloading stops when the index reaches the LogIndex. If the RollOver flag is set, the index should rollover after 16777215.

7.4 JSON Message Details

The JSON requests and associated responses are summarized in the table below.

Figure 7.1 – Load and Unload Index Record Management

Section	Request	Response	Description
General Status	reqGenSts	resGenSts	Request the general status from the module.
Statistics	reqCacheStats	resCacheStats	Request the statistics for the logged records.
Unload Index	reqUnloadIdxUpdate	resUnloadIdxUpdate	Request the module to move the unload index to mark records as unloaded.
Records	reqCacheRecords	resCacheRecords	Request a set of records that has been logged in the module.
Reset	reqLogIndexReset	resLogIndexReset	Request the module to reset all the log indexes. Any unloaded records will be lost when this request is sent.
Tagnames	reqGetTagNames	resGetTagNames	Request a set of tag names that have been configured for logging in the module.
Trend Data	reqTrendData	resTrendData	Request a set of records that has been logged in the module using a specific timeline (when using a custom date, it will be in Gregorian format). Timestamps are based on the GMT/UTC time zone and no local time zone translation is provided. REST API calls result in resCacheRecords being returned by the device.
Trend Data (UTC)	reqTrendDataUTC	resTrendDataUTC	Request a set of records that has been logged in the module using a specific timeline (when using a custom date, it will be in Unix epoch format - number of seconds since January 1, 1970). Timestamps are based on the GMT/UTC time zone and no local time zone translation is provided. REST API calls result in resCacheRecords being returned by the device.
Invalid Request	(Any)	resInvalidRequest	Invalid request.

7.4.1 General Status

Used to request the general status of the module.

Request

Message: reqGenSts
Parameters: None
Example:

```
{
	"header": {
		"messageType": "reqGenSts"
	},
	"requestData": {}
}
```

Response

Message: resGenSts
Parameters:

Table 7.2 – resGenSts - Response Data Descriptions

Response Data	Description
Running	Set if the module has a valid configuration and is reading tags.
ConfigValid	Set if a valid configuration is executing in the module.
ContinousLogging	Set if Logging Mode set to Overwrite, clear for Hold.
Rollover	Set if the log index has reached the maximum 32bit number (0xFFFFFFFF) and rolled over back to zero.
LoggingInhibited	Set if logging has been inhibited by the Logix controller.
LoggingStopped	Set if logging has stopped because the maximum amount of records has been reached and *ContinuousLogging* has been set to *Hold*.
ConfigCRC	The downloaded configuration checksum. This value will be used to ensure that the tag names and ids in the module is valid since the last update.
LogIndex	The current record index being written to.
UnloadIndex	The upload record index.
DataSource	The current data source of the records to be read and logged. **EtherNet/IP** The current source is an EtherNet/IP device (eg. Logix controller). **DF1** The current source is a DF1 device (eg. PLC5 controller). **Modbus RTU** The current source is a Modbus RTU device (RS232). **Modbus TCP/IP** The current source is a Modbus TCP/IP device.
currentDateTime	The module's current time displayed in Gregorian Format.
currentUTC	The module's current time displayed in UTC Format (number of seconds since January 1, 1970).
serialNum	The serial number of the module.
instance	The instance name of the module. This is set in the module configuration in the PLX50CU and downloaded to the module.

Example:

```
{
	"header": {
		"messageType": "resGenSts"
	},
	"responseData": {
		"Running": 1,
		"ConfigValid": 1,
		"ContinousLogging": 1,
		"Rollover": 0,
		"LoggingInhibited": 0,
		"LoggingStopped": 0,
		"ConfigCRC":1234,
		"LogIndex":1,
		"UnloadIndex":0,
		"DataSource": "EtherNet/IP"
	}
}
```

Data Source Values: EtherNet/IP, DF1, Modbus RTU, Modbus TCP/IP

7.4.2 Cache Statistics

Used to request the statistics of logged records.

