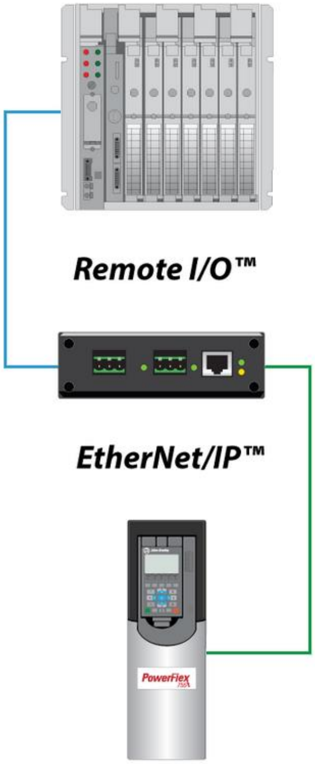
Technical Note

# AN-X2-AB-DHRIO Remote I/O to EIP Drives Configuration

**Document Code: TN171005-000**
Author: Andy Koenig
Date: January 18, 2017



## Asia Pacific

**Malaysia Office**
Phone: +603.7724.2080
asiapc@prosoft-technology.com
*Languages spoken: Chinese, English*

**China Office**
Phone: +86.21.5187.7337
asiapc@prosoft-technology.com
*Languages spoken: Chinese, English*

## Europe, Middle East, Africa

**France Office**
Phone: +33 (0)5.34.36.87.20
europe@prosoft-technology.com
*Languages spoken: French, English*

**Middle East and Africa**
Phone: +971.(0)4.214.6911
mea@prosoft-technology.com
*Languages spoken: English, Hindi*

## North America

**Corporate Headquarters**
Phone: +1 661.716.5100
support@prosoft-technology.com
*Languages spoken: English, Spanish*

## Latin America

**Brazil Office**
Phone: +55.11.5083.3776
brasil@prosoft-technology.com
*Languages spoken: Portuguese, English*

**Mexico and Central America Office**
Phone: +52.222.3.99.6565
soporte@prosoft-technology.com
*Languages spoken: Spanish, English*

**Regional Office**
Phone: +1.281.298.9109
latinam@prosoft-technology.com
*Languages spoken: Spanish, English*

ASIA PACIFIC | AFRICA | EUROPE | MIDDLE EAST | LATIN AMERICA | NORTH AMERICA

# 1 Purpose

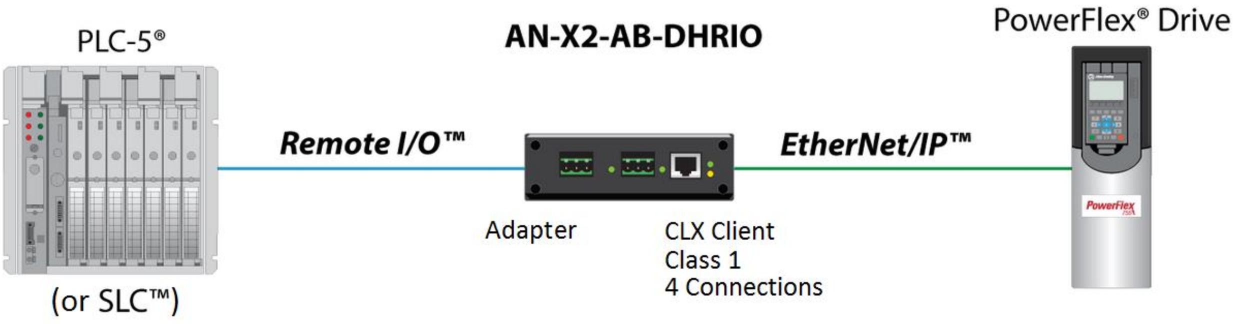
This document describes the process for configuring an AN-X2-AB-DHRIO gateway to connect Allen-Bradley Remote I/O to up to four EtherNet/IP enabled drives.

This document references Allen-Bradley drives, but other drives and some types of robots might also use the AN-X2-AB-DHRIO gateway. The gateway has several modes of operation, including the Drives mode described here. (Some robots may require the gateway in a different mode.)

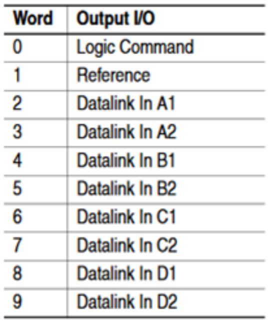
This procedure assumes knowledge of the PLC application, Remote I/O and configuration details of the drive.

For physical connections, further configuration details and troubleshooting, refer to the *AN-X2-ABDHRIO Remote I/O Drive Interface User Manual*.

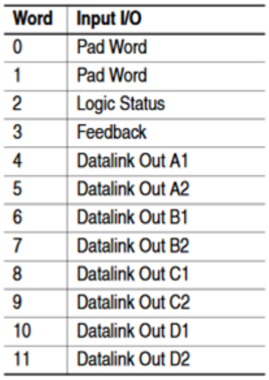
# 2 Background



The AN-X2-AB-DHRIO gateway supports upgrading legacy Allen-Bradley drives on Remote I/O to new EtherNet/IP enabled drives. The gateway maps Drive tags into a RIO adapter. Your drive documentation should provide a list of available tags.

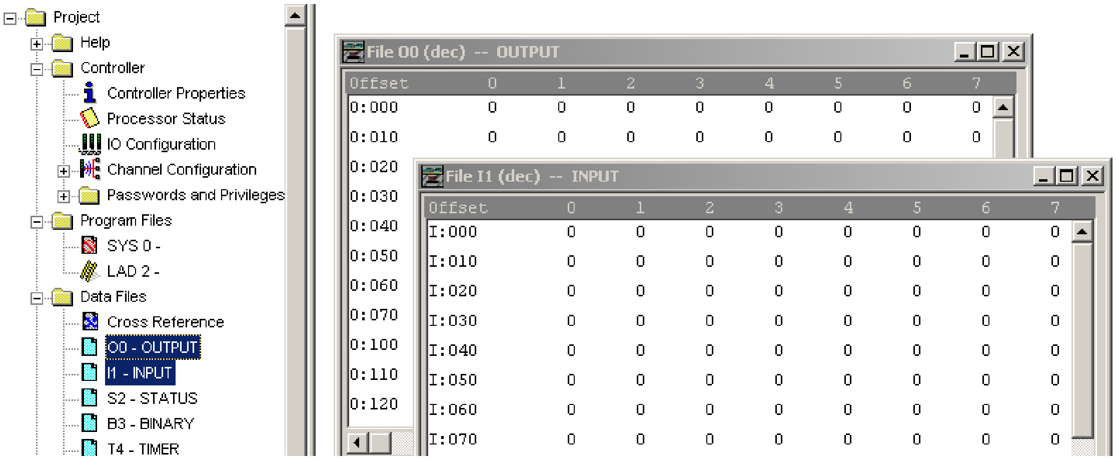


| Word | Output I/O |
|---|---|
| 0 | Logic Command |
| 1 | Reference |
| 2 | Datalink In A1 |
| 3 | Datalink In A2 |
| 4 | Datalink In B1 |
| 5 | Datalink In B2 |
| 6 | Datalink In C1 |
| 7 | Datalink In C2 |
| 8 | Datalink In D1 |
| 9 | Datalink In D2 |



| Word | Input I/O |
|---|---|
| 0 | Pad Word |
| 1 | Pad Word |
| 2 | Logic Status |
| 3 | Feedback |
| 4 | Datalink Out A1 |
| 5 | Datalink Out A2 |
| 6 | Datalink Out B1 |
| 7 | Datalink Out B2 |
| 8 | Datalink Out C1 |
| 9 | Datalink Out C2 |
| 10 | Datalink Out D1 |
| 11 | Datalink Out D2 |

The AN-X2 can map 8 words in and 8 words out per drive, up to 4 drives. The inputs map to the processor’s I1 data file. The outputs map to the processor’s O0 data file. Additional data can be accessed using block transfers.



- Project
  - Help
  - Controller
    - Controller Properties
    - Processor Status
    - IO Configuration
    - Channel Configuration
    - Passwords and Privileges
  - Program Files
    - SYS 0 -
    - LAD 2 -
  - Data Files
    - Cross Reference
    - O0 - OUTPUT
    - I1 - INPUT
    - S2 - STATUS
    - B3 - BINARY
    - T4 - TIMER

File O0 (dec) -- OUTPUT

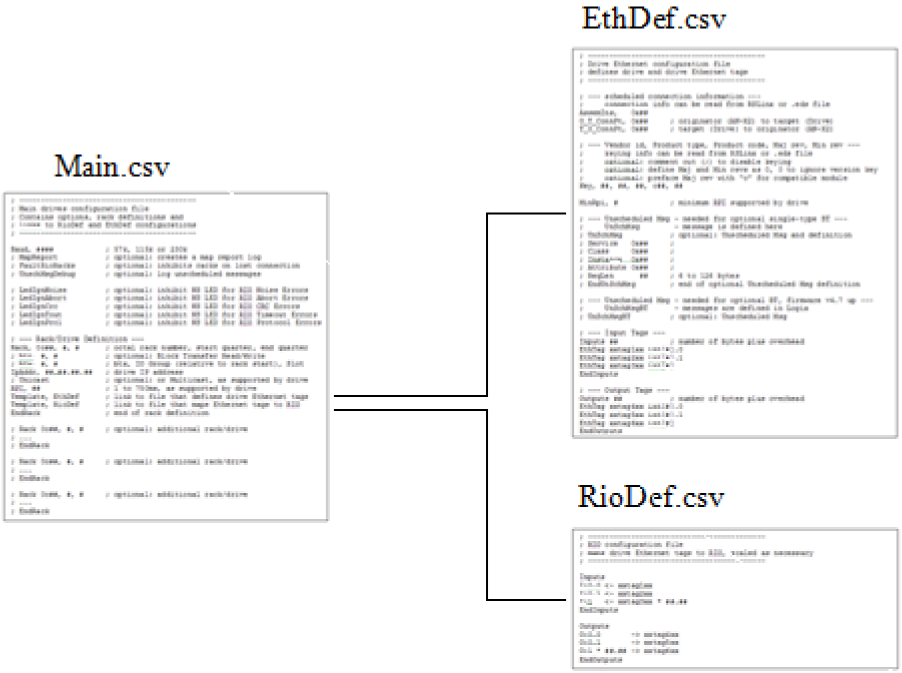
| Offset | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| O:000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| O:010 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| O:020 | | | | | | | | |
| O:030 | | | | | | | | |
| O:040 | | | | | | | | |
| O:050 | | | | | | | | |
| O:060 | | | | | | | | |
| O:070 | | | | | | | | |
| O:100 | | | | | | | | |
| O:110 | | | | | | | | |
| O:120 | | | | | | | | |

File I1 (dec) -- INPUT

| Offset | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| I:000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| I:010 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| I:020 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| I:030 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| I:040 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| I:050 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| I:060 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| I:070 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

The AN-X2 is configured with three text files.

- The Main file defines the adapter racks, baud and options, and identifies the EtherNet/IP and RIO configuration files for each drive.
- The EthDef file defines the Drive input and output tags, connection and messages. Preconfigured EthDef files are available for many Allen-Bradley drives.
- The RioDef file defines the tag mapping from EtherNet/IP to RIO, scaling tags as needed.

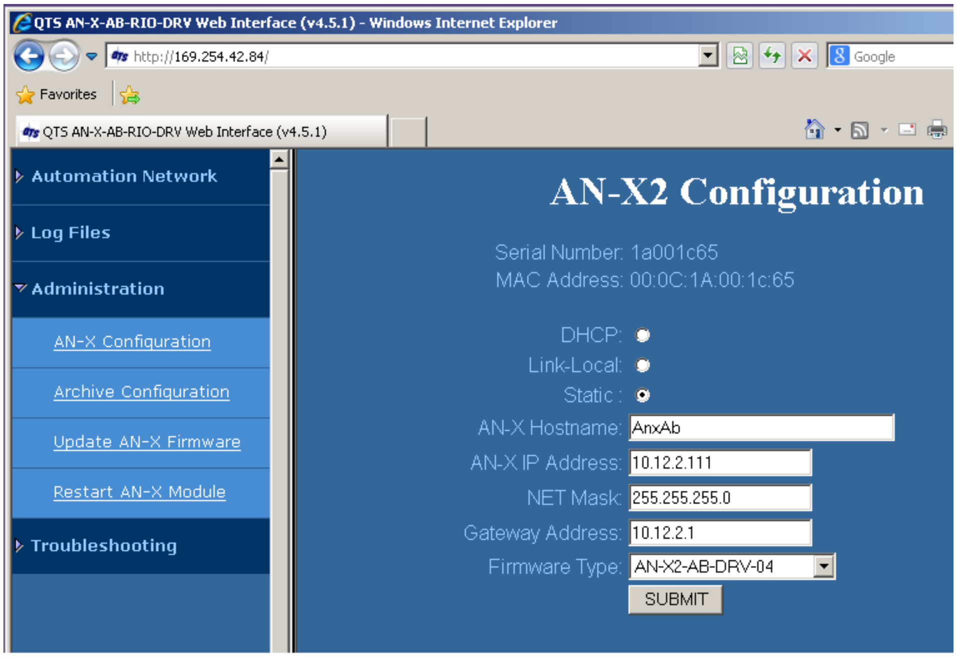


# 3 Configuration Process

## 3.1 Configure AN-X2 Mode and Ethernet

A web-browser is used for the initial configuration. An un-configured AN-X2 has a default IP address of 169.254.42.84.

If the AN-X2 has an unknown configuration, please consult the user manual “Reconfiguring an AN-X from an Unknown State”.



AN-X2 Configuration

Serial Number: 1a001c65
MAC Address: 00:0C:1A:00:1c:65

DHCP: ○
Link-Local: ○
Static : ●
AN-X Hostname: AnxAb
AN-X IP Address: 10.12.2.111
NET Mask: 255.255.255.0
Gateway Address: 10.12.2.1
Firmware Type: AN-X2-AB-DRV-04

### 3.1.1 Open AN-X2 Configuration

Open a web-browser and enter the current AN-X2 IP in the browser address.

### 3.1.2 Select Connection Type

Select the ***Static*** IP type in most cases. Your network admin can assign a unique IP address for the AN-X2.

If you select DHCP, a DHCP server assigns an IP address each time the AN-X2 powers up. Unless you have control of the DHCP server, that server can assign a different IP address each time. To find the address assigned, view the DHCP server. When submitting the changes, if a DHCP server is not found, the AN-X2 reverts to the default link-local address and repeatedly flashes the SYS (or MS) LED 3 reds followed by a pause.

If you select Link-local, the fixed link-local address, 169.254.42.84, is used. The AN-X2 must be on the same subnet as the computer to use the link-local IP address. It cannot be connected through a router. Link-local is typically used only for configuration.

### 3.1.3 Enter Hostname

Enter a Hostname for the AN-X2. This name is used internally by AN-X2 and may be used to identify the AN-X2 if you have a DNS server on your network. The name can be from 1 to 30 characters long.

### 3.1.4 Enter Static IP

If you select static IP address, enter the IP address, netmask and default gateway for your network. You must enter a valid default gateway address even if there is no device at the gateway address on the network.

***Important***: If you are connecting AN-X2 to an existing Ethernet network, consult the network administrator to obtain a static IP address.

### 3.1.5 Select Firmware Type DRV-04

From the dropdown list, select the Drive firmware AN-X2-AB-DRV04.

### 3.1.6 Submit Configuration

After editing parameters, click the SUBMIT button to write the configuration to the AN-X2. The changes will take effect once the AN-X restarts.

