

**The Ethernet/IP plugin
PRINTED MANUAL**

Ethernet/IP plugin

© 1999-2017 AGG Software

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Printed: 7/6/2017

Publisher

AGG Software

Production

© 1999-2017 AGG Software

<http://www.aggsoft.com>

Table of Contents

Part 1 Introduction	1
Part 2 System requirements	1
Part 3 Installing Ethernet/IP	2
Part 4 Glossary	3
Part 5 User Manual	3
1 Data query	3
2 Request method	5
3 AB Micro 800	5
4 AB MicroLogix 1400	7
Part 6 Troubles?	10
1 Possible problems	10

1 Introduction

This parser plugin allows to read tag values from ControlLogix, PLC 5, SLC 500 and MicroLogix controllers using EtherNet/IP protocol. The parser uses "Unconnected Messages" with PCCC-style commands. EtherNet/IP is an industrial network protocol that adapts the Common Industrial Protocol to standard Ethernet.

Features:

- Can send valid data request to any EtherNet/IP-compatible device;
- Symbolic or class/instance/attribute addressing;
- Supports various data types: STRING, BOOL, CONTROL, COUNTER, DINT, INT, LINT, REAL, SINT, TIMER, USINT, UINT, UDINT, ULINT, LREAL, STRING, DATETIME, DATE, TIME;
- Automatically detects a data type of returned data.
- Can read arrays.
- Can poll data by a custom interval.

2 System requirements

The following requirements must be met for "Ethernet/IP" to be installed:

Operating system: Windows 2000 SP4 and above, including both x86 and x64 workstations and servers. A latest service pack for the corresponding OS is required.

Free disk space: Not less than 5 MB of free disk space is recommended.

Special access requirements: You should log on as a user with Administrator rights in order to install this module.

The main application (core) must be installed, for example, Advanced Serial Data Logger.

Notes for Microsoft Vista and above:

Since our software saves data to the registry and installs to the Program Files folder, the following requirements must be met:

1. You need Administrator rights to run and install our software
2. The shortcut icon of our software will be located on the desktop;
3. Windows Vista will ask for your confirmation to continue the installation.

NOTE: You can configure the user account only once in order not to see the above dialog box any more. Search Google for the solution of this problem.

3 Installing Ethernet/IP

1. Close the main application (for example, Advanced Serial Data Logger) if it is running;
2. Copy the program to your hard drive;
3. Run the module installation file with a double click on the file name in Windows Explorer;
4. Follow the instructions of the installation software. Usually, it is enough just to click the "Next" button several times;
5. Start the main application. The name of the module will appear on the "Modules" tab of the "Settings" window if it is successfully installed.

If the module is compatible with the program, its name and version will be displayed in the module list. You can see examples of installed modules on fig.1-2. Some types of modules require additional configuration. To do it, just select a module from the list and click the "Setup" button next to the list. The configuration of the module is described below.

You can see some types of modules on the "Log file" tab. To configure such a module, you should select it from the "File type" list and click the "Advanced" button.