Request

Message: reqCacheStats
Parameters: None
Example:

```
{
	"header": {
		"messageType": "reqCacheStats"
	},
	"requestData": {}
}
```

Response

Message: resCacheStats
Parameters:

Table 7.3 – resCacheStats - Response Data Descriptions

Response Data	Description
totalRecordCount	The total number of cached records uploaded or not.
cachedRecordCount	The number of cached records not yet uploaded.
cachePercentage	The number of cached records not yet uploaded, as a percentage of the total record capacity of 16,777,216.

Example:

```
{
	"header": {
		"messageType": "resCacheStats"
	},
	"responseData": {
		"totalRecordCount": 0,
		"cacheRecordCount": 0,
		"cachePercentage": 0
	}
}
```

7.4.3 Unload Log Index Update

Used to request the module to move the unload index to mark records as “unloaded”.

Request

Message: reqUnloadIdxUpdate
Parameters:

Table 7.4 – reqUnloadIdxUpdate - Request Data Descriptions

Request Data	Description
unloadLogIndex	The upload record index that must be written to the module. This issued to move the unload index indicating to the module that certain records have been read from the module and stored (e.g. into a database).

Example:

```
{
	“header”: {
		“messageType”: “reqUnloadIdxUpdate”
	},
	“requestData”: {
		“unloadLogIndex”: 0
	}
}
```

Response

Message: resUnloadIdxUpdate
Parameters: None
Example:

```
{
	“header”: {
		“messageType”: “resUnloadIdxUpdate”
	},
	“responseData”: {}
}
```

7.4.4 Cache Records

Used to request logged records from the module at a specified log index. A maximum of 5 records can be read at a time.

Important: When a PLX51-DLplus-232 is powered down, a certain amount of fragmentation can occur within the Non-volatile memory (due to the NAND NV memory page alignment). This can result in the data to be displayed as 0xFF (eg. the year will be 65535 for the record – 0xFFFF) in certain log indexes. If this value is received, ignore that specific log index.

Request

Message: reqCacheRecords
Parameters:

Table 7.5 – reqCacheRecords - Request Data Descriptions

Request Data	Description
logIndex	The index where module must start reading the records which will be sent back to the JSON client in the response.
recordCount	The number of records that must be read (note a maximum of 5 records can be read at a time).

Example:

```
{
	"header": {
		"messageType": "reqCacheRecords"
	},
	"requestData": {
		"logIndex": 0,
		"recordCount": 0
	}
}
```

Response

Message: resCacheRecords
Parameters:

Table 7.6 – resCacheRecords - Response Data Descriptions

Response Data	Description
reqLogIndex	The requested start index where module must start reading the records for this response.
currentLogIndex	The current record index being written to.
currentLogUnloadIndex	The upload record index.
Storage	This will indicate if the records are already stored in Non-volatile (NV) memory or if it is still in volatile RAM. **Non-volatile** The returned records have already been saved into NV memory. **RAM** The records are still in RAM.
records	**(repeated for each record returned)**
tsUTC	The timestamp for when this record was created in UTC format (number of seconds since January 1, 1970).
50msTick	The sub-second timestamp for when this record was created. This sub-second resolution is 50ms. This parameter is the number of 50ms ticks. For example, a returned value of 5 indicates 250ms.
tagId	The tag id of the record. This can be used to find the tag name of the tag when reading the tag names from the module with the *reqGetTagNames* message type.
datatype	The data type for the record. **REAL -** The data type is a 32-bit float. **SINT -** The data type is an 8-bit signed integer. **INT -** The data type is a 16-bit signed integer. **DINT -** The data type is a 32-bit signed integer. **BOOL -** The data type is a boolean.
checksum	The checksum for the current record. Used if the user wants to validate the record. The checksum is calculated by doing a CRC on the record (with the CRC value set to zero) using a polynomial of 0x1DCF. Note that this is not required, see the section on the record checksum calculation.
data	The record data that was logged.