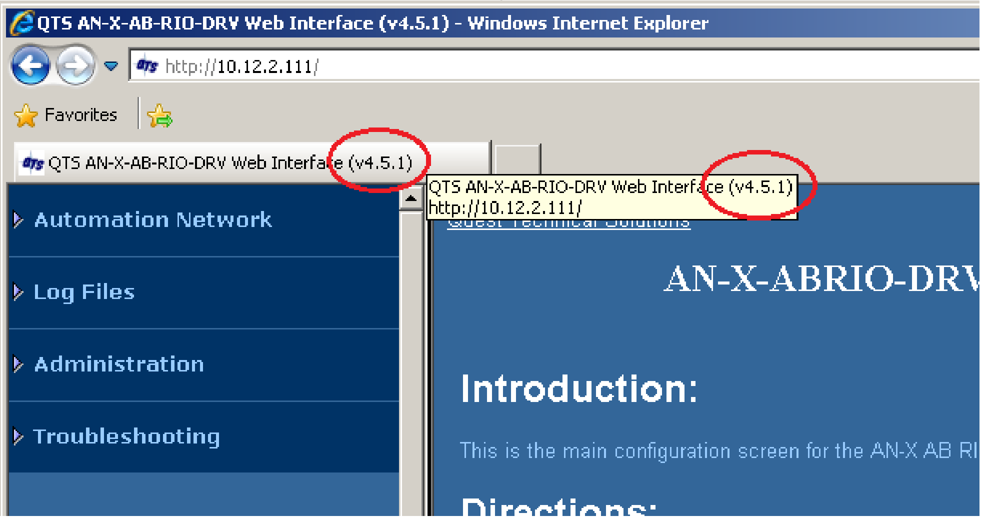
Click Continue to restart the AN-X2. Wait until the AN-X2 has completely restarted before continuing.

If you have changed the IP address, enter the new IP address in the browser's address field.

If you have changed the Firmware Type, flush the browser cache (delete browser history) to force the browser to reload the new AN-X2 page. (Ctrl-F5 will reload pages in Firefox or Internet Explorer, Shift-F5 will reload pages in Google Chrome.)

### 3.1.7 Note Firmware Version

Some features are dependent on the version. The version appears in the tab or, on some browsers, it appears when you hover over the tab.


QTS AN-X-AB-RIO-DRV Web Interface (v4.5.1) - Windows Internet Explorer
http://10.12.2.111/
Favorites
QTS AN-X-AB-RIO-DRV Web Interface (v4.5.1)
QTS AN-X-AB-RIO-DRV Web Interface (v4.5.1)
http://10.12.2.111/
Quest Technical Solutions
Automation Network
Log Files
Administration
Troubleshooting
AN-X-ABRIO-DRV
Introduction:
Directions:

This is the main configuration screen for the AN-X AB RI

## 3.2 Configure Main Definition File

The Main configuration file defines the Remote I/O logical rack(s) in the AN-X2, baud rate and additional options (described below). The rack definitions include references to the RIO and EtherNet/IP configuration files. A semi-colon “;” indicates a comment. Any text following a semi-colon is ignored by the AN-X2.

The configuration files are best edited in Notepad. Editing in Excel will introduce redundant (but harmless) commas throughout the file. The filename can be changed, but the extension must remain .csv.

Main.csv:

```
; -----------------------------------------
; Main drives configuration file
; Contains options, rack definitions and
; links to RioDef and EthDef configurations
; -----------------------------------------

Baud, ####            ; 57k, 115k or 230k
; MapReport           ; optional: creates a map report log
; FaultRioRacks       ; optional: inhibits racks on lost connection
; UnschMsgDebug       ; optional: log unscheduled messages

; LedIgnNoise         ; optional: inhibit NS LED for RIO Noise Errors
; LedIgnAbort         ; optional: inhibit NS LED for RIO Abort Errors
; LedIgnCrc           ; optional: inhibit NS LED for RIO CRC Errors
; LedIgnTout          ; optional: inhibit NS LED for RIO Timeout Errors
; LedIgnPrcl          ; optional: inhibit NS LED for RIO Protocol Errors

; --- Rack/Drive Definition ---
Rack, 0o##, #, #      ; octal rack number, start quarter, end quarter
; btr, #, #           ; optional: Block Transfer Read/Write
; btw, #, #           ; btx, IO Group (relative to rack start), Slot
IpAddr, ##.##.##.##   ; drive IP address
; Unicast             ; optional: as supported by drive
RPI, ##               ; 1 to 750ms, as supported by drive
Template, EthDef      ; link to file that defines drive Ethernet tags
Template, RioDef      ; link to file that maps Ethernet tags to RIO
EndRack               ; end of rack definition

; Rack 0o##, #, #     ; optional: additional rack/drive
; ...
; EndRack

; Rack 0o##, #, #     ; optional: additional rack/drive
; ...
; EndRack

; Rack 0o##, #, #     ; optional: additional rack/drive
; ...
; EndRack
```

### 3.2.1 Select `Baud` Rate

Set the RIO baud rate of 57k, 115k or 230k. A comma separates the keyword `Baud` and rate value.

```
Baud, 57k
```

### 3.2.2 Select Optional `MapReport`

Include the `MapReport` option (remove the preceding semi-colon) to generate a data map report.

```
MapReport
```

This report can be used for documentation, debugging and technical support. The resulting *MapReport.log* is available in the AN-X2 web interface under *View All Logs*.

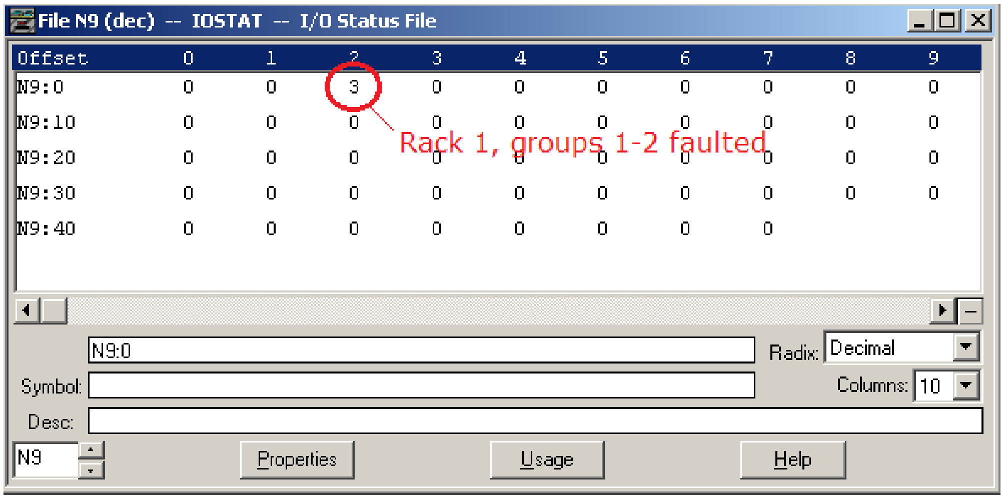
### 3.2.3 Select Optional `FaultRioRacks`

Include the `FaultRioRacks` option (remove the preceding semi-colon) to allow the PLC to indicate EIP network connection problems.

```
FaultRioRacks
```

**TIP**: Leave `FaultRioRacks` commented until communication between the rack and the RIO scanner is initially established. Then enable the option as desired.

`FaultRioRacks` causes the AN-X2 to initially inhibit its emulated remote I/O racks. When the AN-X gets a good connection with the Ethernet drive, inhibit is removed from the associated rack. If the AN-X2 loses the EIP connection, the rack is inhibited again. This allows the PLC program to monitor the remote rack status and determine if the drive is communicating on the EIP network. For example, a PLC5 IO-status table is showing an EIP disconnect:



File N9 (dec) -- IOSTAT -- I/O Status File

| Offset | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| N9:0 | 0 | 0 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| N9:10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| N9:20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| N9:30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| N9:40 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |

Rack 1, groups 1-2 faulted

N9:0
Radix: Decimal
Symbol:
Columns: 10
Desc:
N9

### 3.2.4 Select Optional `UnschMsgDebug`

Available in v4.7 and later firmware (section 3.1.7).
Include the `UnschMsgDebug` option (remove the preceding semi-colon) to write the contents of unscheduled messages to the Ethernet/IP log.

```
UnschMsgDebug
```

`UnschMsgDebug` can help with debugging unscheduled messages. The contents of the unscheduled message Requests and Responses are written to the Ethernet/IP log, showing exactly what the AN-X is sending and what the drive is returning.

### 3.2.5 Select Optional `LedIgn`*`Xxx`* Inhibits

Available in v4.7 and later firmware (section 3.1.7).
The AN-X2 module may see errors on the RIO network that do not affect the operation of the AN-X2, but that cause the NS LED to blink or stay red. (For example, the AN-X2 might see a corrupt message whose recipient cannot be identified. This might be a message to the AN-X2 and a legitimate error, or it might be a nuisance error.)

To stop the LED flashing nuisance errors, include the appropriate options.

```
LedIgnNoise
LedIgnAbort
LedIgnCrc
LedIgnTout
LedIgnPrcl
```

Generally, only `LedIgnNoise`, `LedIgnAbort` and `LedIgnCrc` will be needed. Use discretion when disabling any NS LED error indications.

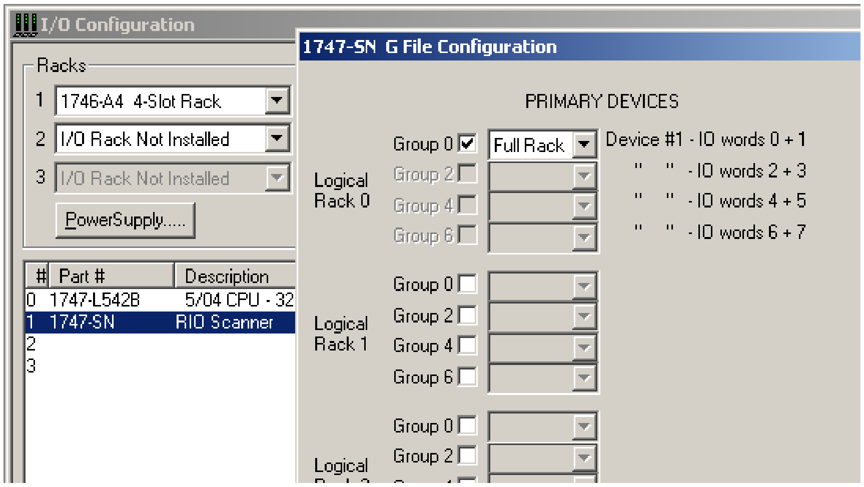
Diagnostic counters will continue to increment for all errors. This option only suppresses the LED error indication.

### 3.2.6 Define Remote I/O `Rack` in the AN-X2

Drive definitions begin with a line that contains the keyword `Rack`, and end with a line that consists of just the keyword `EndRack`.

Define the rack number and rack size to match the PLC. Rack numbers are octal.

```
Rack, 0o00, 1, 4 ; rack number 0, full rack
```



I/O Configuration

Racks
1 1746-A4 4-Slot Rack
2 I/O Rack Not Installed
3 I/O Rack Not Installed
PowerSupply.....

| # | Part # | Description |
|---|---|---|
| 0 | 1747-L542B | 5/04 CPU - 32 |
| 1 | 1747-SN | RIO Scanner |
| 2 | | |
| 3 | | |

1747-SN G File Configuration

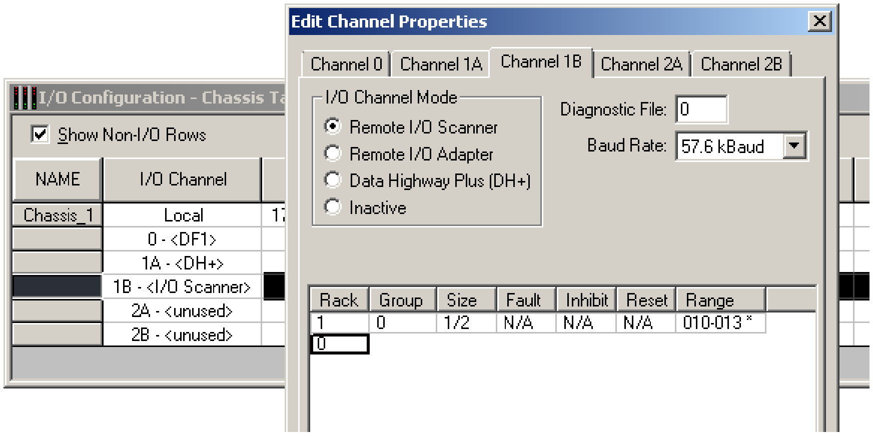
PRIMARY DEVICES

Logical Rack 0
Group 0 ☑ Full Rack — Device #1 - IO words 0 + 1
Group 2 □ — " " - IO words 2 + 3
Group 4 □ — " " - IO words 4 + 5
Group 6 □ — " " - IO words 6 + 7

Logical Rack 1
Group 0 □
Group 2 □
Group 4 □
Group 6 □

Logical
Group 0 □
Group 2 □

```
Rack, 0o01, 1, 2 ; rack number 1, first half rack
```



Edit Channel Properties

Channel 0 | Channel 1A | Channel 1B | Channel 2A | Channel 2B

I/O Channel Mode
● Remote I/O Scanner
○ Remote I/O Adapter
○ Data Highway Plus (DH+)
○ Inactive

Diagnostic File: 0
Baud Rate: 57.6 kBaud

| Rack | Group | Size | Fault | Inhibit | Reset | Range |
|---|---|---|---|---|---|---|
| 1 | 0 | 1/2 | N/A | N/A | N/A | 010-013 * |
| 0 | | | | | | |

I/O Configuration - Chassis Ta

☑ Show Non-I/O Rows

| NAME | I/O Channel |
|---|---|
| Chassis_1 | Local |
| | 0 - <DF1> |
| | 1A - <DH+> |
| | 1B - <I/O Scanner> |
| | 2A - <unused> |
| | 2B - <unused> |

The definition consists of the keyword `Rack`, the rack number in octal `0oxx` (xx is 0-76), the start quarter (1-4) and the end quarter (1-4). Commas separate the keyword and each element. The end quarter must be greater than or equal to the start quarter.

### 3.2.7 Define Optional `btw/btr` Block Transfers

Optional unscheduled data transfers between the PLC and AN-X2 are performed using Block Transfer Read and Write instructions.

```
btw, 7, 0
btr, 7, 1
```

Block transfer definitions consist of the block transfer keyword (read or write), the *relative* I/O group (0 to 1 for quarter rack, 0 to 3 for half rack, or 0 to 7 for full rack) and the slot (0 or 1). Commas separate the keyword and elements.

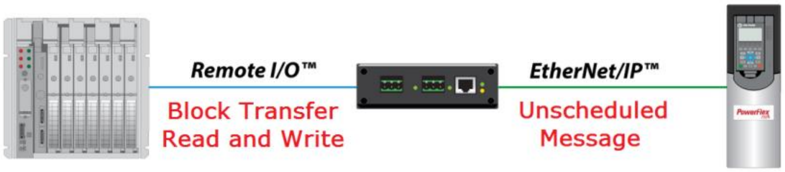
***Important***: The location of the block transfer must not overlap discrete remote I/O output data.

The I/O group is relative to the start of the rack. If a rack is defined half rack starting at the third quarter, an I/O group number 3 in the block transfer definition means actual I/O group 7 in the rack.

```
Rack, 0o01, 3, 4 ; rack number 1, second half
btw, 3, 0        ; actual group 7
```

| Quarter | 1 | | 2 | | **3** | | **4** | |
|---|---|---|---|---|---|---|---|---|
| Relative I/O group | | | | | 0 | 1 | 2 | **3** |
| Actual I/O group | 0 | 1 | 2 | 3 | 4 | 5 | 6 | **7** |

Unscheduled data transfers between the AN-X2 and Drive are performed using unscheduled messages. Message definition is discussed in EtherNet/IP Definition File, sections 3.3.4 and 3.3.5. Examples are included in those sections.



### 3.2.8 Define Drive `IpAddr`

Specify the Drive's IP Address using the keyword `IpAddr`. A comma separates the keyword and address.

```
IpAddr, 169.254.42.84 ; but your IP address
```

### 3.2.9 Select Optional Unicast

Include the Unicast option to make a unicast connection to the Ethernet drive.

```
Unicast
```

The default is otherwise multicast. Unicast is preferred, but not all drives support unicast.

### 3.2.10 Define Drive RPI

Specify the drive RPI in milliseconds using the keyword RPI. A comma separates the keyword and RPI value. The range is from 1 (or the MinRpi supported by the drive, whichever is larger) to 750ms.

```
RPI, 50
```

### 3.2.11 Identify EthDef and RioDef Configuration File Templates

Specify the filenames containing the drive Ethernet tags and the Ethernet-to-RIO tag mapping. Use the keyword Template. A comma separates the keyword and filename.

```
Template, EthDef       ; drive Ethernet tags
Template, RioDef       ; map Ethernet tags to RIO
```

Do not include the .csv file extension. The filenames can be changed.

EthDef is discussed in section 3.3. RioDef is discussed in section 3.4.

### 3.2.12 Identify EndRack

EndRack denotes the end of the rack definition.

```
EndRack
```

### 3.2.13 Define Additional Racks

Repeat the rack definition for up to three additional racks/drives.

