

# **Allen-Bradley Ethernet Device Driver Help**

© 2009 Kepware Technologies

# Table of Contents

<b>1</b>	<b>Getting Started.....</b>	<b>3</b>
	Help Contents.....	3
	Overview.....	3
<b>2</b>	<b>Device Setup.....</b>	<b>3</b>
	Device Setup.....	3
	Cable Connections.....	3
	Communications Parameters.....	4
	Protocol Parameters.....	4
	SLC500 Slot Configuration.....	4
	SLC500 Slot Configuration.....	4
	Modular I/O Selection Guide.....	5
<b>3</b>	<b>Optimizing Your AB Ethernet Communications.....</b>	<b>7</b>
	Optimizing Your AB Ethernet Communications.....	7
<b>4</b>	<b>Data Types Description.....</b>	<b>8</b>
	Data Types Description.....	8
<b>5</b>	<b>Address Descriptions.....</b>	<b>8</b>
	Address Descriptions.....	8
	General Addressing.....	8
	Output Files.....	9
	Input Files.....	10
	Status Files.....	11
	Binary Files.....	12
	Timer Files.....	13
	Counter Files.....	13
	Control Files.....	14
	Integer Files.....	14
	Float Files.....	15
	ASCII Files.....	16
	String Files.....	16
	SLC5/05 Open Addressing.....	17
	SLC5/05 Open Addressing.....	17
	PLC-5 Family Addressing.....	17
	PLC-5 Family Addressing.....	17
	BCD Files.....	17
	PID Files.....	17
	Message Files.....	19
	Block Transfer Files.....	19
<b>6</b>	<b>Error Descriptions.....</b>	<b>20</b>
	Error Descriptions.....	20
	Address Validation.....	21
	Address Validation.....	21
	Missing address.....	21
	Device address '<address>' contains a syntax error.....	21
	Address '<address>' is out of range for the specified device or register.....	21
	Device address '<address>' is not supported by model '<model name>'.....	21
	Data Type '<type>' is not valid for device address '<address>'.....	22
	Device address '<address>' is read-only.....	22

Array size is out of range for address '<address>'..... 22

Array support is not available for the specified address: '<address>'..... 22

**Communication Error Messages..... 23**

    Communications Error Messages..... 23

    Winsock initialization failed (OS Error = n)..... 23

    Winsock V1.1 or higher must be installed to use the Allen-Bradley Ethernet device driver..... 23

**Device Status Messages..... 23**

    Device Status Messages..... 23

    Device '<device name>' is not responding..... 23

    Unable to write to '<address>' on device '<device name>'..... 24

**Device Specific Messages..... 24**

    Device Specific Messages..... 24

    Unable to read data starting at address <address> on device '<device name>'. Frame receive..... 24

    Unable to read data starting at address <address> on device '<device name>'. [STS=<value>, EXT  
STS=<value>]. Tag deactivated..... 24

    Unable to write to address <address> on device '<device name>'. Frame received contains errors..... 25

    Unable to write to address <address> on device '<device name>'. [STS..... 25

**Index**

## **Allen-Bradley Ethernet Device Driver Help**

---

Help version 1.017

### **CONTENTS**

#### **Overview**

What is the Allen-Bradley Ethernet Device Driver?

#### **Device Setup**

How do I configure a device for use with this driver?

#### **Optimizing Your AB Ethernet Communications**

How do I get the best performance from the Allen-Bradley Ethernet driver?

#### **Data Types Description**

What data types are supported by this driver?

#### **Address Descriptions**

How do I address a data location on a Allen-Bradley Ethernet device?

#### **Error Descriptions**

What error messages are produced by the Allen-Bradley Ethernet driver?

### **Overview**

---

The Allen-Bradley Ethernet Device Driver was designed specifically for use with 32 bit OPC server products running on Intel microprocessor based computers. For operating system (OS) requirements, please refer to the OPC server help documentation.

This driver supports the Allen Bradley SLC5/05 series and PLC5 series PLCs. Address ranges are open to support future models of this series of PLCs.