Example:

```
{
	"header": {
		"messageType": "resCacheRecords"
	},
	"responseData": {
		"reqLogIndex": 0,
		"currentLogIndex": 0,
		"currentLogUnloadIndex": 0,
		"storage": "Non-volatile",
		"records": [
			{
				"tsUTC": 1553401472,
				"50msTick": 3,
				"tagId": 12,
				"dataType": "REAL",
				"checksum": 5555,
				"data": 12.345
			},
			{
				"tsUTC": 1553401472,
				"50msTick": 5,
				"tagId": 14,
				"dataType": "DINT",
				"checksum": 4444,
				"data": 98765
			}
		]
	}
}
```

Storage Values: Non-volatile, RAM

7.4.5 Reset Log Indexes

Used to request the module to reset all the log indexes. Any unloaded records will be lost when this request is sent.

Request

Message: reqLogIndexReset
Parameters: None
Example:

```
{
	"header": {
		"messageType": "reqLogIndexReset"
	},
	"requestData": {}
}
```

Response

Message: resLogIndexReset
Parameters: None
Example:

```
{
	"header": {
		"messageType": "resLogIndexReset"
	},
	"responseData": {}
}
```

7.4.6 Get Tag Names

Used to request the set of tag names that have been configured for logging in the module. A maximum of 4 tag names can be read at a time.

Request

Message: reqGetTagNames
Parameters:

Table 7.7 – reqGetTagNames - Request Data Descriptions

Request Data	Description
tagNameIndex	The starting index for the tag names that must be returned.
tagNameCount	The number of tag names that must be returned in the response. Note a maximum of 4 tag names can be read at a time.

Example:

```
{
	"header": {
		"messageType": "reqGetTagNames"
	},
	"requestData": {
		"tagNameIndex": 7,
		"tagNameCount": 2
	}
}
```

Response

Message: resGetTagNames
Parameters: None
Example:

```
{
	"header": {
		"messageType": "resGetTagNames"
	},
	"responseData": {
		"tagNameCount": 2,
		"tags": [
			{
				"tagDataType": "REAL",
				"tagId": 14,
				"tagName": "Outlet Flow 01"
			},
			{
				"tagDataType": "DINT",
				"tagId": 15,
				"tagName": "High Level Count"
			}
		]
	}
}
```

7.4.7 Trend Data

Used to request a set of records that has been logged in the module using a relative or specific timeline. When using a custom date, the date will be specified in Gregorian format. A maximum of 5 records can be read at a time.

Important: When a module is powered down a certain amount of fragmentation can occur with the Non-volatile memory (due to the NAND NV memory page alignment). This can result in the data to be displayed as 0xFF (eg. the year will be 65535 for the record – 0xFFFF) in certain log indexes. If this value is received, ignore that specific log index.

Request

Message: reqTrendData
Parameters:

Table 7.8 – reqTrendData - Request Data Descriptions

Request Data	Description
command	This is the command value that is used to determine what action the module must take with the current request. **Start** The *Start* command is used to start the Trend Data extraction process. This command must be sent the first time during the transaction. **Poll** Once the *Start* command has been sent, the JSON client will need to poll the module to extract the required logged records. The user will send the poll command until the status indicates *Last Packet*. Note: Sending a new *Start* command before the last packet has been Polled, will effectively Abort the previous set of results.
Duration	The user can ask for specific pre-configured durations by setting this duration parameter. **Last 5 min -** Retrieve records logged in the last 5 minutes. **Last 30 min** - Retrieve records logged in the last 30 minutes. **Last hour** - Retrieve records logged in the last 1 hour. **Last 6 hours** - Retrieve records logged in the last 6 hours. **Last 12 hours** - Retrieve records logged in the last 12 hours. **Last day** - Retrieve records logged in the last day. **Last 5 days** - Retrieve records logged in the last 5 days. **Last week** - Retrieve records logged in the last week. **Last month** - Retrieve records logged in the last month. **Last year** - Retrieve records logged in the last year. **Custom Dates** - The *startTime* and *stopTime* parameters below will be used to specify the time period from which the records must be extracted.
startTime	The start time for the records extraction period. The time is in Gregorian format: *yyyy/mm/dd hh:mm:ss*
stopTime	The stop time for the records extraction period. The time is in Gregorian format: *yyyy/mm/dd hh:mm:ss*
extractedTags	The user can select to either return all tags for the requested time period, or up to 5 specific tags.