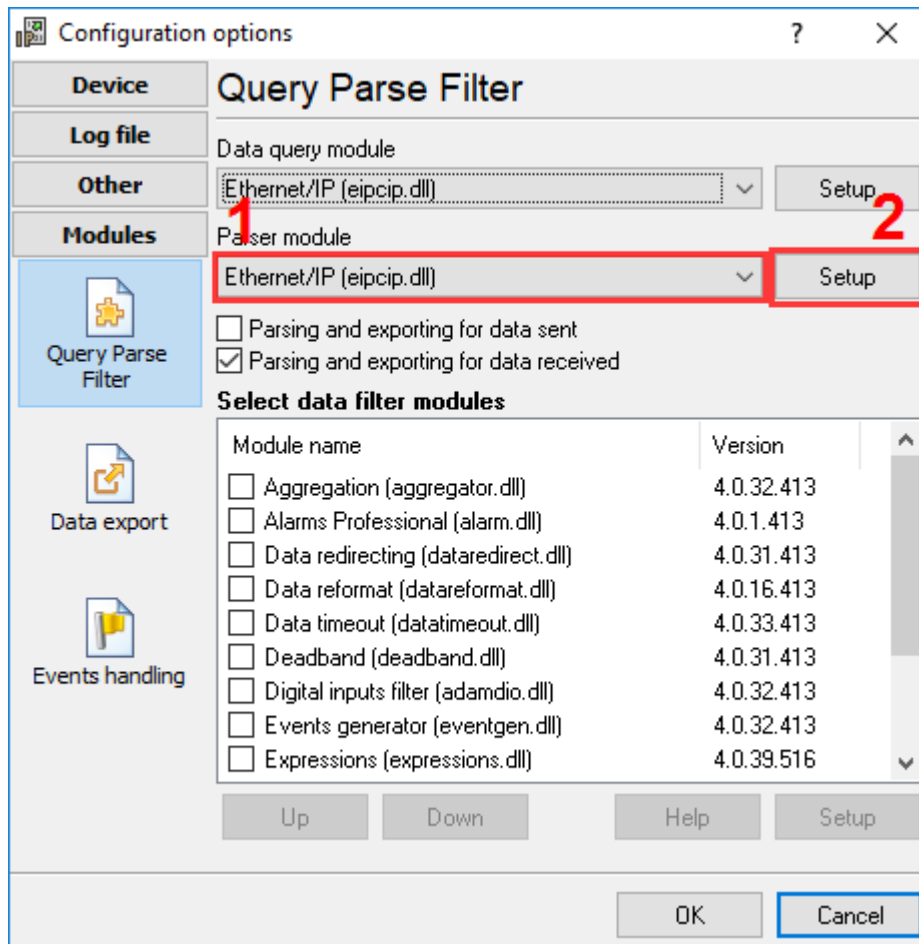


Fig.1. Example of installed module

4 Glossary

Plug-in - module

Main program – the program shell that uses this module. For example: Advanced Serial Data Logger

Parser – the module that processes the data flow singling out data packets from it and variables from data packets. These variables are used in data export modules after that.

Core - see "Main program".

5 User Manual

5.1 Data query

To add new item click "Actions->Add new request". The dialog window will be shown (fig.1). Enter a request description, that can contain any characters and click the "OK" button.

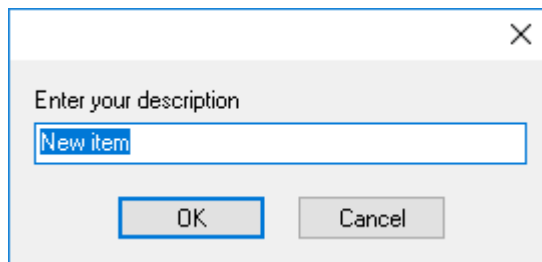


Fig.1. Name dialog

The new request will appear in the requests tree (fig.2). Each request has few important options:

- **Device path** - a path to your EIP compatible PLC in the Ethernet/IP network. Should be empty for direct connections;
- **Address** - it's an address (symbolic name) of a data item in the PLC's memory. If the item is an array then you may add an index in this array;

Examples:

VALUE1 - read data from the "VALUE1" tag.

VALUE1[2] - VALUE1 is an array and the program will read the 3rd element from this array (the array index starts from 0).

@22/1/1 - read data from class 22, instance 1 and attribute 1.

- **Number of elements to read** - for arrays you may read the specified number of sequential elements;
- **Request timeout** - after reaching the timeout limit the program will automatically cancel current request and execute next request in the queue. The timeout value depends on the network on

which master (program) and slave (device) is running. If the network is slow then timeout value should be larger and if network is fast then timeout value can be small.

- **Export name** - if this value is not empty the program will export the tag's value using this name. If the name is empty the the program will using the address as a name.
- **Scale** - if this parameters is not equal 1 then the program will scale a returned value using this coefficient;
- **Default value** - this value will be used if the parser can't get the specified values from a response.