The rack locations must not overlap. The following is acceptable:

```
Rack, 0o01, 1, 2  ; rack 1, first half
Rack, 0o01, 3, 4  ; rack 1, second half
Rack, 0o02, 1, 2  ; rack 2, first half
Rack, 0o02, 3, 4  ; rack 2, second half
```

The same EthDef and RioDef templates can be used for each rack.

btr/btw are rack-relative, so can be identical for each rack.

### 3.3 Configure EtherNet/IP Definition File

The EthDef configuration file defines drive connection and keying parameters, the drive input and output tags, and optional unscheduled messaging. A semi-colon “;” indicates a comment. Any text following a semi-colon is ignored by the AN-X2.

The configuration files are best edited in Notepad. Editing in Excel will introduce redundant (but harmless) commas throughout the file. The filename can be changed, but must match the name used in main.csv. The file extension must remain .csv.

EthDef.csv:

```
; -----------------------------------------
; Drive Ethernet configuration file
; defines drive and drive Ethernet tags
; -----------------------------------------

; --- scheduled connection information ---
;     connection info can be read from .eds file
AssemIns,   0x##
O_T_ConnPt, 0x##     ; originator (AN-X2) to target (Drive)
T_O_ConnPt, 0x##     ; target (Drive) to originator (AN-X2)

; --- Vendor id, Product type, Product code, Maj rev, Min rev ---
;     keying info can be read from RSLinx or .eds file
;     Optional: comment out (;) to disable keying
;     optional: preface Maj rev with “c” for compatible module
Key, ##, ##, ##, c##, ##

MinRpi, #            ; minimum RPI supported by drive

; --- Unscheduled Msg – needed for optional single-type BT ---
;     UnSchMsg        – message is defined here
; UnSchMsg           ; optional: Unscheduled Msg and definition
; Service   0x##     ;
; Class     0x##     ;
; Instance  0x##     ;
; Attribute 0x##     ;
; ReqLen      ##     ; 3 to 126 bytes
; EndUnSchMsg        ; end of optional Unscheduled Msg definition

; --- Unscheduled Msg – needed for optional BT, firmware v4.7 up ---
;     UnSchMsgBT      – messages are defined in Logix
; UnSchMsgBT         ; optional: Unscheduled Msg

; --- Input Tags ---
Inputs ##            ; number of bytes plus overhead
EthTag DriveStatus_tag00 int[#].0
EthTag DriveStatus_tag01 int[#].1
EthTag Feedback          int[#]
EndInputs

; --- Output Tags ---
Outputs ##           ; number of bytes plus overhead
EthTag LogicCommand_tag00 int[#].0
EthTag LogicCommand_tag01 int[#].1
EthTag Reference          int[#]
EndOutputs
```

***Important***: Preconfigured EtherNet/IP Definition Files are available for various drives. These are included in the Drive Files download on the AN-X2-AB-DHRIO web page. This section describes how to create a new EtherNet/IP Definition file if one is not already available.

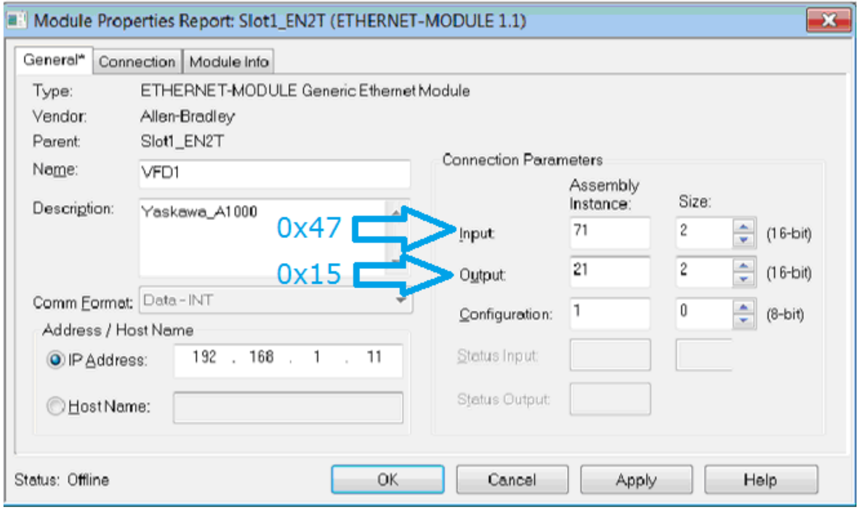
**A Note about Wireshark:** Wireshark software is a network protocol analyzer available at wireshark.org. Wireshark can be an invaluable aid in determining your EIP configuration. Wireshark is described in the appendix.

### 3.3.1 Define Scheduled Connection `AssemIns`, `O_T_ConnPt`, `T_O_ConnPt`.

Use the keywords `AssemIns`, `O_T_ConnPt` (output) and `T_O_ConnPt` (input) as shown. A comma separates the keywords and values.

```
AssemIns,   0x04
O_T_ConnPt, 0x15 ; Originator (AN-X) to Target
T_O_ConnPt, 0x47 ; Target (drive) to Originator
```

Specify the scheduled connection as you would for any module, like:



Module Properties Report: Slot1_EN2T (ETHERNET-MODULE 1.1)

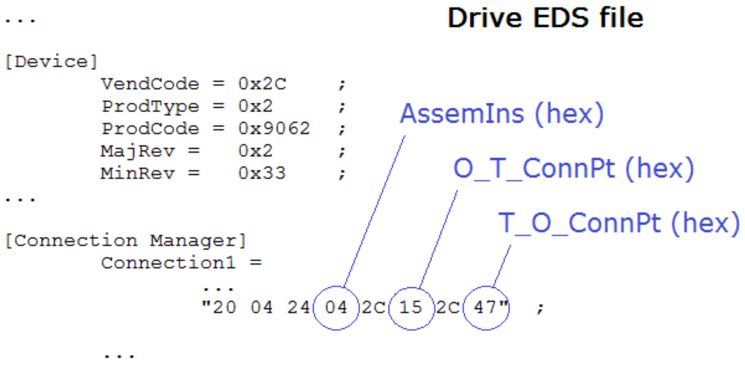
Type: ETHERNET-MODULE Generic Ethernet Module
Vendor: Allen-Bradley
Parent: Slot1_EN2T
Name: VFD1
Description: Yaskawa_A1000
Comm Format: Data - INT
Address / Host Name
● IP Address: 192 . 168 . 1 . 11
○ Host Name:

Connection Parameters

| | Assembly Instance: | Size: | |
|---|---|---|---|
| Input: | 71 | 2 | (16-bit) |
| Output: | 21 | 2 | (16-bit) |
| Configuration: | 1 | 0 | (8-bit) |
| Status Input: | | | |
| Status Output: | | | |

Status: Offline

Connection details can be found in the drive's EDS file or user manual. If multiple connections are available, pick the connection that provides the specific data required, like *Basic Speed Control* or *Extended Speed Control*.



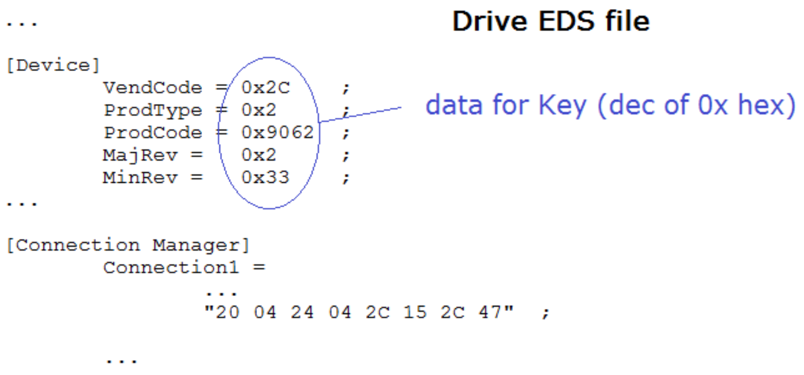
```
...
[Device]
        VendCode = 0x2C    ;
        ProdType = 0x2     ;
        ProdCode = 0x9062  ;
        MajRev =   0x2     ;
        MinRev =   0x33    ;
...
[Connection Manager]
        Connection1 =
                ...
                "20 04 24 04 2C 15 2C 47"  ;

        ...
```

### 3.3.2 Define Drive `Key`

Specify drive keying details using the keyword `Key`. Commas separate the keyword and values.

```
;     VendCode, ProdType, ProdCode, MajRev, MinRev
Key,    0x2C,     0x2,      0x9062,   c2,      33
```

A "`c`" before MajRev indicates a compatible version module on some drives.
Version "`..., 0, 0`" indicates no version keying on some drives.
Though not recommended, the `Key` line can be commented out ";" to indicate no keying. (This can help in troubleshooting.)
Keying details can be found in the drive's EDS file.

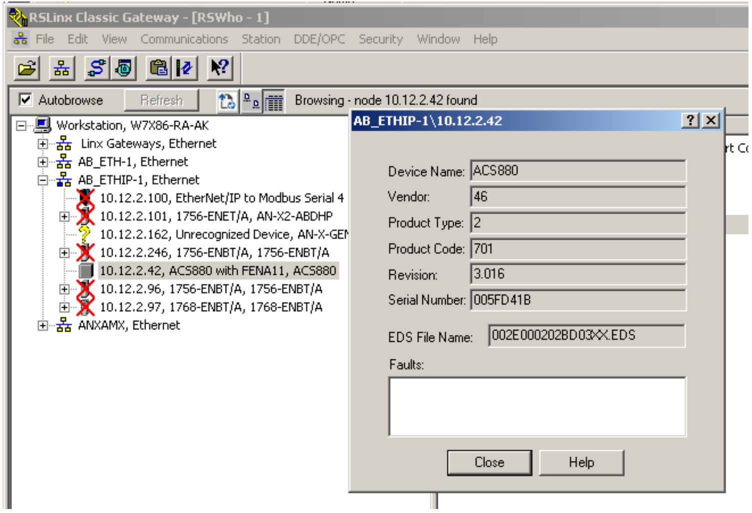


Drive EDS file

```
...
[Device]
        VendCode = 0x2C    ;
        ProdType = 0x2     ;
        ProdCode = 0x9062  ;
        MajRev  =  0x2     ;
        MinRev  =  0x33    ;
...

[Connection Manager]
        Connection1 =
                ...
                "20 04 24 04 2C 15 2C 47"  ;

        ...
```

data for Key (dec of 0x hex)

Keying information can also be seen in RSLinx. Right-click a module and select *Device Properties*. This drive is `Key, 46, 2, 701, 3, 16`.



RSLinx Classic Gateway - [RSWho - 1]

☑ Autobrowse   Browsing - node 10.12.2.42 found

- Workstation, W7X86-RA-AK
  - Linx Gateways, Ethernet
  - AB_ETH-1, Ethernet
  - AB_ETHIP-1, Ethernet
    - 10.12.2.100, EtherNet/IP to Modbus Serial 4
    - 10.12.2.101, 1756-ENET/A, AN-X2-ABDHP
    - 10.12.2.162, Unrecognized Device, AN-X-GEN
    - 10.12.2.246, 1756-ENBT/A, 1756-ENBT/A
    - 10.12.2.42, ACS880 with FENA11, ACS880
    - 10.12.2.96, 1756-ENBT/A, 1756-ENBT/A
    - 10.12.2.97, 1768-ENBT/A, 1768-ENBT/A
  - ANXAMX, Ethernet

AB_ETHIP-1\10.12.2.42

Device Name: ACS880
Vendor: 46
Product Type: 2
Product Code: 701
Revision: 3.016
Serial Number: 005FD41B
EDS File Name: 002E000202BD03XX.EDS
Faults:

### 3.3.3 Define the Drive `MinRpi`

Specify the Drive's minimum RPI using the keyword `MinRpi`. A comma separates the keyword and value.

```
MinRpi, 1
```

The RPI specified in the Main Definition File cannot be less than the minimum supported by the drive. Consult the drive manual for RPI.

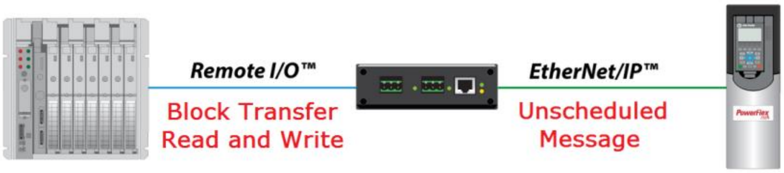
### 3.3.4 Define Optional Unscheduled Messaging UnSchMsg

To specify optional unscheduled messaging, edit the message format and remove the semi-colon preceding the lines.

```
UnSchMsg
Service    0x4B
Class      0x93
Instance   0x00
Attribute  0x00
ReqLen       30      ; 3 to 126 bytes
EndUnSchMsg
```

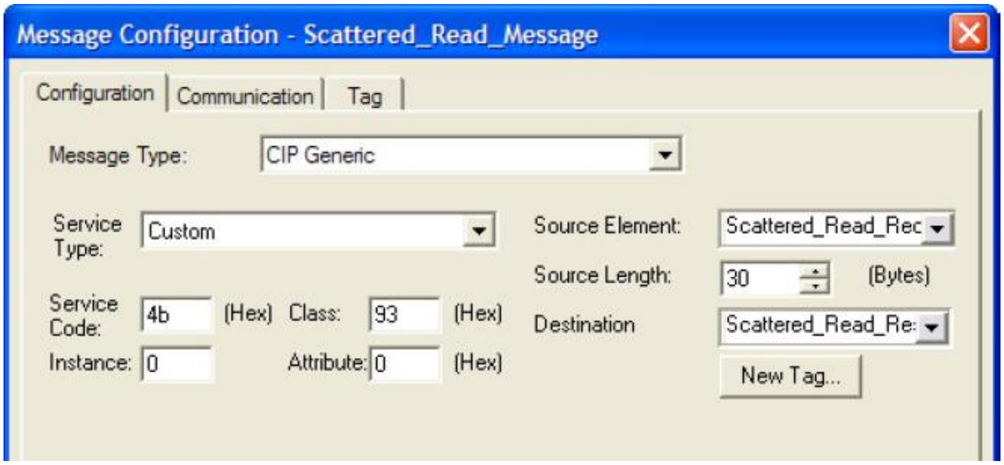
UnSchMsg allows a single type of unscheduled message to transfer data between the AN-X2 and drive.

To use unscheduled messaging, btr/btw must be included in the Main Definition File (section 3.2.7).



Starting with AN-X2 firmware version 4.7 (section 3.1.7), UnSchMsgBT is available for multiple message types. Do not use UnSchMsg if you are using UnSchMsgBT (section 3.3.5).

Consult your drive manual for message formats. Define Service, Class, Instance, and Attribute as you would any RSLogix5000 CIP Generic message. *Scattered Read Message* for a drive with a 20-COMM-E is shown here.


Message Configuration - Scattered_Read_Message

Message Type: CIP Generic

Service Type: Custom

Service Code: 4b (Hex) Class: 93 (Hex)

Instance: 0 Attribute: 0 (Hex)

Source Element: Scattered_Read_Rec

Source Length: 30 (Bytes)

Destination Scattered_Read_Re:

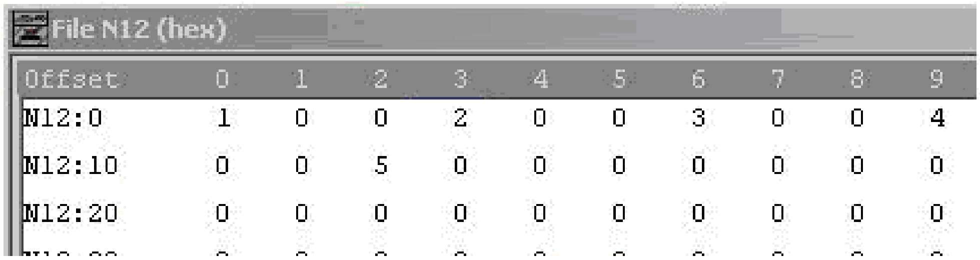
Define `ReqLen` (requested length). For this example, transferring a single drive parameter requires 6 bytes. To transfer 5 parameters requires 30 bytes.

The maximum number of parameters you can transfer in a single block transfer is 21, or 126 bytes.

For the *Scattered Read Message*, the PLC must execute a block transfer write containing a list of parameters to be read from the Ethernet drive.

For example, to read parameters 1, 2, 3, 4, and 5 from a drive, the block transfer write has length 15 (30 bytes) and consists of 5 blocks of 3 words. Each block consists of the parameter number and two placeholders.