Request Data	Description
	All tags All records logged during the requested time period will be extracted. **Tags 1 to 5** Only records matching the tag Ids in the below parameters will be extracted.
tag Idx 1	First Tag Id that will be extracted.
tag Idx 2	Second Tag Id that will be extracted.
tag Idx 3	Third Tag Id that will be extracted.
tag Idx 4	Forth Tag Id that will be extracted.
tag Idx 5	Fifth Tag Id that will be extracted.

Example:

```
{
	"header": {
		"messageType": "reqTrendData"
	},
	"requestData": {
		"command": "Start",
		"duration": "Last 5 min",
		"startTime": "2019/03/02 14:22:00",
		"stopTime": "2019/03/02 14:29:00",
		"extractedTags": "Tags 1 to 5",
		"tag Idx 1": 12,
		"tag Idx 2": 19,
		"tag Idx 3": 24,
		"tag Idx 4": 27,
		"tag Idx 5": 28
	}
}
```

Response

Message: resTrendData
Parameters: None
Example:

```
{
	"header": {
		"messageType": "resTrendData"
	},
	"responseData": {
		"status": "Last Packet",
		"recordCount": 2,
		"records": [
			{
				"tsUTC": 1553401472,
				"50msTick": 3,
				"tagId": 12,
				"dataType": "REAL",
				"checksum": 5555,
				"data": 12.345
			},
			{
				"tsUTC": 1553401472,
				"50msTick": 5,
				"tagId": 14,
				"dataType": "DINT",
				"checksum": 4444,
				"data": 98765
			}
		]
	}
}
```

7.4.8 Trend Data UTC

Used to request a set of records that has been logged in the module using a relative or absolute date and time. When using an absolute date-time, it must be specified in Unix epoch format as the number of seconds since January 1, 1970. A maximum of 5 records can be read at a time.

Important: When a module is powered down a certain amount of fragmentation can occur with the Non-volatile memory (due to the NAND NV memory page alignment). This can result in the data to be displayed as 0xFF (eg. the year will be 65535 for the record – 0xFFFF) in certain log indexes. If this value is received, ignore that specific log index.

Request

Message: reqTrendDataUTC
Parameters:

Table 7.9 – reqTrendDataUTC - Request Data Descriptions

Request Data	Description
command	This is the command value that is used to determine what action the module must take with the current request. **Start** The *Start* command is used to start the Trend Data extraction process. This command must be sent the first time during the transaction. **Poll** Once the *Start* command has been sent, the JSON client will need to poll the module to extract the required logged records. The user will send the poll command until the status indicates *Last Packet*.
Duration	The user can ask for specific pre-configured durations by setting this duration parameter. **Last 5 min -** Retrieve records logged in the last 5 minutes. **Last 30 min** - Retrieve records logged in the last 30 minutes. **Last hour** - Retrieve records logged in the last 1 hour. **Last 6 hours** - Retrieve records logged in the last 6 hours. **Last 12 hours** - Retrieve records logged in the last 12 hours. **Last day** - Retrieve records logged in the last day. **Last 5 days** - Retrieve records logged in the last 5 days. **Last week** - Retrieve records logged in the last week. **Last month** - Retrieve records logged in the last month. **Last year** - Retrieve records logged in the last year. **Custom Dates** - The *startTime* and *stopTime* parameters below will be used to specify the time period from which the records must be extracted.
startTime	The start time for the records extraction period. The time is in Unix epoch format (number of seconds since January 1, 1970).
stopTime	The stop time for the records extraction period. The time is in Unix epoch format (number of seconds since January 1, 1970).
extractedTags	The user can select to either return all tags for the requested time period, or up to 5 specific tags. **All tags** All records logged during the requested time period will be extracted.

Request Data	Description
	Tags 1 to 5 Only records matching the tag Ids in the below parameters will be extracted.
tag Idx 1	First Tag Id that will be extracted.
tag Idx 2	Second Tag Id that will be extracted.
tag Idx 3	Third Tag Id that will be extracted.
tag Idx 4	Forth Tag Id that will be extracted.
tag Idx 5	Fifth Tag Id that will be extracted.