Requests queue

Property	Value
Request #1	
<input checked="" type="checkbox"/> Send requests, otherwise parse response only	
Device path	1
Address (e.g. Tag[0] or @22/1...	Tag
Number of elements to read	1
Request timeout (ms)	3000
Request method	
<input type="radio"/> Once, on the program startup	
<input checked="" type="radio"/> Polling	
Interval (ms)	10
Interval units	Second
Export name	EN
Default value	0
Scale (numbers only)	2
Request #2	

Action Export data for all requests at once

Minimal interval between data packets (ms)

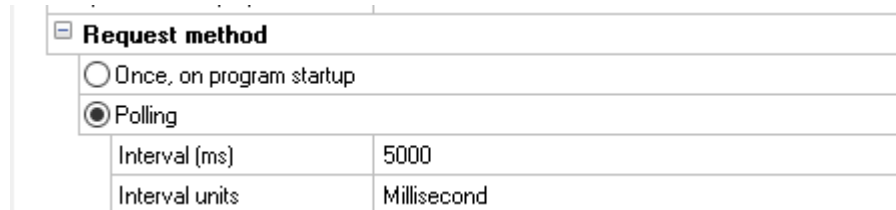
OK Cancel

Fig.2. Requests

5.2 Request method

The plugin can send requests in two modes:

- **Once, on program startup** - the program will send request once, when the program starts.
- **Polling** - the program will be sending request periodically based on an interval specified. The interval between requests depends on the network on which master (program) and slave (device) is running. If the network is slow then time for each request will be larger and vice versa. Because, the program are executing all requests in the queue one by one, then time between requests depends on the number of requests in the queue.



The screenshot shows a configuration window titled "Request method". It contains two radio buttons: "Once, on program startup" (unselected) and "Polling" (selected). Below the radio buttons is a table with two rows and two columns:

Interval (ms)	5000
Interval units	Millisecond

Fig.3. Request methods

If you added few requests to the queue, then you can move it up and down. Select a request title and execute a corresponding menu item by clicking the "Actions" button.

With help of this button you can change an item description and delete requests.

You can access all actions through the popup menu in the request tree.

5.3 AB Micro 800

This PLC series uses a symbolic addressing method. The program may read data from the "Global variables" area by a tag name. The logger should work in the TCP client mode and connect to a port #44818, directly to the PLC.

Name	Alias	Data Type	Dimension	Project Value	Initial Value	Comment	Retained
ALARM_MSG_9		ASCELOCADD					
ALARM_1_RESET		BOOL		FALSE			
ALARM_2_RESET		BOOL		FALSE			
ALARM_3_RESET		BOOL		FALSE			
ALARM_4_RESET		BOOL		FALSE			
ALARM_5_RESET		BOOL		FALSE			
ALARM_6_RESET		BOOL		FALSE			
ALARM_7_RESET		BOOL		FALSE			
ALARM_8_RESET		BOOL		FALSE			
ALARM_9_RESET		BOOL		FALSE			
ALARM_10_RESET		BOOL		FALSE			
ALARM_11_RESET		BOOL		FALSE			
HMI_NUMBER_1		STRING		'9737652320'	'9737652320'		2
HMI_NUMBER_1_SELECTED		BOOL		FALSE			
HMI_NUMBER_2		STRING		'8849093988'	'8849093988'		2
HMI_NUMBER_2_SELECTED		BOOL		FALSE			
HMI_NUMBER_3		STRING					2
HMI_NUMBER_3_SELECTED		BOOL		FALSE			
HMI_NUMBER_4		STRING					2
HMI_NUMBER_4_SELECTED		BOOL		FALSE			
HMI_NUMBER_5		STRING					2
HMI_NUMBER_5_SELECTED		BOOL		FALSE			
Tag1		INT			25		

Fig.1. Tag in a PLC

Configuration options

Device: Query Parse Filter

Log file: Data query module

Other: Ethernet/IP [AB Micro 800] (eipcip.dll) [Setup]

Modules: Parser module
Ethernet/IP [AB Micro 800] (eipcip.dll) [Setup]