### **Device Setup**

---

#### **Supported Devices**

SLC5/05 processor\*

PLC5 series excluding the PLC5/250 series

\*Address ranges have been opened up in the driver to allow for future devices. Therefore, the driver may still support a device even if it is not listed above.

#### **Communication Protocol**

Allen-Bradley Ethernet

#### **Timeout Parameters**

Connect timeout: 1 to 60 seconds

Request timeout: 100 to 30000 milliseconds

Attempts: 1 to 10

### **Cable Connections**

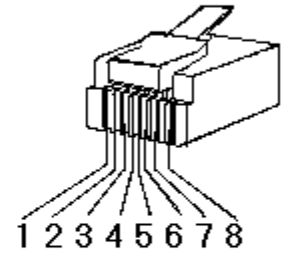
---

## Patch Cable (Straight Through)

TD + 1	OR/WHT	OR/WHT	1	TD +
TD - 2	OR	OR	2	TD -
RD + 3	GRN/WHT	GRN/WHT	3	RD +
4	BLU	BLU	4	
5	BLU/WHT	BLU/WHT	5	
RD - 6	GRN	GRN	6	RD -
7	BRN/WHT	BRN/WHT	7	
8	BRN	BRN	8	

RJ45 RJ45

## 10 BaseT



## Crossover Cable

TD + 1	OR/WHT	GRN/WHT	1	TD +
TD - 2	OR	GRN	2	TD -
RD + 3	GRN/WHT	OR/WHT	3	RD +
4	BLU	BLU	4	
5	BLU/WHT	BLU/WHT	5	
RD - 6	GRN	OR	6	RD -
7	BRN/WHT	BRN/WHT	7	
8	BRN	BRN	8	

RJ45 RJ45

## 8-pin RJ45

**Communications Parameters****Port Number**

This parameter specifies the port number that the remote device is configured to use. The default port number is 2222.

**Protocol**

The AB Ethernet driver connects to the supported devices using Transfer Control Protocol (TCP) and requires Winsock V1.1 or higher.

**Block Request Size**

Request size refers to the number of bytes that may be requested from a device at one time. To refine the driver's performance, configure the request size to one of the following settings: 32, 64, 128, 256, 512, 1024, and 2000 bytes. The default value is 512 bytes. For Boolean arrays, the block size is the bit equivalent (i.e block size multiplied by 8). For example, a block size of 512 bytes is equal to  $512 * 8 = 4096$  bits.

**Protocol Parameters****Destination Node Address (DST)**

For DF1 gateway applications, enter the node address of the destination device. For example: the node address of the DH+ or DH-485 device.

For non-DF1 gateway applications, leave the default node address of 0 (zero).

**SLC500 Slot Configuration****Slot Configuration**

SLC5/05 models (modular I/O racks) need to be configured for use with the AB Ethernet driver if the I/O is to be accessed by the driver. Up to 30 slots can be configured per device. To use the slot configuration dialog, follow the instructions below.

1. Select the slot that will be configured by left-clicking on the row in the slot/module list box.
2. Left-click again to select a module from the available modules list box.
3. Click **Add** to add the module.
4. To remove a module, select the correct slot in the slot/module list box and then click **Remove**.

**Note:** The available module selections are the same as those in the Allen Bradley APS software.

Knowledge of the number of input and output words in each slot is necessary for the driver to correctly address the I/O. Only the number of input and output words in slots (up to the slot of interest) is needed to address I/O in that slot. For example, if only slot 3 will be accessed, users must configure slots 1 and 2 (if they contain any I/O) and slot 3, but not slot 4 or greater.

See Also: [Modular I/O Selection Guide](#)

## Modular I/O Selection Guide

The following table lists the number of input and output words available for each I/O module in the Slot Configuration list.