Example:

```
{
	"header": {
		"messageType": "reqTrendDataUTC"
	},
	"requestData": {
		"command": "Start",
		"duration": "Last 5 min",
		"startUTC": 1553401472,
		"stopUTC": 1553401672,
		"extractedTags": "All tags",
		"tag Idx 1": 12,
		"tag Idx 2": 19,
		"tag Idx 3": 24,
		"tag Idx 4": 27,
		"tag Idx 5": 28
	}
}
```

Response

Message: resTrendDataUTC
Parameters: None
Example:


```
{
	"header": {
		"messageType": "resTrendDataUTC"
	},
	"responseData": {
		"status": "Last Packet",
		"recordCount": 2,
		"records": [
			{
				"tsUTC": 1553401472,
				"50msTick": 3,
				"tagId": 12,
				"dataType": "REAL",
				"checksum": 5555,
				"data": 12.345
			},
			{
				"tsUTC": 1553401472,
				"50msTick": 5,
				"tagId": 14,
				"dataType": "DINT",
				"checksum": 4444,
				"data": 98765
			}
		]
	}
}
```

7.4.9 Invalid Request Response

When a request message is received with incorrect or had illegal request parameters, the following response will be received.

Response

Message: resInvalidRequest
Parameters: None
Example:

```
{
      "header": {
            "messageType": "resInvalidRequest"
      },
      "responseData": {}
}
```

7.5 HTTP Examples

Below are examples extracted from Wireshark on two message request types.

7.5.1 General Status Message

Request

```
POST / HTTP/1.1
Host: 192.168.1.230
Connection: keep-alive
Content-Length: 55
Origin: http://192.168.1.230
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.103
Safari/537.36
Content-Type: application/json
Accept: */*
Referer: http://192.168.1.230/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9

{"header":{"messageType":"reqGenSts"},"requestData":{}}
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 364

{"header":{"messageType":"resGenSts"},"responseData":{"Running":
1,"ConfigValid": 1,"ContinousLogging": 1,"Rollover":
0,"LoggingInhibited": 0,"LoggingStopped": 0,"ConfigCRC":
5947,"LogIndex": 4487,"UnloadIndex": 0,"DataSource":
"EtherNet/IP","currentDateTime": "2019/04/30
11:42:51","currentUTC": 1556624571,"serialNum":
"35216C41","instance": "Data Logger Plus"}}
```

7.5.2 Cache Records Message

Request

```
POST / HTTP/1.1
Host: 192.168.1.230
Connection: keep-alive
Content-Length: 91
Origin: http://192.168.1.230
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.103
Safari/537.36
Content-Type: application/json
Accept: */*
Referer: http://192.168.1.230/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9

{"header":{"messageType":"reqCacheRecords"},"requestData":{"logIn
dex":100,"recordCount":4}}
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 720

{"header":{"messageType":"resCacheRecords"},"responseData":{"reqL
ogIndex": 100,"currentLogIndex": 559,"currentLogUnloadIndex":
0,"storage": "Non-volatile","records": [{"year": 2019,"mon":
4,"day": 30,"hour": 11,"min": 46,"sec": 12,"50msTick":
14,"tagId": 12,"dataType": "DINT","checksum": 42,"data":
555},{"year": 2019,"mon": 4,"day": 30,"hour": 11,"min": 46,"sec":
12,"50msTick": 14,"tagId": 13,"dataType": "DINT","checksum":
92,"data": 777},{"year": 2019,"mon": 4,"day": 30,"hour":
11,"min": 46,"sec": 12,"50msTick": 14,"tagId": 8,"dataType":
"DINT","checksum": 44,"data": 111},{"year": 2019,"mon": 4,"day":
30,"hour": 11,"min": 46,"sec": 12,"50msTick": 14,"tagId":
9,"dataType": "DINT","checksum": 133,"data": 222}]}}
```

8 Technical Specifications

8.1 Dimensions

Below are the PLX51-DLplus-232 enclosure and DIN rail dimensions. All dimensions are in millimeters.