Parsing and exporting for data sent
 Parsing and exporting for data received

Select data filter modules

Module name	Version
<input type="checkbox"/> Aggregation (aggregator.dll)	4.0.32.413
<input type="checkbox"/> Alarms Professional (alarm.dll)	4.0.1.413
<input type="checkbox"/> Data redirecting (dataredirect.dll)	4.0.31.413
<input type="checkbox"/> Data reformat (datareformat.dll)	4.0.16.413
<input type="checkbox"/> Data timeout (datatimeout.dll)	4.0.33.413
<input type="checkbox"/> Deadband (deadband.dll)	4.0.31.413
<input type="checkbox"/> Digital inputs filter (adamdio.dll)	4.0.32.413
<input type="checkbox"/> Events generator (eventgen.dll)	4.0.32.413
<input type="checkbox"/> Expressions (expressions.dll)	4.0.39.516

[Up] [Down] [Help] [Setup]

[OK] [Cancel]

Fig.2. Data parser

Property	Value
Request #1	
<input checked="" type="checkbox"/> Send requests, otherwise parse response only	
Device path	1
Address (e.g. Tag[0] or @22/1...	Tag1
Number of elements to read	1
Request timeout (ms)	3000
Request method	
<input type="radio"/> Once, on the program startup	
<input checked="" type="radio"/> Polling	
Interval (ms)	10
Interval units	Second
Export name	AAA
Default value	0
Scale (numbers only)	1

Action Export data for all requests at once

Minimal interval between data packets (ms)

OK Cancel

Fig.3. Queue

5.4 AB MicroLogix 1400

This PLC series does not support a symbolic addressing method. The program may read data from area by a file type and address. The following file types are supported:

- R - Control;
- C - Counter;
- F - Floating-point;
- I - Input;
- N - Integer;
- O - Output;
- T - Timer;

The data address should look like: N7:0

N - file data type ID.
7 - file number.

0 - element address

The logger should work in the TCP client mode and connect to a port #44818, directly to the PLC.