**1** 0 0 **2** 0 0 **3** 0 0 **4** 0 0 **5** 0 0

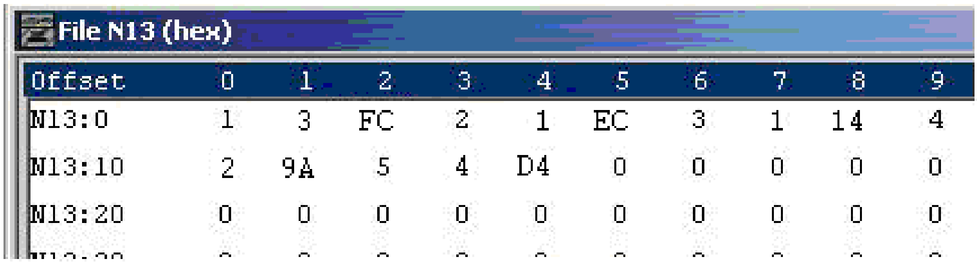


| Offset | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| N12:0 | 1 | 0 | 0 | 2 | 0 | 0 | 3 | 0 | 0 | 4 |
| N12:10 | 0 | 0 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| N12:20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

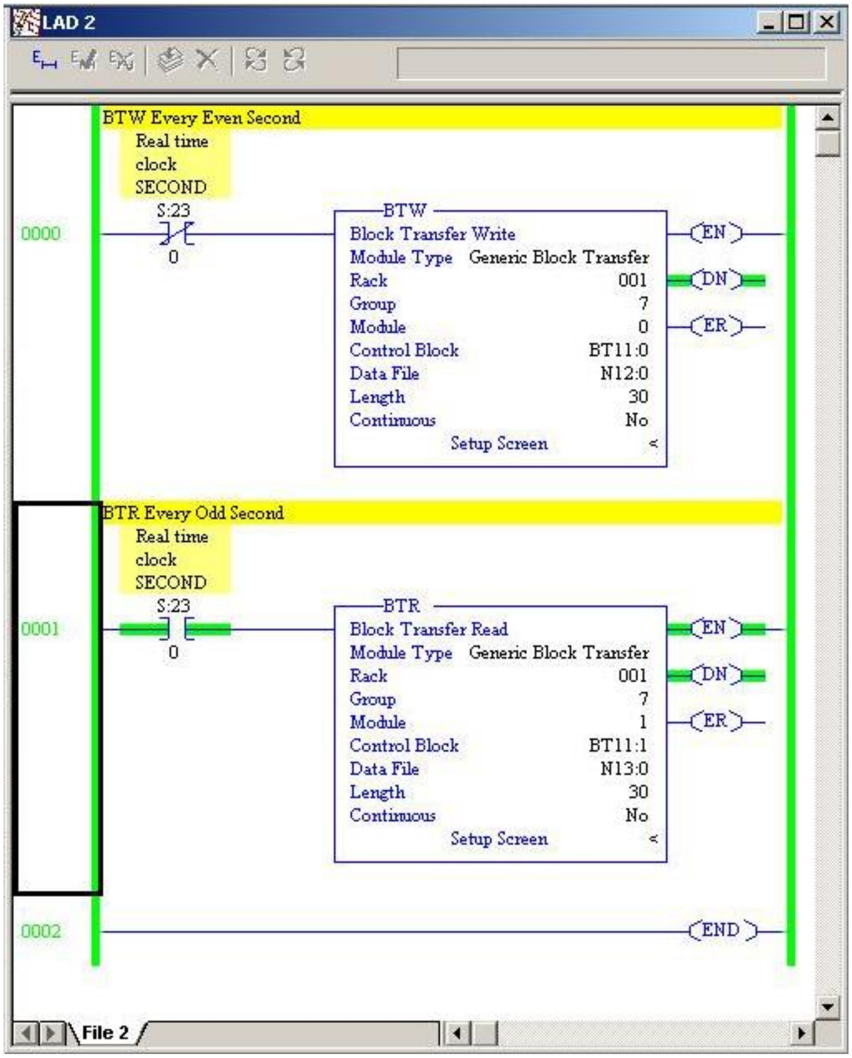
The AN-X sends a Scattered-Read message to the Ethernet drive including the list of parameters. When the AN-X receives a reply, it stores the data in the block transfer read data area.

The PLC must then issue a block transfer read request to retrieve the data. The length is again 15 (30 bytes) and consists of 5 blocks of three words. Each block contains the parameter number, the parameter's least significant word and the parameter's most significant word.

**1** LSW MSW **2** LSW MSW **3** LSW MSW **4** LSW MSW **5** LSW MSW



| Offset | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| N13:0 | 1 | 3 | FC | 2 | 1 | EC | 3 | 1 | 14 | 4 |
| N13:10 | 2 | 9A | 5 | 4 | D4 | 0 | 0 | 0 | 0 | 0 |
| N13:20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |



```
BTW Every Even Second
   Real time
   clock
   SECOND
    S:23                  BTW
0000 --]/[--------------- Block Transfer Write                      -(EN)-
      0                   Module Type      Generic Block Transfer   -(DN)-
                          Rack                                001   -(ER)-
                          Group                                 7
                          Module                                0
                          Control Block                     BT11:0
                          Data File                         N12:0
                          Length                               30
                          Continuous                           No
                                       Setup Screen             <

BTR Every Odd Second
   Real time
   clock
   SECOND
    S:23                  BTR
0001 --] [--------------- Block Transfer Read                       -(EN)-
      0                   Module Type      Generic Block Transfer   -(DN)-
                          Rack                                001   -(ER)-
                          Group                                 7
                          Module                                1
                          Control Block                     BT11:1
                          Data File                         N13:0
                          Length                               30
                          Continuous                           No
                                       Setup Screen             <

0002 ------------------------------------------------------------- -(END)-
```

***Important***: Again, consult your drive manual for message formats. For example, this same *Scattered Read Message* would be different for a drive with a 20-750-ENETR, with `Service` **4D** and **8** bytes per parameter.

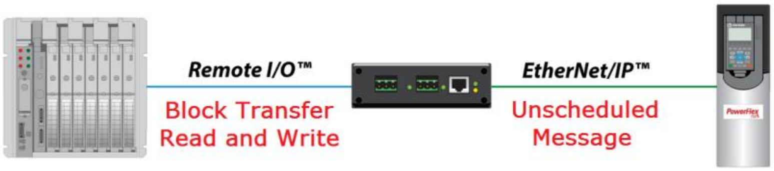
### 3.3.5 Define Optional Unscheduled Messaging `UnSchMsgBT`

To specify optional unscheduled messaging, remove the semi-colon preceding the line:

**`UnSchMsgBT`**

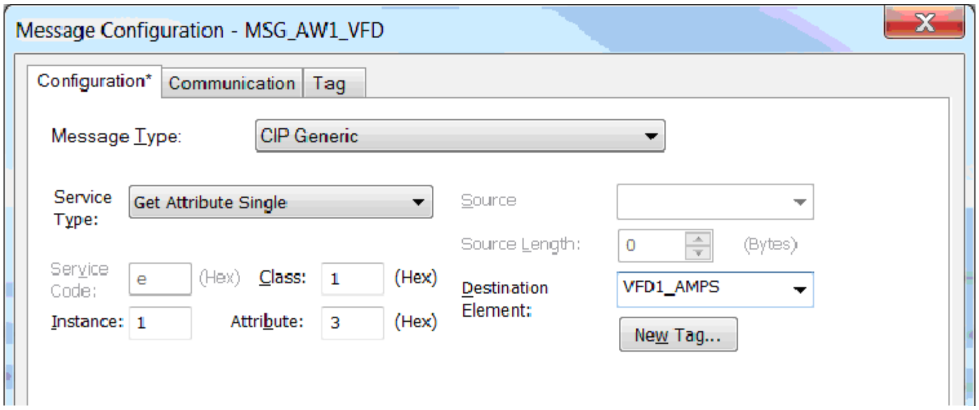
`UnSchMsgBT` allows multiple types of unscheduled messages to transfer data between the AN-X2 and drive.

To use unscheduled messaging, `btr/btw` must be included in the Main Definition File (section 3.2.7).



`UnSchMsgBT` is available in AN-X2 modules with firmware version 4.7 or later. For earlier version firmware, use `UnSchMsg` (section 3.3.4). Do not use `UnSchMsgBT` if you are using `UnSchMsg` or firmware prior to 4.7.

Consult your drive manual for message formats. Define your message as you would any RSLogix5000 CIP Generic message. *Get Attribute Single, Identity Object* for a drive with a 20-COMM-E is shown here.


Message Configuration - MSG_AW1_VFD

Configuration* | Communication | Tag

Message Type: CIP Generic

Service Type: Get Attribute Single

Source:

Source Length: 0 (Bytes)

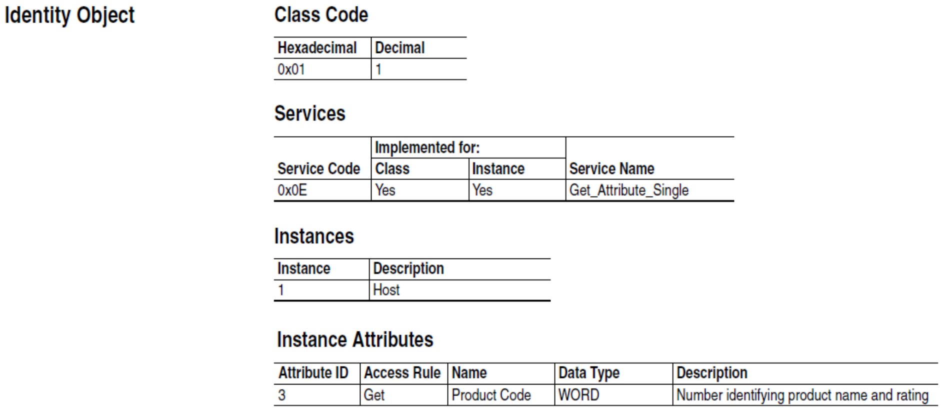
Service Code: e (Hex)

Class: 1 (Hex)

Instance: 1

Attribute: 3 (Hex)

Destination Element: VFD1_AMPS


Identity Object

Class Code

| Hexadecimal | Decimal |
|---|---|
| 0x01 | 1 |

Services

| | Implemented for: | | |
|---|---|---|---|
| Service Code | Class | Instance | Service Name |
| 0x0E | Yes | Yes | Get_Attribute_Single |

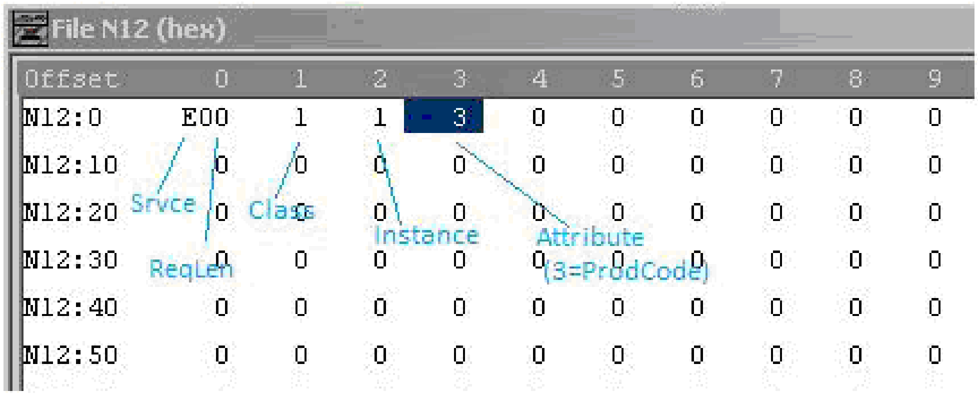
Instances

| Instance | Description |
|---|---|
| 1 | Host |

Instance Attributes

| Attribute ID | Access Rule | Name | Data Type | Description |
|---|---|---|---|---|
| 3 | Get | Product Code | WORD | Number identifying product name and rating |

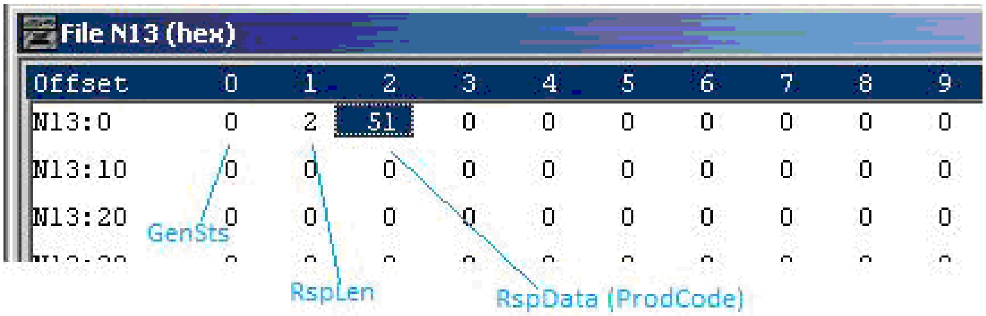
Unlike `UnSchMsg`, which is defined fully in EthDef.csv, `UnSchMsgBT` is defined in the PLC. The Service, Class, Instance, Attribute and Request Length are specified in the message request data at Offsets 0 through 3. The arguments are mapped as follows:



File N12 (hex)

| Offset | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| N12:0 | E00 | 1 | 1 | 3 | 0 | 0 | 0 | 0 | 0 | 0 |
| N12:10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| N12:20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| N12:30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| N12:40 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| N12:50 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

For this example, N12 is used for a CIP message request.
N12:0 Low Byte ...Write Data Length in bytes (0 in this example)
N12:0 Hi Byte ......CIP Service Code (0E in this example)
N12:1 ....................CIP Class
N12:2 ....................CIP Instance
N12:3 ....................CIP Attribute
N12:4 - n................Write Data (none in this example)



File N13 (hex)

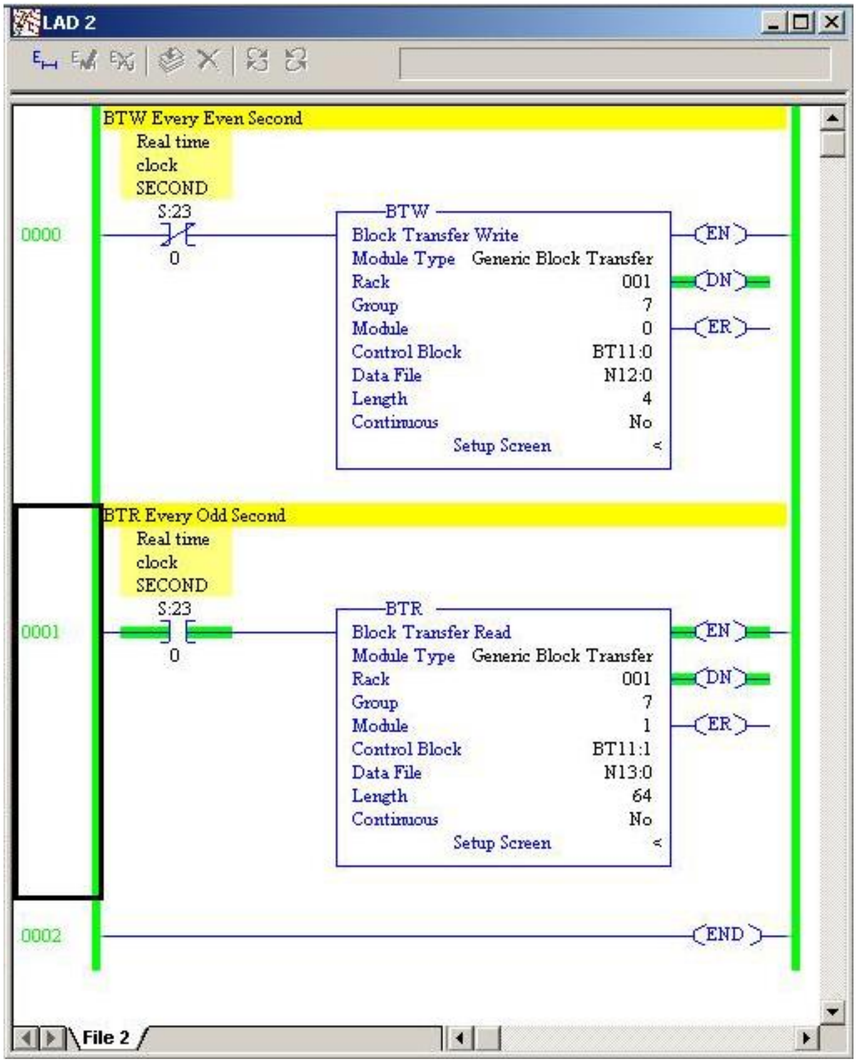
| Offset | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| N13:0 | 0 | 2 | 51 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| N13:10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| N13:20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

For this example, N13 is used for the CIP message response.
N13:0 ....................General Status (0xffff=-1=Busy, 0x0000=No Error)
N13:1 ....................Response Data Length
OR ExtSTS if General Status is not 0
N13:2 - 63 .............CIP Response Data

The data file length must be enough to read the 2-word status header plus the reply data.