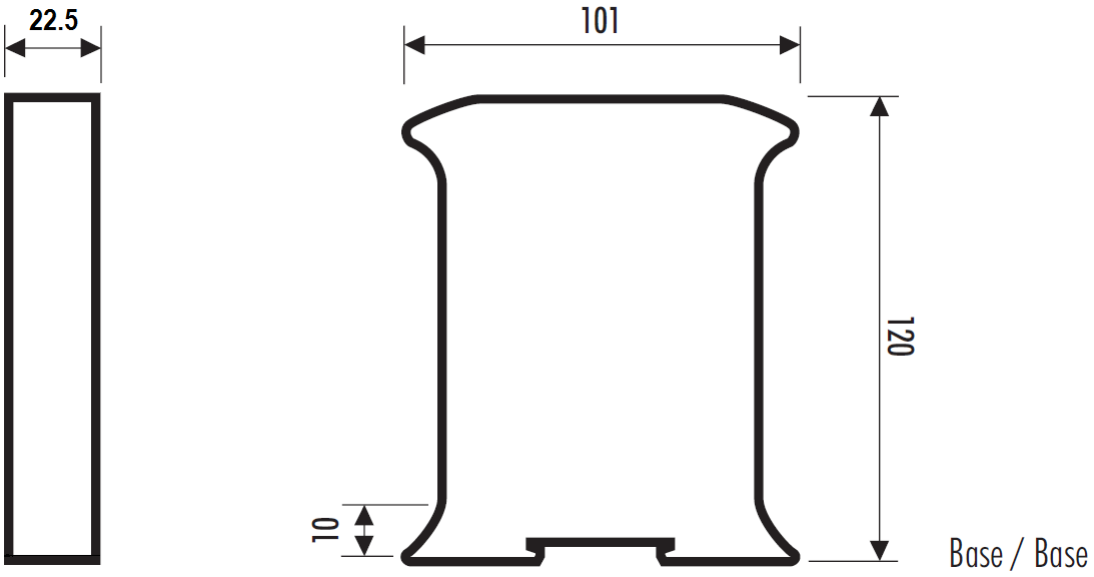


Figure 8.1 - Module enclosure dimensions

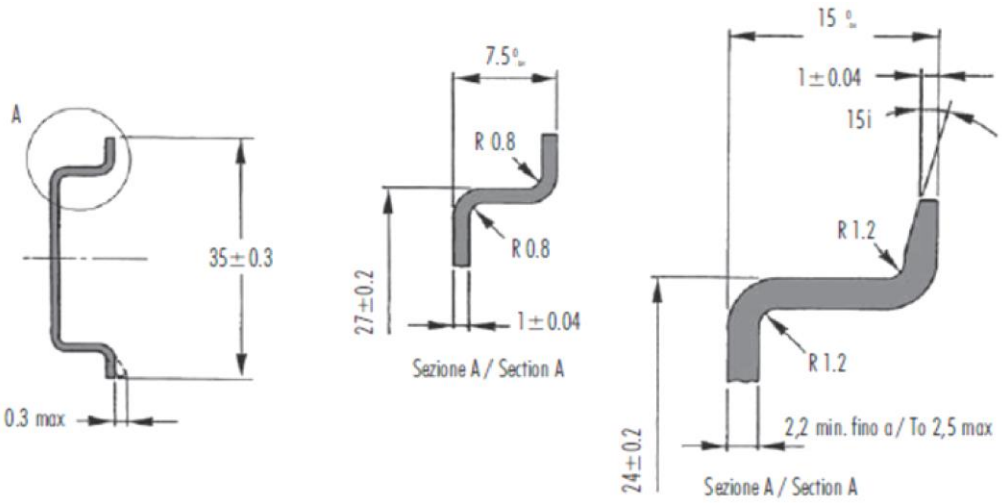


Figure 8.2 - Required DIN dimensions

8.2 Electrical

Table 8.1 - Electrical specification

Specification	Rating
Power requirements	Input: 10 to 28V DC, (70 mA @ 24 VDC)
Power consumption	1.7 W
Connector	3-way terminal
Conductors	24 to 18 AWG
Enclosure rating	IP20, NEMA/UL Open Type
Temperature	-20 to 70 °C
Earth connection	Yes, terminal based
Emissions	IEC61000-6-4
ESD Immunity	EN 61000-4-2
Radiated RF Immunity	IEC 61000-4-3
EFT/B Immunity	EFT: IEC 61000-4-4
Surge Immunity	Surge: IEC 61000-4-5
Conducted RF Immunity	IEC 61000-4-6