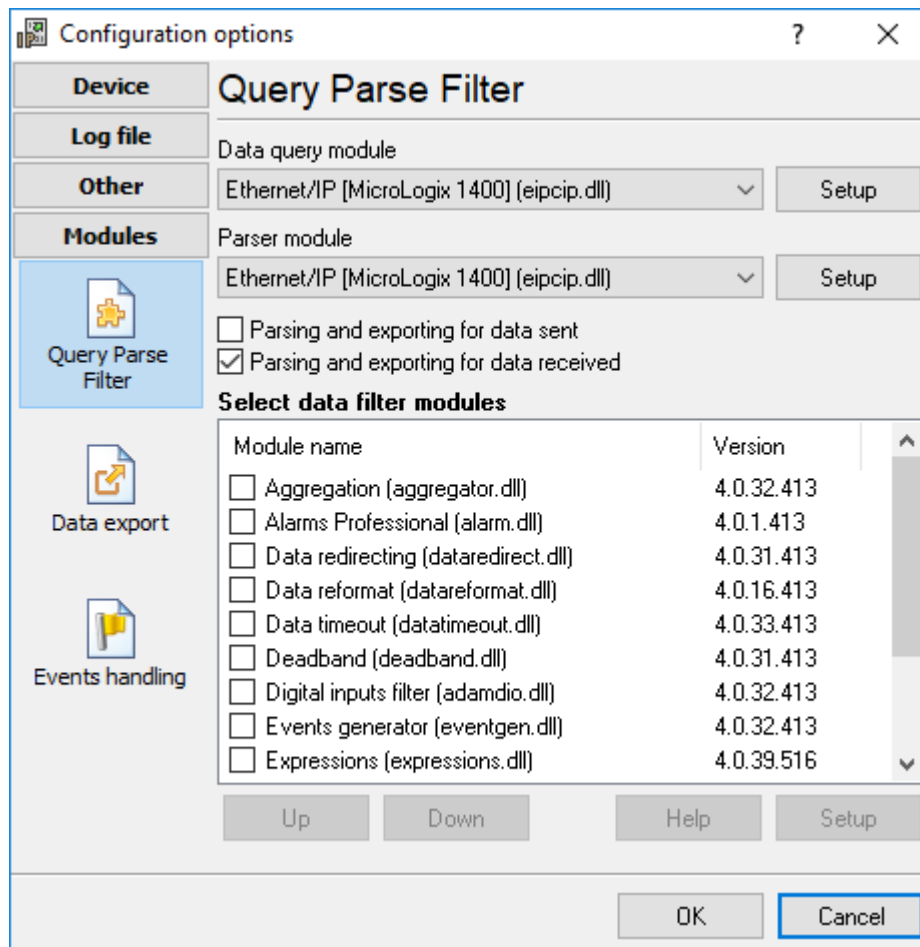


Fig.2. Data parser

Ethernet/IP Ethernet/IP [AB Micro 800] Ethernet/IP [Log...]

Requests queue

Property	Value
Request #1	
<input checked="" type="checkbox"/> Send requests, otherwise parse response only	
Device path	1
Address (e.g. Tag[0] or @22/1...	N7:0
Number of elements to read	1
Request timeout (ms)	3000
Request method	
<input type="radio"/> Once, on the program startup	
<input checked="" type="radio"/> Polling	
Interval (ms)	10
Interval units	Second
Export name	N7_0
Default value	0
Scale (numbers only)	1
Request #2	
<input checked="" type="checkbox"/> Send requests, otherwise parse response only	
Device path	1
Address (e.g. Tag[0] or @22/1...	F8:1
Number of elements to read	1
Request timeout (ms)	3000
Request method	
<input type="radio"/> Once, on the program startup	

Action Export data for all requests at once

Minimal interval between data packets (ms)

OK Cancel

Fig.3. Queue

6 Troubles?

6.1 Possible problems

No data for publication/exporting – no data is passed for exporting. Solution: configure the parser, make sure that one or more variables are declared in the parser.

Error on binding variable with name %s [%s] – the error usually occurs if data does not correspond to the specified format. For example, the date and time format does not correspond to the data.

Unable to disconnect from the database [%s] and **Unable to connect to a database [%s]** – it is impossible to connect/disconnect to/from the database. You should check the parameters of the database connection. The analysis of the additional information will help you locate the error.

Database access error [%s]. Stop operations with the database? – the message appears if an error occurs during an attempt to execute an SQL query if the second variant of reacting to errors is selected. The message implies a "Yes" or "No" answer. The analysis of the additional information will help you locate the error.

Unable to verify your SQL script [%s] – the message appears when an attempt to analyze your SQL query fails. Check if the syntax of your SQL query is correct.

Tested successfully – the message appears if your database connection is successfully tested. It requires no additional actions.

Database isn't used – the message appears if the module is temporarily disabled (the "Temporarily disabled" check box is selected) or the database name field is empty. Check the connection parameters.

Database isn't selected - the message appears if the database type is not selected. Check the connection parameters.

Database: %s – %s contains the database name. The message appears if the database connection is successful. Usually, you see it when you call the module for the first time. It requires no additional actions.

Invalid data block length (columns=%d,length=%d) – an internal application error. It means that the data sent by the parser is in an invalid format. Perhaps, you are using the module incompatible with the version of the Advanced Serial Data Logger kernel. Update the versions of both the kernel and the module.

The time of connection is not due yet (%d,%d) – the message appears during an attempt to connect to the database after the connection to it has been lost and the "Reconnect after" option is enabled. No additional actions are required.

Invalid procedure call. Bad arguments – an attempt to call the module using invalid parameters. Perhaps, you are using the module incompatible with the version of the Advanced Serial Data Logger kernel. Update the versions of both the kernel and the module.

Writing to the database is complete - the message appears if your queue of SQL queries is successfully executed. It requires no additional actions.

Writing to the database is complete with errors – the message appears if the executing your queue of SQL queries was interrupted by an error. It requires no additional actions.

Your SQL is empty. Please, specify some SQL text first – the message appears if you do not enter the text for your SQL query. Check if the options on the "SQL queue" tab are configured correctly.

Invalid temporary path – the path to the temporary file specified by you does not exist. Enter a new path in the "Temporary folder" field on the "Errors handling" tab.

%s, %d – will be replaced by additional information.