General and Ext Status should be checked before processing the response data.

In the example, reading the Identity Object, the return size is variable so to be safe we read extra data in the BTR as shown:



BTW Every Even Second
Real time clock SECOND

0000 S:23 0

BTW
Block Transfer Write
Module Type Generic Block Transfer
Rack 001
Group 7
Module 0
Control Block BT11:0
Data File N12:0
Length 4
Continuous No
Setup Screen

(EN) (DN) (ER)

BTR Every Odd Second
Real time clock SECOND

0001 S:23 0

BTR
Block Transfer Read
Module Type Generic Block Transfer
Rack 001
Group 7
Module 1
Control Block BT11:1
Data File N13:0
Length 64
Continuous No
Setup Screen

(EN) (DN) (ER)

0002 (END)

### 3.3.6 Define Drive `Inputs` and `Outputs`

Consult the drive manual for the list of available data for your drive. It might look like the following:

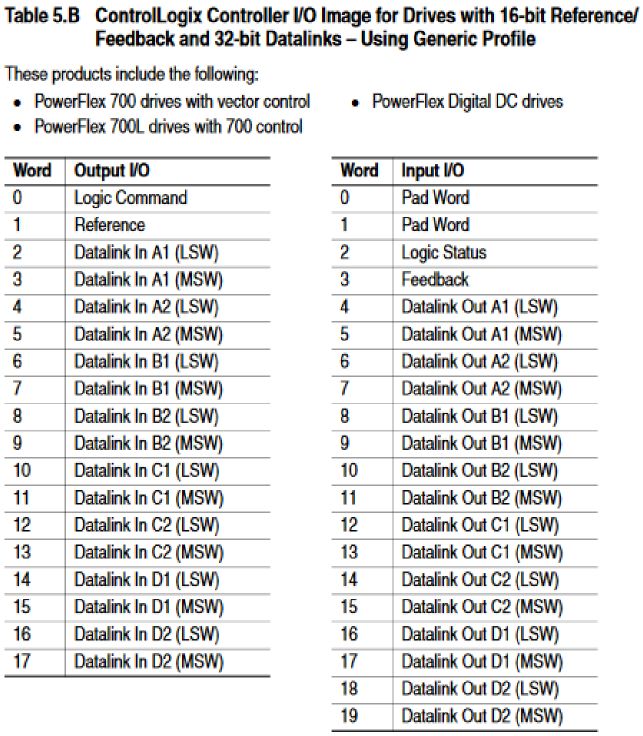


**Table 5.B ControlLogix Controller I/O Image for Drives with 16-bit Reference/ Feedback and 32-bit Datalinks – Using Generic Profile**

These products include the following:

- PowerFlex 700 drives with vector control
- PowerFlex 700L drives with 700 control
- PowerFlex Digital DC drives

| Word | Output I/O |
|---|---|
| 0 | Logic Command |
| 1 | Reference |
| 2 | Datalink In A1 (LSW) |
| 3 | Datalink In A1 (MSW) |
| 4 | Datalink In A2 (LSW) |
| 5 | Datalink In A2 (MSW) |
| 6 | Datalink In B1 (LSW) |
| 7 | Datalink In B1 (MSW) |
| 8 | Datalink In B2 (LSW) |
| 9 | Datalink In B2 (MSW) |
| 10 | Datalink In C1 (LSW) |
| 11 | Datalink In C1 (MSW) |
| 12 | Datalink In C2 (LSW) |
| 13 | Datalink In C2 (MSW) |
| 14 | Datalink In D1 (LSW) |
| 15 | Datalink In D1 (MSW) |
| 16 | Datalink In D2 (LSW) |
| 17 | Datalink In D2 (MSW) |

| Word | Input I/O |
|---|---|
| 0 | Pad Word |
| 1 | Pad Word |
| 2 | Logic Status |
| 3 | Feedback |
| 4 | Datalink Out A1 (LSW) |
| 5 | Datalink Out A1 (MSW) |
| 6 | Datalink Out A2 (LSW) |
| 7 | Datalink Out A2 (MSW) |
| 8 | Datalink Out B1 (LSW) |
| 9 | Datalink Out B1 (MSW) |
| 10 | Datalink Out B2 (LSW) |
| 11 | Datalink Out B2 (MSW) |
| 12 | Datalink Out C1 (LSW) |
| 13 | Datalink Out C1 (MSW) |
| 14 | Datalink Out C2 (LSW) |
| 15 | Datalink Out C2 (MSW) |
| 16 | Datalink Out D1 (LSW) |
| 17 | Datalink Out D1 (MSW) |
| 18 | Datalink Out D2 (LSW) |
| 19 | Datalink Out D2 (MSW) |

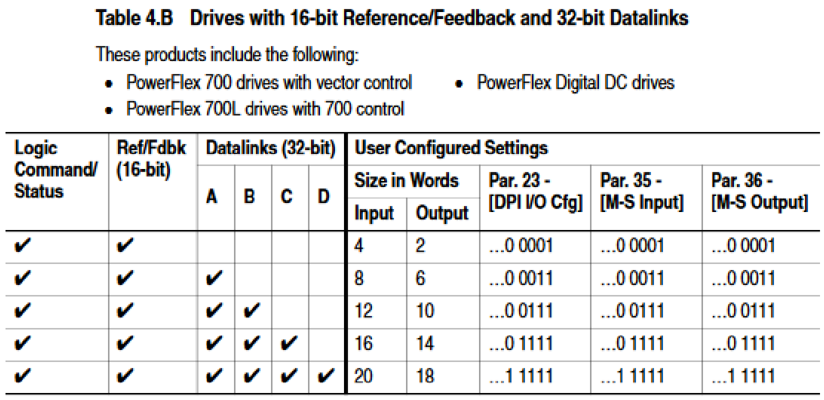
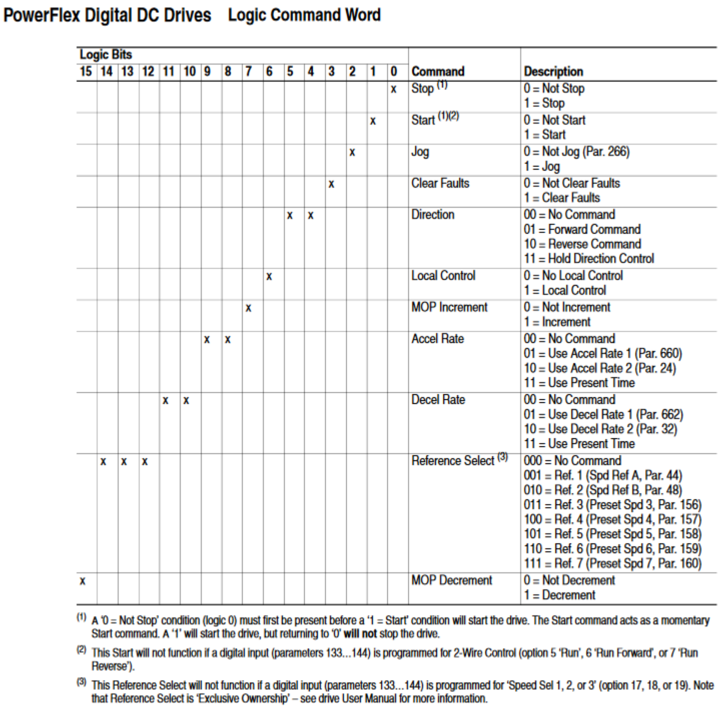


**Table 4.B Drives with 16-bit Reference/Feedback and 32-bit Datalinks**

These products include the following:

- PowerFlex 700 drives with vector control
- PowerFlex 700L drives with 700 control
- PowerFlex Digital DC drives

| Logic Command/ Status | Ref/Fdbk (16-bit) | Datalinks (32-bit) | | | | User Configured Settings | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | Size in Words | | Par. 23 - [DPI I/O Cfg] | Par. 35 - [M-S Input] | Par. 36 - [M-S Output] |
| | | A | B | C | D | Input | Output | | | |
| ✔ | ✔ | | | | | 4 | 2 | ...0 0001 | ...0 0001 | ...0 0001 |
| ✔ | ✔ | ✔ | | | | 8 | 6 | ...0 0011 | ...0 0011 | ...0 0011 |
| ✔ | ✔ | ✔ | ✔ | | | 12 | 10 | ...0 0111 | ...0 0111 | ...0 0111 |
| ✔ | ✔ | ✔ | ✔ | ✔ | | 16 | 14 | ...0 1111 | ...0 1111 | ...0 1111 |
| ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | 20 | 18 | ...1 1111 | ...1 1111 | ...1 1111 |



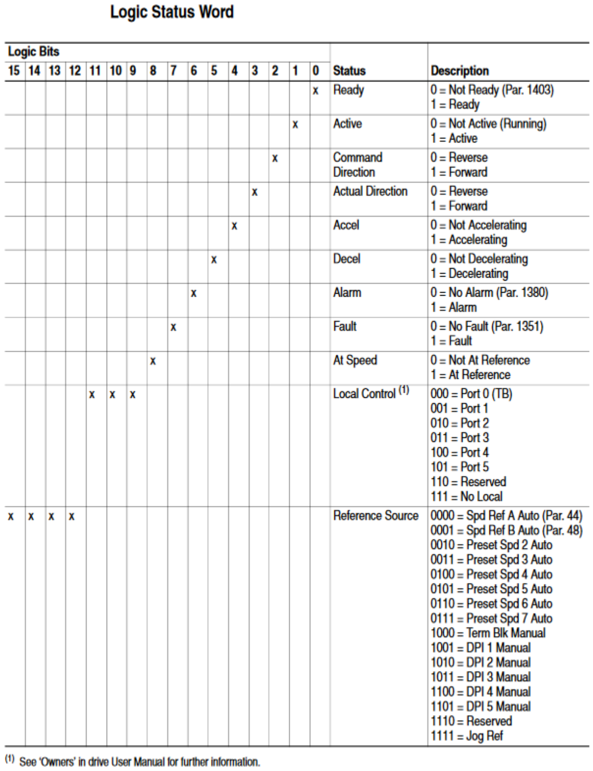
**PowerFlex Digital DC Drives Logic Command Word**

| Logic Bits | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | **Command** | **Description** |
| | | | | | | | | | | | | | | | x | Stop [1] | 0 = Not Stop<br>1 = Stop |
| | | | | | | | | | | | | | | x | | Start [1][2] | 0 = Not Start<br>1 = Start |
| | | | | | | | | | | | | | x | | | Jog | 0 = Not Jog (Par. 266)<br>1 = Jog |
| | | | | | | | | | | | | x | | | | Clear Faults | 0 = Not Clear Faults<br>1 = Clear Faults |
| | | | | | | | | | | x | x | | | | | Direction | 00 = No Command<br>01 = Forward Command<br>10 = Reverse Command<br>11 = Hold Direction Control |
| | | | | | | | | | x | | | | | | | Local Control | 0 = No Local Control<br>1 = Local Control |
| | | | | | | | | x | | | | | | | | MOP Increment | 0 = Not Increment<br>1 = Increment |
| | | | | | | x | x | | | | | | | | | Accel Rate | 00 = No Command<br>01 = Use Accel Rate 1 (Par. 660)<br>10 = Use Accel Rate 2 (Par. 24)<br>11 = Use Present Time |
| | | | | x | x | | | | | | | | | | | Decel Rate | 00 = No Command<br>01 = Use Decel Rate 1 (Par. 662)<br>10 = Use Decel Rate 2 (Par. 32)<br>11 = Use Present Time |
| | x | x | x | | | | | | | | | | | | | Reference Select [3] | 000 = No Command<br>001 = Ref. 1 (Spd Ref A, Par. 44)<br>010 = Ref. 2 (Spd Ref B, Par. 48)<br>011 = Ref. 3 (Preset Spd 3, Par. 156)<br>100 = Ref. 4 (Preset Spd 4, Par. 157)<br>101 = Ref. 5 (Preset Spd 5, Par. 158)<br>110 = Ref. 6 (Preset Spd 6, Par. 159)<br>111 = Ref. 7 (Preset Spd 7, Par. 160) |
| x | | | | | | | | | | | | | | | | MOP Decrement | 0 = Not Decrement<br>1 = Decrement |

[1] A '0 = Not Stop' condition (logic 0) must first be present before a '1 = Start' condition will start the drive. The Start command acts as a momentary Start command. A '1' will start the drive, but returning to '0' **will not** stop the drive.

[2] This Start will not function if a digital input (parameters 133...144) is programmed for 2-Wire Control (option 5 'Run', 6 'Run Forward', or 7 'Run Reverse').

[3] This Reference Select will not function if a digital input (parameters 133...144) is programmed for 'Speed Sel 1, 2, or 3' (option 17, 18, or 19). Note that Reference Select is 'Exclusive Ownership' – see drive User Manual for more information.



**Logic Status Word**

| Logic Bits | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | **Status** | **Description** |
| | | | | | | | | | | | | | | | x | Ready | 0 = Not Ready (Par. 1403)<br>1 = Ready |
| | | | | | | | | | | | | | | x | | Active | 0 = Not Active (Running)<br>1 = Active |
| | | | | | | | | | | | | | x | | | Command Direction | 0 = Reverse<br>1 = Forward |
| | | | | | | | | | | | | x | | | | Actual Direction | 0 = Reverse<br>1 = Forward |
| | | | | | | | | | | | x | | | | | Accel | 0 = Not Accelerating<br>1 = Accelerating |
| | | | | | | | | | | x | | | | | | Decel | 0 = Not Decelerating<br>1 = Decelerating |
| | | | | | | | | | x | | | | | | | Alarm | 0 = No Alarm (Par. 1380)<br>1 = Alarm |
| | | | | | | | | x | | | | | | | | Fault | 0 = No Fault (Par. 1351)<br>1 = Fault |
| | | | | | | | x | | | | | | | | | At Speed | 0 = Not At Reference<br>1 = At Reference |
| | | | | x | x | x | | | | | | | | | | Local Control [1] | 000 = Port 0 (TB)<br>001 = Port 1<br>010 = Port 2<br>011 = Port 3<br>100 = Port 4<br>101 = Port 5<br>110 = Reserved<br>111 = No Local |
| x | x | x | x | | | | | | | | | | | | | Reference Source | 0000 = Spd Ref A Auto (Par. 44)<br>0001 = Spd Ref B Auto (Par. 48)<br>0010 = Preset Spd 2 Auto<br>0011 = Preset Spd 3 Auto<br>0100 = Preset Spd 4 Auto<br>0101 = Preset Spd 5 Auto<br>0110 = Preset Spd 6 Auto<br>0111 = Preset Spd 7 Auto<br>1000 = Term Blk Manual<br>1001 = DPI 1 Manual<br>1010 = DPI 2 Manual<br>1011 = DPI 3 Manual<br>1100 = DPI 4 Manual<br>1101 = DPI 5 Manual<br>1110 = Reserved<br>1111 = Jog Ref |

[1] See 'Owners' in drive User Manual for further information.

From the available drive data, construct a list of tags. The Input tag list begins with the keyword `Inputs` followed by the size of the tag list *including overhead*. Determining size is discussed in the next section. The list ends with the keyword `EndInputs`.

Tag definitions consist of the keyword `EthTag`, a tag name (up to 63 characters), the data type and location. The location is expressed as an offset into the table of Ethernet data. The location is enclosed in square brackets.

```
EthTag DriveStatus_Ready         int[0].0
```

The data type can be `int`, `dint` or `real`.