8.3 Ethernet

Table 8.2 - Ethernet specification

Specification	Rating
Connector	RJ45
Conductors	CAT5 STP/UTP
ARP connections	Max 20
TCP connections	Max 20
CIP connections	Max 10
Communication rate	10/100 Mbps
Duplex mode	Full/Half
Auto-MDIX support	Yes

8.4 Data Cache

Table 8.3 - Data Cache specification

Specification	Rating
Max Record Count	16,777,216
Maximum tag count	200
Log criteria supported	Delta change Heart beat Tag Triggers
Minimum Log Interval	50 ms
Data Types Supported	BOOL, SINT, INT, DINT, or REAL
Cached Records Non-Volatile	Yes

Specification	Rating
Log triggers supported	Yes
Data Sources	Logix Tags DF1 Files Modbus (RTU and TCP/IP) registers

8.5 Serial Port

Table 8.4 - Serial Port specification

Specification	Rating
Connector	4-way terminal
Conductor	24 to 18 AWG
Isolation voltage	2.5 kV
BAUD	1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
Parity	None, Even, Odd

8.6 DF1

Table 8.5 - DF1 specification

Specification	Rating
Duplex	Full/Half
Error detection	CRC, BCC
Embedded response	Auto, On

8.7 Modbus

Table 8.6 - Modbus specification

Specification	Rating
Supported Ports	Modbus RTU Modbus TCP/IP
Functions Supported	Read Coils (Function Code 1) Read Discrete Inputs (Function Code 2) Read Holding Register (Function Code 3) Read Input Register (Function Code 4)

9 Support, Service & Warranty

9.1 Contacting Technical Support

ProSoft Technology, Inc. (ProSoft) is committed to providing the most efficient and effective support possible. Before calling, please gather the following information to assist in expediting this process:

1 Product Version Number
2 System architecture
3 Network details

If the issue is hardware related, we will also need information regarding:

4 Module configuration and associated ladder files, if any
5 Module operation and any unusual behavior
6 Configuration/Debug status information
7 LED patterns
8 Details about the serial, Ethernet or fieldbus devices interfaced to the module, if any.

Note: For technical support calls within the United States, ProSoft Technology's 24/7 after-hours phone support is available for urgent plant-down issues.

North America (Corporate Location)	**Europe / Middle East / Africa Regional Office**
Phone: +1.661.716.5100 info@prosoft-technology.com Languages spoken: English, Spanish REGIONAL TECH SUPPORT support@prosoft-technology.com	Phone: +33.(0)5.34.36.87.20 france@prosoft-technology.com Languages spoken: French, English REGIONAL TECH SUPPORT support.emea@prosoft-technology.com
Latin America Regional Office	**Asia Pacific Regional Office**
Phone: +52.222.264.1814 latinam@prosoft-technology.com Languages spoken: Spanish, English REGIONAL TECH SUPPORT support.la@prosoft-technology.com	Phone: +60.3.2247.1898 asiapc@prosoft-technology.com Languages spoken: Bahasa, Chinese, English, Japanese, Korean REGIONAL TECH SUPPORT support.ap@prosoft-technology.com

For additional ProSoft Technology contacts in your area, please visit: www.prosoft-technology.com/About-Us/Contact-Us.

9.2 Warranty Information

For complete details regarding ProSoft Technology's TERMS & CONDITIONS OF SALE, WARRANTY, SUPPORT, SERVICE AND RETURN MATERIAL AUTHORIZATION INSTRUCTIONS, go to www.prosoft-technology.com/legal

Documentation is subject to change without notice.