To define a tag for a bit within an int or a dint, append a dot and then the bit number in decimal, 0-15 for ints and 0-31 for dints.
There can be up to 128 input and 128 output tags.

```
Inputs ##  ;  Size TBD in next section
EthTag DriveStatus_Ready         int[0].0
EthTag DriveStatus_Active        int[0].1
EthTag DriveStatus_CommandDir    int[0].2
EthTag DriveStatus_ActualDir     int[0].3
EthTag DriveStatus_Accelerating  int[0].4
EthTag DriveStatus_Decelerating  int[0].5
EthTag DriveStatus_Alarm         int[0].6
EthTag DriveStatus_Faulted       int[0].7
EthTag DriveStatus_AtSpeed       int[0].8
EthTag DriveStatus_Manual0       int[0].9
EthTag DriveStatus_Manual1       int[0].10
EthTag DriveStatus_Manual2       int[0].11
EthTag DriveStatus_SpdRefBit0    int[0].12
EthTag DriveStatus_SpdRefBit1    int[0].13
EthTag DriveStatus_SpdRefBit2    int[0].14
EthTag DriveStatus_SpdRefBit3    int[0].15
EthTag Feedback                  int[1]
EndInputs
```

***Important***: The data type for Feedback might be `dint` or `real`.

The Output tag list begins with the keyword `Outputs` followed by the size of the tag list *including overhead*. Determining size is discussed in the next section. The list ends with the keyword `EndOutputs`.

```
Outputs ##  ;  Size TBD in next section
EthTag LogicCommand_Stop         int[0].0
EthTag LogicCommand_Start        int[0].1
EthTag LogicCommand_Jog          int[0].2
EthTag LogicCommand_ClearFaults  int[0].3
EthTag LogicCommand_Forward      int[0].4
EthTag LogicCommand_Reverse      int[0].5
EthTag LogicCommand_Manual       int[0].6
EthTag LogicCommand_Increment    int[0].7
EthTag LogicCommand_AccelRate0   int[0].8
EthTag LogicCommand_AccelRate1   int[0].9
EthTag LogicCommand_DecelRate0   int[0].10
EthTag LogicCommand_DecelRate1   int[0].11
EthTag LogicCommand_SpdRefSel0   int[0].12
EthTag LogicCommand_SpdRefSel1   int[0].13
EthTag LogicCommand_SpdRefSel2   int[0].14
EthTag LogicCommand_Decrement    int[0].15
EthTag Reference                 int[1]
EndOutputs
```

***Important***: The data type for Reference might be `dint` or `real`.

The default drive data offset is a 16-bit word. The offset can instead be expressed as a `BYTE`, `DINT` or `REAL` by prefacing the offset value.
Tag0 is a 16-bit int read from int 1 of the Ethernet data table

```
EthTag Tag0 int[1]
```

Tag1 is a 16-bit int read from bytes 1 and 2 of the Ethernet data table.

```
EthTag Tag1 int[BYTE 1]
```

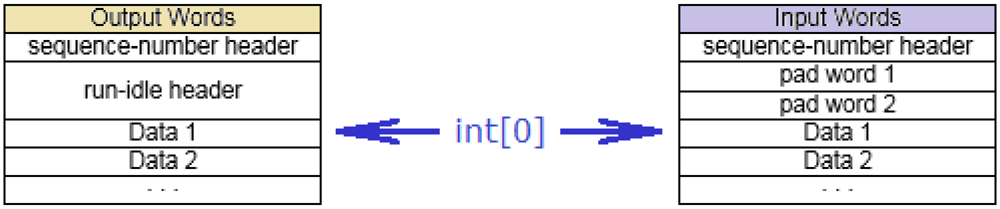
Tag2 is a 32-bit dint read from ints 3 and 4 of the Ethernet data table

```
EthTag Tag2 dint[INT 3]
```

***Please note for non-Allen-Bradley drives:***

CIP messages contain additional header data (3 words output, 1 word input). Allen-Bradley drives include pad words in their input data.

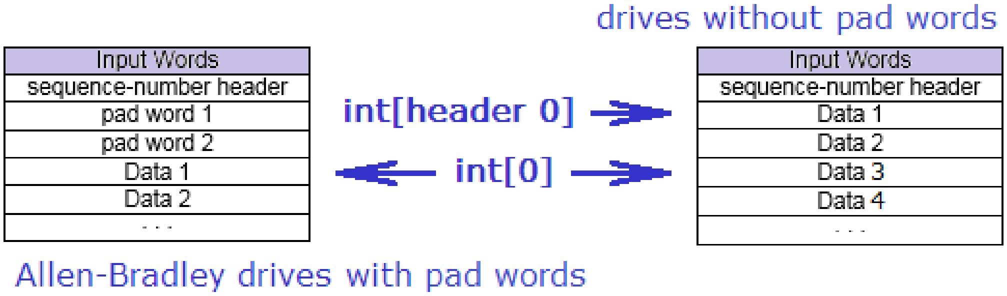
`EthTag` *`tag`* `int[`**`0`**`]` references the first word after the headers and pad words.



| Output Words |
|---|
| sequence-number header |
| run-idle header |
| Data 1 |
| Data 2 |
| . . . |

| Input Words |
|---|
| sequence-number header |
| pad word 1 |
| pad word 2 |
| Data 1 |
| Data 2 |
| . . . |

Other manufacturers' drives may have no input pad words and may store usable data before `int[0]`. For these drives, use the keyword **`header`** to force the AN-X2 to read the first two words of data immediately after the header.



| Input Words |
|---|
| sequence-number header |
| pad word 1 |
| pad word 2 |
| Data 1 |
| Data 2 |
| . . . |

| Input Words |
|---|
| sequence-number header |
| Data 1 |
| Data 2 |
| Data 3 |
| Data 4 |
| . . . |

For drives without pad words:

```
EthTag Tag1 int[header 0].0 ; Data 1 bit0, no pad
EthTag Tag2 int[header 0].1 ; Data 1 bit1, no pad
EthTag Tag3 int[header 1]   ; Data 2,      no pad
EthTag Tag4 int[0]          ; Data 3,      no pad
EthTag Tag5 int[1]          ; Data 4,      no pad
```

`header` is applied only to the Input tags as necessary.
`header` will not apply to Output tags.

### 3.3.7 Determine `Inputs` and `Outputs` Sizes

Consult the drive manual for the size of your drive's EIP data. It might look like the following:

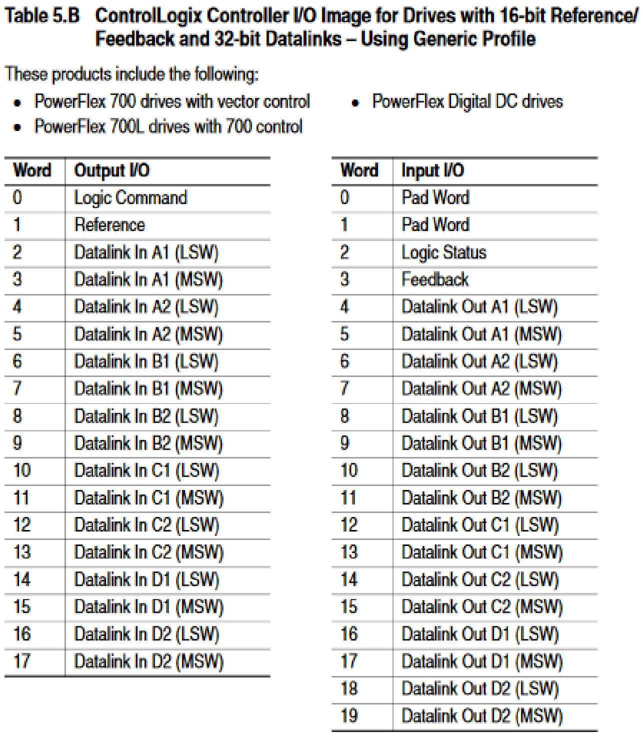


**Table 5.B ControlLogix Controller I/O Image for Drives with 16-bit Reference/ Feedback and 32-bit Datalinks – Using Generic Profile**

These products include the following:

- PowerFlex 700 drives with vector control
- PowerFlex 700L drives with 700 control
- PowerFlex Digital DC drives

| Word | Output I/O |
|---|---|
| 0 | Logic Command |
| 1 | Reference |
| 2 | Datalink In A1 (LSW) |
| 3 | Datalink In A1 (MSW) |
| 4 | Datalink In A2 (LSW) |
| 5 | Datalink In A2 (MSW) |
| 6 | Datalink In B1 (LSW) |
| 7 | Datalink In B1 (MSW) |
| 8 | Datalink In B2 (LSW) |
| 9 | Datalink In B2 (MSW) |
| 10 | Datalink In C1 (LSW) |
| 11 | Datalink In C1 (MSW) |
| 12 | Datalink In C2 (LSW) |
| 13 | Datalink In C2 (MSW) |
| 14 | Datalink In D1 (LSW) |
| 15 | Datalink In D1 (MSW) |
| 16 | Datalink In D2 (LSW) |
| 17 | Datalink In D2 (MSW) |

| Word | Input I/O |
|---|---|
| 0 | Pad Word |
| 1 | Pad Word |
| 2 | Logic Status |
| 3 | Feedback |
| 4 | Datalink Out A1 (LSW) |
| 5 | Datalink Out A1 (MSW) |
| 6 | Datalink Out A2 (LSW) |
| 7 | Datalink Out A2 (MSW) |
| 8 | Datalink Out B1 (LSW) |
| 9 | Datalink Out B1 (MSW) |
| 10 | Datalink Out B2 (LSW) |
| 11 | Datalink Out B2 (MSW) |
| 12 | Datalink Out C1 (LSW) |
| 13 | Datalink Out C1 (MSW) |
| 14 | Datalink Out C2 (LSW) |
| 15 | Datalink Out C2 (MSW) |
| 16 | Datalink Out D1 (LSW) |
| 17 | Datalink Out D1 (MSW) |
| 18 | Datalink Out D2 (LSW) |
| 19 | Datalink Out D2 (MSW) |

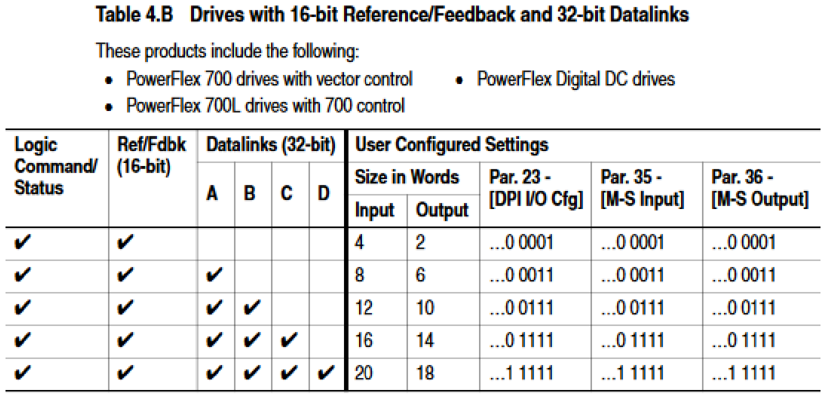


**Table 4.B Drives with 16-bit Reference/Feedback and 32-bit Datalinks**

These products include the following:

- PowerFlex 700 drives with vector control
- PowerFlex 700L drives with 700 control
- PowerFlex Digital DC drives

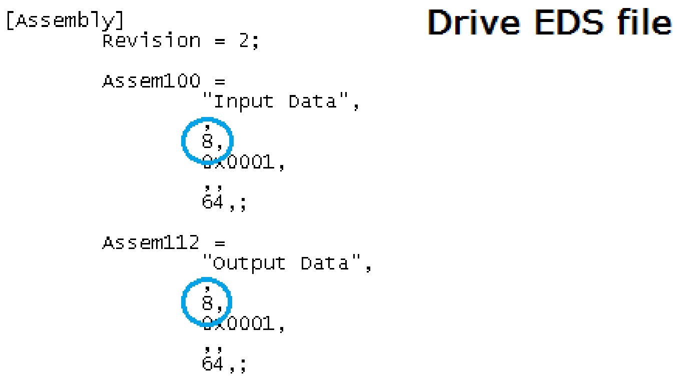
| Logic Command/ Status | Ref/Fdbk (16-bit) | Datalinks (32-bit) | | | | User Configured Settings | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | A | B | C | D | Size in Words | | Par. 23 - [DPI I/O Cfg] | Par. 35 - [M-S Input] | Par. 36 - [M-S Output] |
| | | | | | | Input | Output | | | |
| ✔ | ✔ | | | | | 4 | 2 | ...0 0001 | ...0 0001 | ...0 0001 |
| ✔ | ✔ | ✔ | | | | 8 | 6 | ...0 0011 | ...0 0011 | ...0 0011 |
| ✔ | ✔ | ✔ | ✔ | | | 12 | 10 | ...0 0111 | ...0 0111 | ...0 0111 |
| ✔ | ✔ | ✔ | ✔ | ✔ | | 16 | 14 | ...0 1111 | ...0 1111 | ...0 1111 |
| ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | 20 | 18 | ...1 1111 | ...1 1111 | ...1 1111 |

Find the length of the input and output data for your drive.

The length of the Input data in the Allen-Bradley table above also includes two Input pad words (4 bytes). Other manufacturers' drives might have no input pad data.

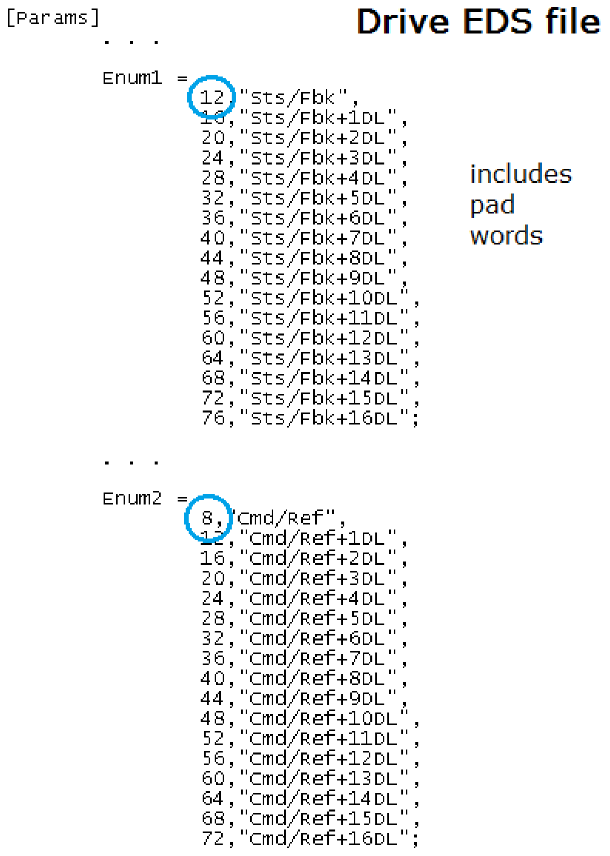
***Important***: Allen-Bradley Datalinks allow the transfer of additional drive data. In the table above, each Datalink contains two tags. Datalink A contains tags A1 and A2. For this drive, you must count the size of the entire two-tag datalink, even if you use only one of the two tags.

Input and output sizes can be found in the drive's EDS file. Choose the appropriate sizes for your Assembly Instances (`O_T_ConnPt`, `T_O_ConnPt`, selected in section 3.3.1).



```
[Assembly]
        Revision = 2;

        Assem100 =
                "Input Data",
                ,
                8,
                0x0001,
                ,,
                64,;

        Assem112 =
                "Output Data",
                ,
                8,
                0x0001,
                ,,
                64,;
```

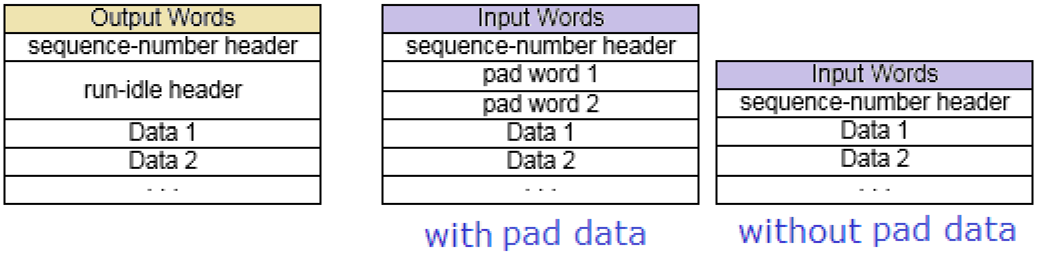
When Datalinks are available, multiple sizes may be listed. Select the sizes appropriate for the number of Datalinks enabled in the drive. (Status, Command, Feedback, Reference and Datalinks are all 4-bytes in this EDS.)



```
[Params]
       . . .

       Enum1 =
               12,"Sts/Fbk",
               16,"Sts/Fbk+1DL",
               20,"Sts/Fbk+2DL",
               24,"Sts/Fbk+3DL",
               28,"Sts/Fbk+4DL",
               32,"Sts/Fbk+5DL",
               36,"Sts/Fbk+6DL",
               40,"Sts/Fbk+7DL",
               44,"Sts/Fbk+8DL",
               48,"Sts/Fbk+9DL",
               52,"Sts/Fbk+10DL",
               56,"Sts/Fbk+11DL",
               60,"Sts/Fbk+12DL",
               64,"Sts/Fbk+13DL",
               68,"Sts/Fbk+14DL",
               72,"Sts/Fbk+15DL",
               76,"Sts/Fbk+16DL";

       . . .

       Enum2 =
                8,"Cmd/Ref",
               12,"Cmd/Ref+1DL",
               16,"Cmd/Ref+2DL",
               20,"Cmd/Ref+3DL",
               24,"Cmd/Ref+4DL",
               28,"Cmd/Ref+5DL",
               32,"Cmd/Ref+6DL",
               36,"Cmd/Ref+7DL",
               40,"Cmd/Ref+8DL",
               44,"Cmd/Ref+9DL",
               48,"Cmd/Ref+10DL",
               52,"Cmd/Ref+11DL",
               56,"Cmd/Ref+12DL",
               60,"Cmd/Ref+13DL",
               64,"Cmd/Ref+14DL",
               68,"Cmd/Ref+15DL",
               72,"Cmd/Ref+16DL";
```

CIP messages contain additional header data that must also be counted in the input and output size:

- 2 bytes overhead for sequence-number (output and input).
- 4 bytes overhead for run-idle (output).



| Output Words |
|---|
| sequence-number header |
| run-idle header |
| Data 1 |
| Data 2 |
| . . . |

| Input Words |
|---|
| sequence-number header |
| pad word 1 |
| pad word 2 |
| Data 1 |
| Data 2 |
| . . . |

with pad data

| Input Words |
|---|
| sequence-number header |
| Data 1 |
| Data 2 |
| . . . |

without pad data

To determine the size of the EIP Inputs and Outputs, add the data byte counts together with any overhead byte counts.

4-bytes data (2-byte Logic Command, 2-byte Speed Reference) plus 6 bytes (overhead) equals 10 bytes of Output.

```
Outputs 10
EthTag LogicCommand int[0]
EthTag Reference    int[1]
EndInputs
```

For an Allen-Bradley drive with pad words:
8-bytes data (2-byte Drive Status, 2-byte Speed Feedback, 4-bytes pad words) plus 2 bytes (overhead) equals 10 bytes of Input.

```
Inputs 10
EthTag DriveStatus int[0]
EthTag Feedback    int[1]
EndInputs
```

For a drive without pad words:
4-bytes data (2-byte Drive Status, 2-byte Speed Feedback) plus 2 bytes (overhead) equals 6 bytes of Input.

```
Inputs 6
EthTag DriveStatus int[header 0]
EthTag Feedback    int[header 1]
EndInputs
```

## 3.4 Configure RIO Definition File

The RioDef file defines the tag mapping from Ethernet to RIO, scaling the tags as needed. A semi-colon ";" indicates a comment. Any text following a semi-colon is ignored by the AN-X2.

The configuration files are best edited in Notepad. The filename can be changed, but must match the name used in main.csv. The file extension must remain .csv.

RioDef.csv:

```
; -----------------------------------------
; RIO configuration file
; maps drive Ethernet tags to RIO, scaled as necessary
; -----------------------------------------

Inputs
I:0.0 <- DriveStatus_tag00
I:0.1 <- DriveStatus_tag01
I:1   <- Feedback * ##.##
EndInputs

Outputs
O:0.0       -> LogicCommand_tag00
O:0.1       -> LogicCommand_tag01
O:1 * ##.## -> Reference
EndOutputs
```

The I/O group is relative to the start of the rack to which the template is applied. If the rack is defined as a last quarter rack, and the input is I/O group 0, then I/O group 6 is used in the rack.

```
Rack, 0o01, 4, 4 ; rack number 1, last quarter
I:0 <- Tag1
```

| Quarter | 1 | | 2 | | 3 | | 4 | |
|---|---|---|---|---|---|---|---|---|
| I/O group | 0 | 1 | 2 | 3 | 4 | 5 | 6 | **7** |

Multiple drives defined at different starting remote I/O addresses can use the same RioDef.csv file. If a second drive's rack is defined as the third quarter rack, and the input is I/O group 0, then I/O group 4 is used in the rack

```
Rack, 0o01, 3, 3 ; rack number 1, third quarter
I:0 <- Tag1
```

| Quarter | 1 | | 2 | | 3 | | 4 | |
|---|---|---|---|---|---|---|---|---|
| I/O group | 0 | 1 | 2 | 3 | 4 | 5 | 6 | **7** |

If you make changes in RioDef.csv, all drives using the file will see those changes (after restarting the AN-X2).

### 3.4.1 Define RIO Inputs

Map the desired EthDef tags to RIO inputs.

It is not necessary to map all the EthDef tags. You may map just the tags you wish to use. You may map the tags to anywhere in your rack.

***Important***: Do not to map more than one Ethernet input to the same remote I/O input. The AN-X2 will allow this, but both mappings will be active and overwrite one another.

```
Inputs
I:0.0  <- DriveStatus_Ready
I:0.1  <- DriveStatus_Active
I:0.2  <- DriveStatus_CommandDir
I:0.3  <- DriveStatus_ActualDir
I:0.4  <- DriveStatus_Accelerating
I:0.5  <- DriveStatus_Decelerating
I:0.6  <- DriveStatus_Alarm
I:0.7  <- DriveStatus_Faulted
I:0.10 <- DriveStatus_AtSpeed
I:0.11 <- DriveStatus_Manual0
I:0.12 <- DriveStatus_Manual1
I:0.13 <- DriveStatus_Manual2
I:0.14 <- DriveStatus_SpdRefBit0
I:0.15 <- DriveStatus_SpdRefBit1
I:0.16 <- DriveStatus_SpdRefBit2
I:0.17 <- DriveStatus_SpdRefBit3
I:1.0  <- Feedback
EndInputs
```

Mapping begins with the keyword `Inputs` and ends with `EndInputs`.
Mappings consist of the input word or bit address, space(s), the symbol <–, space(s), and the Ethernet tag name from the EthDef file.
Word addresses are `I:n`, where n is the I/O group, from 0 to 7.
Bit addresses are `I:n.b`, where b is the bit number, from 0 to 17 octal.

If the input data type is REAL, the AN-X2 rounds the fraction and writes into the integer I:. 59.52 becomes 60. 59.48 becomes 59.

For bit mappings, the AN-X can flip a bit when it copies the data. Replace the dash in the assignment operator <– with a ~.

```
I:2.1 <~ Ready
```

If `Ready` contains **1**, then **0** will be written to `I:2.1`.

### 3.4.2 Define RIO `Outputs`

Map the desired RIO outputs to EthDef tags.

It is not necessary to map all the EthDef tags. You may map just the tags you wish to use.

***Important***: Do not to map more than one remote I/O output to the same Ethernet output. The AN-X2 will allow this, but both mappings will be active and overwrite one another.

```
Outputs
O:0.0  -> LogicCommand_Stop
O:0.1  -> LogicCommand_Start
O:0.2  -> LogicCommand_Jog
O:0.3  -> LogicCommand_ClearFaults
O:0.4  -> LogicCommand_Forward
O:0.5  -> LogicCommand_Reverse
O:0.6  -> LogicCommand_Manual
O:0.7  -> LogicCommand_Increment
O:0.10 -> LogicCommand_AccelRate0
O:0.11 -> LogicCommand_AccelRate1
O:0.12 -> LogicCommand_DecelRate0
O:0.13 -> LogicCommand_DecelRate1
O:0.14 -> LogicCommand_SpdRefSel0
O:0.15 -> LogicCommand_SpdRefSel1
O:0.16 -> LogicCommand_SpdRefSel2
O:0.17 -> LogicCommand_Decrement
O:1.0  -> Reference
EndOutputs
```

Mapping begins with the keyword `Outputs` and ends with `EndOutputs`.
Mappings consist of the output word or bit address, space, the symbol `->`, space, and the Ethernet tag name from the EthDef file.
Word addresses are `O:n`, where n is the I/O group, from 0 to 7.
Bit addresses are `O:n.b`, where b is the bit number, from 0 to 17 octal.

If the output data type is REAL, the AN-X2 writes the integer O: value into the REAL. 60 becomes 60.00.

For bit mappings, the AN-X can flip a bit when it copies the data. Replace the dash in the assignment operator `->` with a `~`.

```
O:2.0 ~> Halt
```

If `O:2.0` contains **1**, then **0** will be written to `Halt`.

### 3.4.3 Scale Inputs/Outputs As Needed

***Important***: It is essential that you thoroughly understand the nature of the data being passed and the range of values for the remote I/O data and the allowed range of values for the target Ethernet drive.

***Important***: The AN-X does not perform any checking on scaled data out of range. We strongly recommend that, wherever possible, you perform scaling in the Ethernet drive rather than in the AN-X.

For output words, the scaling is appended to the remote I/O address.

```
O:2 * 9 / 17 -> Speed     ; int or dint data type
O:3 * 1.7e3  -> Torque    ; real data type
```

For input words, the scaling is appended to the Ethernet tag name.

```
I:2 <- Position * 15 / 79 ; int or dint data type
I:3 <- Feedback * 0.33    ; real data type
```

Leave a space between the tag name and scaling parameters for input scaling

Acceptable scaling values depend on the drive tag data type. Refer to the drive’s EthDef file for tag type.

- For data type int and dint, the scaling can contain an integer multiplier, divisor or combination of both.
- For data type real, the scaling must be a single floating point multiplier.

**Example**: In the drive, speed Feedback is a real value with a range of 0-60 Hz. To represent Feedback as a 16-bit value in the PLC with a range 0-32767:

```
I:1 <- Feedback * 546.12   ; (32767 / 60)
```

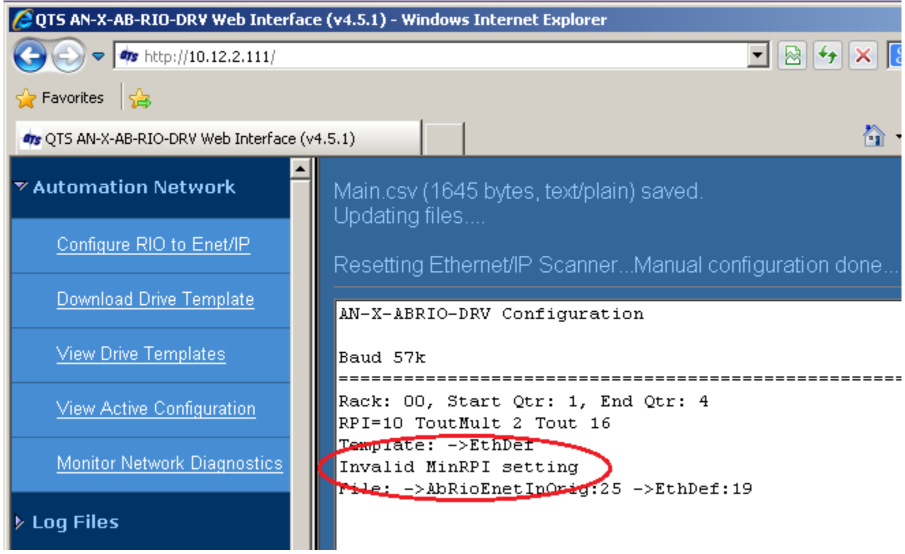
To convert speed Reference in the PLC 0-32767, to real Speed in the drive 0-60 Hz:

```
O:1 * 0.001831 -> Reference ; (60 / 32767)
```

## 3.5 Download Configured Files to the AN-X2

Open a web-browser and enter the current AN-X2 IP in the browser address.

***Important***: When executing the following steps, note any errors that might appear in the text window. For example:


QTS AN-X-AB-RIO-DRV Web Interface (v4.5.1) - Windows Internet Explorer
http://10.12.2.111/
Favorites
QTS AN-X-AB-RIO-DRV Web Interface (v4.5.1)
Automation Network
Configure RIO to Enet/IP
Download Drive Template
View Drive Templates
View Active Configuration
Monitor Network Diagnostics
Log Files

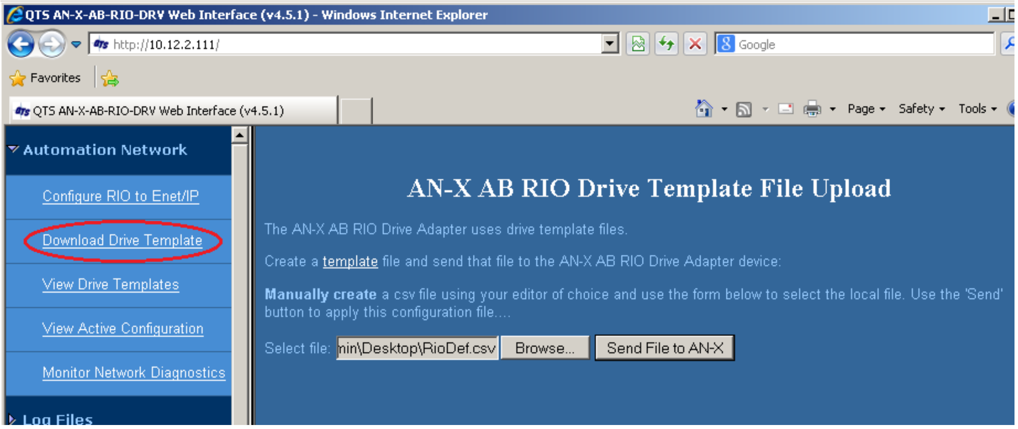
Main.csv (1645 bytes, text/plain) saved.
Updating files....

Resetting Ethernet/IP Scanner...Manual configuration done...

```
AN-X-ABRIO-DRV Configuration

Baud 57k
=====================================================
Rack: 00, Start Qtr: 1, End Qtr: 4
RPI=10 ToutMult 2 Tout 16
Template: ->EthDef
Invalid MinRPI setting
File: ->AbRioEnetInOrig:25 ->EthDef:19
```

### 3.5.1 Download EtherNet/IP Definition File

Select *Download Drive Template* again, browse to the EtherNet/IP definition file and *Send File to AN-X*. Note and correct any errors in the response text window.


QTS AN-X-AB-RIO-DRV Web Interface (v4.5.1) - Windows Internet Explorer
http://10.12.2.111/
Google
Favorites
QTS AN-X-AB-RIO-DRV Web Interface (v4.5.1)
Page
Safety
Tools
Automation Network
Configure RIO to Enet/IP
Download Drive Template
View Drive Templates
View Active Configuration
Monitor Network Diagnostics
Log Files

AN-X AB RIO Drive Template File Upload

The AN-X AB RIO Drive Adapter uses drive template files.

Create a template file and send that file to the AN-X AB RIO Drive Adapter device:

**Manually create** a csv file using your editor of choice and use the form below to select the local file. Use the 'Send' button to apply this configuration file....

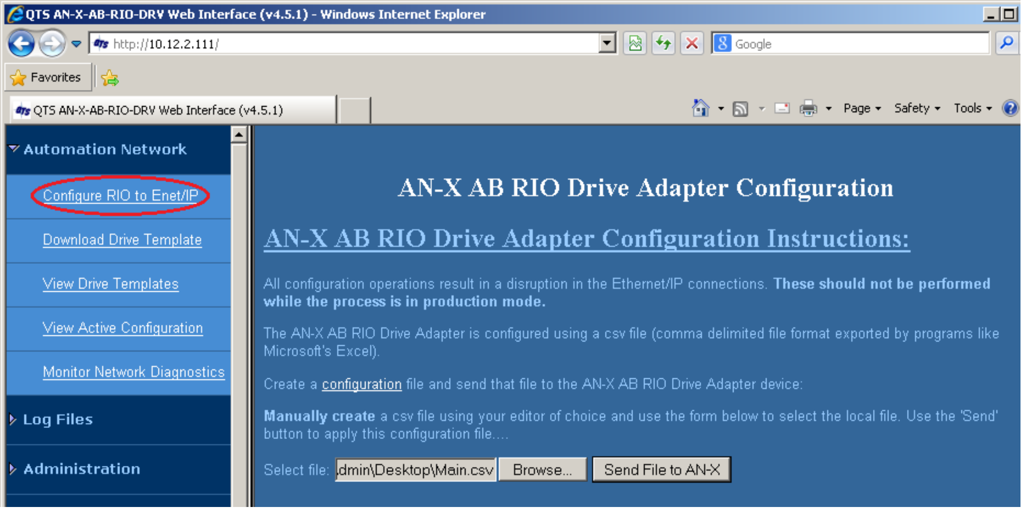
Select file: nin\Desktop\RioDef.csv Browse... Send File to AN-X

### 3.5.2 Download RIO Definition File

Select *Download Drive Template* in the *Automation Network* menus on the left. Browse to the RIO definition file and *Send File to AN-X*. Note and correct any errors in the response text window.

### 3.5.3 Download Main Definition File

Select *Configure RIO to Enet/IP* in the *Automation Network* menus on the left. Browse to the Main definition file and *Send File to AN-X*. Note and correct any errors in the response text window.



AN-X AB RIO Drive Adapter Configuration

AN-X AB RIO Drive Adapter Configuration Instructions:

All configuration operations result in a disruption in the Ethernet/IP connections. **These should not be performed while the process is in production mode.**

The AN-X AB RIO Drive Adapter is configured using a csv file (comma delimited file format exported by programs like Microsoft's Excel).

Create a configuration file and send that file to the AN-X AB RIO Drive Adapter device:

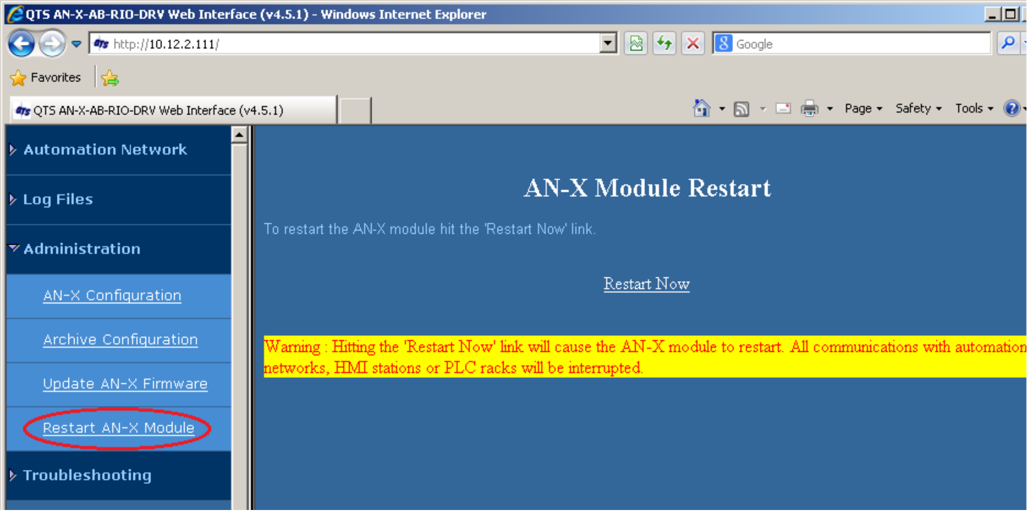
**Manually create** a csv file using your editor of choice and use the form below to select the local file. Use the 'Send' button to apply this configuration file....

Select file: dmin\Desktop\Main.csv

### 3.5.4 Restart the AN-X2

Changes will take effect after restarting the AN-X2.

Select *Restart AN-X Module* in the *Administration* menus on the left. Browse to the Main definition file and *Send File to AN-X*. Select *Restart Now*.



AN-X Module Restart

To restart the AN-X module hit the 'Restart Now' link.

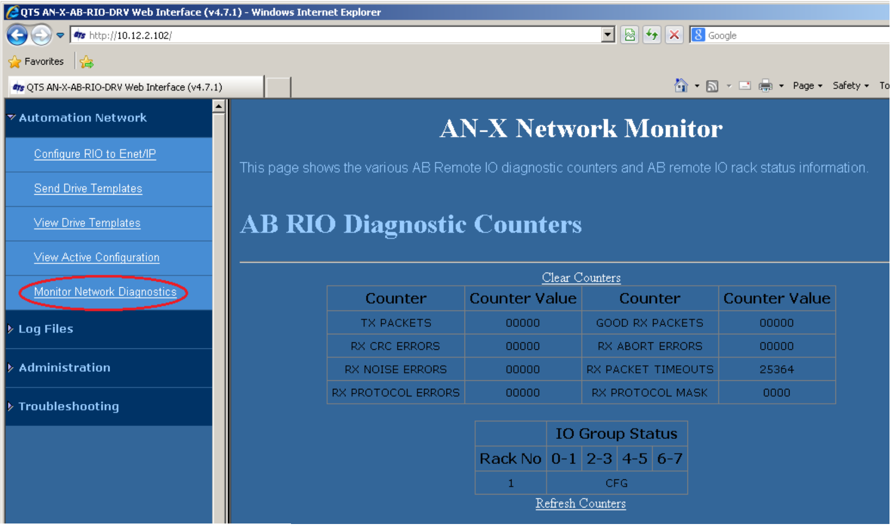
Warning : Hitting the 'Restart Now' link will cause the AN-X module to restart. All communications with automation networks, HMI stations or PLC racks will be interrupted.

## 3.6 Share EthDef.csv

If you have created a new drive EthDef.csv file, please share the tested file with ProSoft Technical Support. Include the drive model.

# 4 Appendix: Things to check

If Monitor Network Diagnostics shows no RIO communications, try swapping the RIO wires. When you move the RIO connector from the old drive to the AN-X2, you might need to swap Lines 1 and 2.



AN-X Network Monitor

This page shows the various AB Remote IO diagnostic counters and AB remote IO rack status information.

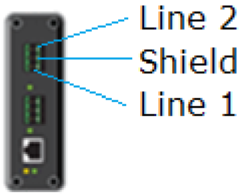
AB RIO Diagnostic Counters

Clear Counters

| Counter | Counter Value | Counter | Counter Value |
|---|---|---|---|
| TX PACKETS | 00000 | GOOD RX PACKETS | 00000 |
| RX CRC ERRORS | 00000 | RX ABORT ERRORS | 00000 |
| RX NOISE ERRORS | 00000 | RX PACKET TIMEOUTS | 25364 |
| RX PROTOCOL ERRORS | 00000 | RX PROTOCOL MASK | 0000 |

|  | IO Group Status |  |  |  |
|---|---|---|---|---|
| Rack No | 0-1 | 2-3 | 4-5 | 6-7 |
| 1 | CFG |  |  |  |

Refresh Counters



Verify the drive keying. If you're using a pre-configured drive template, the drive version may have changed. Check the keying in RSLinx and edit the EthDef file. (Section 3.3.2)

Verify the data type for Reference and Feedback. These might be `int`, `dint` or `real`.

```
EthTag Feedback            real[1]
```

If using data types `dint` or `real`, increment the subsequent tag location by 2 ints.

```
EthTag Feedback            real[1]
EthTag DataLink1           real[3]   ;increment by 2
EthTag DataLink2           real[5]   ;increment by 2
```

If you made changes to the EthDef or RioDef files, remember to also resend the Main file.

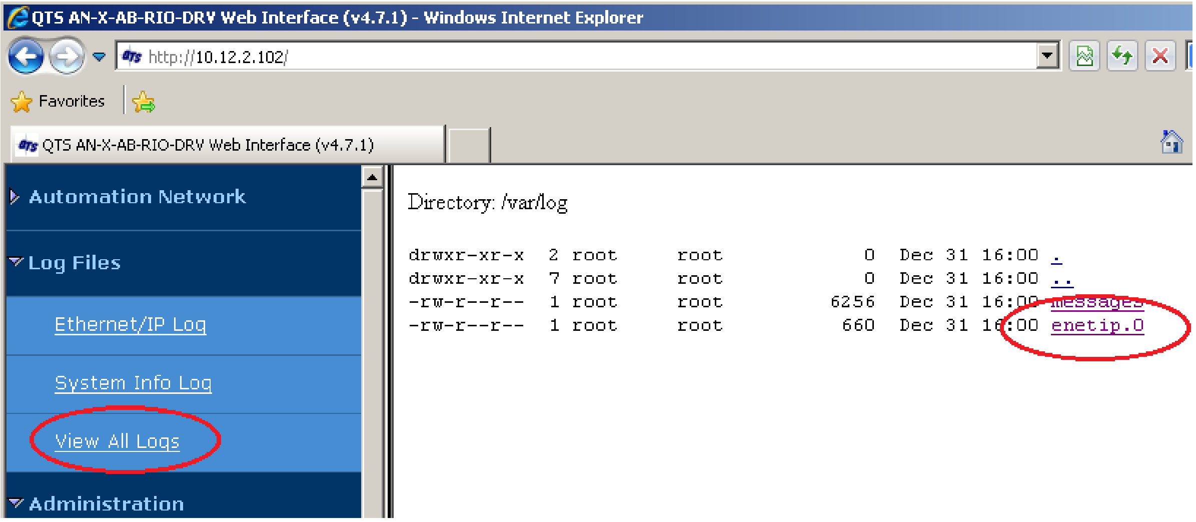
Edit or verify your files with a text editor. MS Excel can introduce quotation marks "" that will create errors in the AN-X2.

# 5 Appendix: Wireshark

Wireshark software is a network protocol analyzer available at wireshark.org. Wireshark can be an invaluable aid in determining your EIP configuration.

To capture all the network traffic, your network switch must capable of port mirroring. (A hub would also work, though not recommended long-term.)

Example:
The AN-X2 LEDs indicate an error and the enetip.0 log records a Forward Open error.


QTS AN-X-AB-RIO-DRV Web Interface (v4.7.1) - Windows Internet Explorer
http://10.12.2.102/
Favorites
QTS AN-X-AB-RIO-DRV Web Interface (v4.7.1)
Automation Network
Log Files
Ethernet/IP Log
System Info Log
View All Logs
Administration

Directory: /var/log

```
drwxr-xr-x   2 root     root            0  Dec 31 16:00 .
drwxr-xr-x   7 root     root            0  Dec 31 16:00 ..
-rw-r--r--   1 root     root         6256  Dec 31 16:00 messages
-rw-r--r--   1 root     root          660  Dec 31 16:00 enetip.0
```

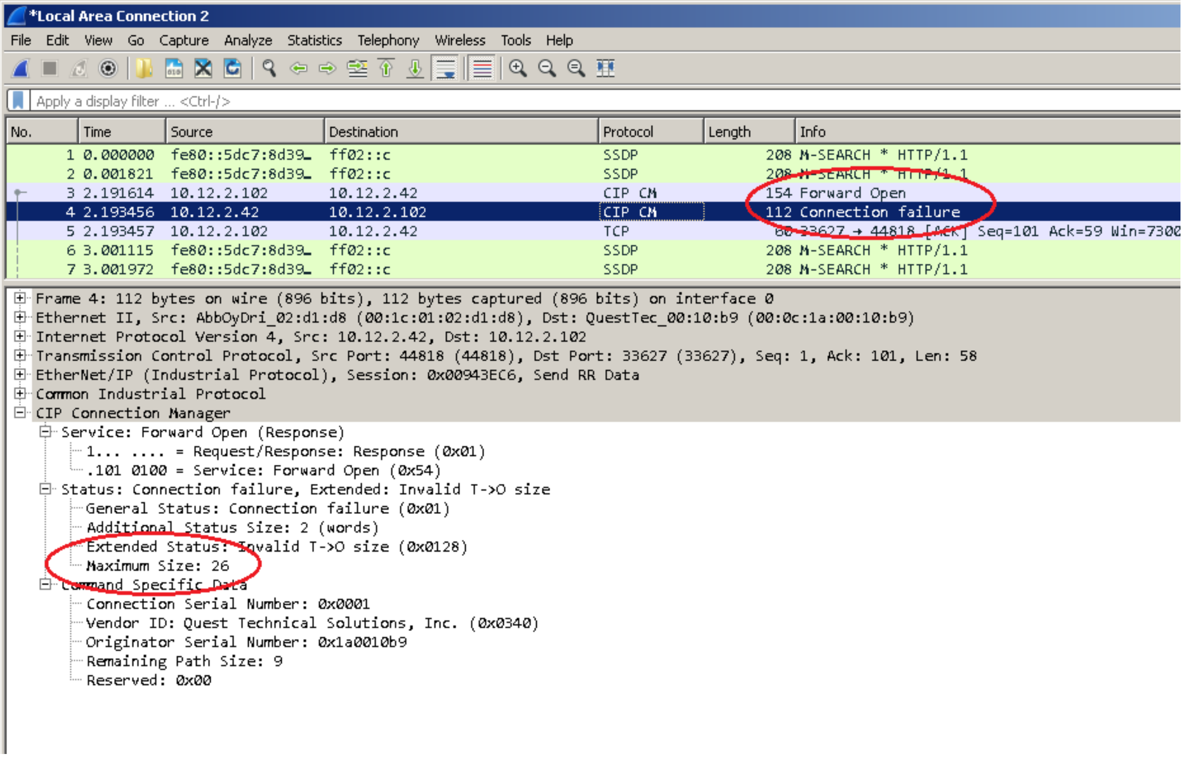
enetip.0

```
00:44.378 606 :CIP: Forward Open Request  [Cip=0 10.12.2.42]
00:44.383 516 :Forward Open Response Error: 01  [Cip=0 10.12.2.42]
00:44.384 317 :GenSts=01
00:44.385 020 :ExtSts=0128
00:44.385 655 :Unknown Extended Status
```

Consulting an online reference shows:

| |
|---|
| General Status Error 0x01 |
| Extended Status Error 0x0128 |
| Invalid Target to Originator Size |
| This error is returned when the Target to Originator (Input data) size specified in the Forward Open does not match what is in Target. |

A Wireshark capture also reveals the Forward Open error. Drilling down through the details we find the input size is 26.



| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 1 | 0.000000 | fe80::5dc7:8d39… | ff02::c | SSDP | 208 | M-SEARCH * HTTP/1.1 |
| 2 | 0.001821 | fe80::5dc7:8d39… | ff02::c | SSDP | 208 | M-SEARCH * HTTP/1.1 |
| 3 | 2.191614 | 10.12.2.102 | 10.12.2.42 | CIP CM | 154 | Forward Open |
| 4 | 2.193456 | 10.12.2.42 | 10.12.2.102 | CIP CM | 112 | Connection failure |
| 5 | 2.193457 | 10.12.2.102 | 10.12.2.42 | TCP | 60 | 33627 → 44818 [ACK] Seq=101 Ack=59 Win=7300 |
| 6 | 3.001115 | fe80::5dc7:8d39… | ff02::c | SSDP | 208 | M-SEARCH * HTTP/1.1 |
| 7 | 3.001972 | fe80::5dc7:8d39… | ff02::c | SSDP | 208 | M-SEARCH * HTTP/1.1 |

```
⊞ Frame 4: 112 bytes on wire (896 bits), 112 bytes captured (896 bits) on interface 0
⊞ Ethernet II, Src: AbbOyDri_02:d1:d8 (00:1c:01:02:d1:d8), Dst: QuestTec_00:10:b9 (00:0c:1a:00:10:b9)
⊞ Internet Protocol Version 4, Src: 10.12.2.42, Dst: 10.12.2.102
⊞ Transmission Control Protocol, Src Port: 44818 (44818), Dst Port: 33627 (33627), Seq: 1, Ack: 101, Len: 58
⊞ EtherNet/IP (Industrial Protocol), Session: 0x00943EC6, Send RR Data
⊞ Common Industrial Protocol
⊟ CIP Connection Manager
   ⊟ Service: Forward Open (Response)
        1... .... = Request/Response: Response (0x01)
        .101 0100 = Service: Forward Open (0x54)
   ⊟ Status: Connection failure, Extended: Invalid T->O size
        General Status: Connection failure (0x01)
        Additional Status Size: 2 (words)
        Extended Status: Invalid T->O size (0x0128)
        Maximum Size: 26
   ⊟ Command Specific Data
        Connection Serial Number: 0x0001
        Vendor ID: Quest Technical Solutions, Inc. (0x0340)
        Originator Serial Number: 0x1a0010b9
        Remaining Path Size: 9
        Reserved: 0x00
```

Modifying EthDef.csv, `Inputs` to **26** corrects the Forward Open error.

```
Inputs 26
EthTag . . .
EthTag . . .
. . .
EndInputs
```

**ProSoft Technology**
9201 Camino Media, Suite 200
Bakersfield, CA 93311
+1 (661) 716-5100
+1 (661) 716-5101 (Fax)
http://www.prosoft-technology.com