Module Type	Input Words	Output Words
1746-I*8 Any 8 pt Discrete Input Module	1	0
1746-I*16 Any 16 pt Discrete Input Module	1	0
1746-I*32 Any 32 pt Discrete Input Module	2	0
1746-O*8 Any 8 pt Discrete Output Module	0	1
1746-O*16 Any 16 pt Discrete Output Module	0	1
1746-O*32 Any 32 pt Discrete Output Module	0	2
1746-IA4 4 Input 100/120 VAC	1	0
1746-IA8 8 Input 100/120 VAC	1	0
1746-IA16 16 Input 100/120 VAC	1	0
1746-IB8 8 Input (Sink) 24 VDC	1	0
1746-IB16 16 Input (Sink) 24 VDC	1	0
1746-IB32 32 Input (Sink) 24 VDC	2	0
1746-IG16 16 Input [TTL] (Source) 5VDC	1	0
1746-IM4 4 Input 200/240 VAC	1	0
1746-IM8 8 Input 200/240 VAC	1	0
1746-IM16 16 Input 200/240 VAC	1	0
1746-IN16 16 Input 24 VAC/VDC	1	0
1746-ITB16 16 Input [Fast] (Sink) 24 VDC	1	0
1746-ITV16 16 Input [Fast] (Source) 24 VDC	1	0
1746-IV8 8 Input (Source) 24 VDC	1	0
1746-IV16 16 Input (Source) 24 VDC	1	0
1746-IV32 32 Input (Source) 24 VDC	2	0
1746-OA8 8 Output (Triac) 100/240 VAC	0	1
1746-OA16 16 Output (Triac) 100/240 VAC	0	1
1746-OB8 8 Output [Trans] (Source) 10/50 VDC	0	1
1746-OB16 16 Output [Trans] (Source) 10/50 VDC	0	1
1746-OB32 32 Output [Trans] (Source) 10/50 VDC	0	2
1746-OBP16 16 Output [Trans 1 amp] (SRC) 24 VDC	0	1

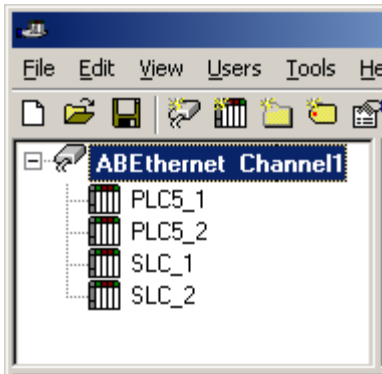
1746-OV8 8 Output [Trans] (Sink) 10/50 VDC	0	1
1746-OV16 16 Output [Trans] (Sink) 10/50 VDC	0	1
1746-OV32 32 Output [Trans] (Sink) 10/50 VDC	0	2
1746-OW4 4 Output [Relay] VAC/VDC	0	1
1746-OW8 8 Output [Relay] VAC/VDC	0	1
1746-OW16 16 Output [Relay] VAC/VDC	0	1
1746-OX8 8 Output [Isolated Relay] VAC/VDC	0	1
1746-OVP16 16 Output [Trans 1 amp] (Sink) 24VDC3	0	1
1746-IO4 2 In 100/120 VAC 2 Out [Rly] VAC/VDC3	1	1
1746-IO8 4 In 100/120 VAC 4 Out [Rly] VAC/VDC4	1	1
1746-IO12 6 In 100/120 VAC 6 Out [Rly] VAC/VDC	1	1
1746-NI4 4 Ch Analog Input	4	0
1746-NIO4I Analog Comb 2 in & 2 Current Out	2	2
1746-NIO4V Analog Comb 2 in & 2 Voltage Out	2	2
1746-NO4I 4 Ch Analog Current Output	0	4
1746-NO4V 4 Ch Analog Voltage Output	0	4
1746-NT4 4 Ch Thermocouple Input Module	8	8
1746-NR4 4 Ch Rtd/Resistance Input Module	8	8
1746-HSCE High Speed Counter/Encoder	8	1
1746-HS Single Axis Motion Controller	4	4
1746-OG16 16 Output [TLL] (SINK) 5 VDC	0	1
1746-BAS Basic Module 500 5/01 Configuration	8	8
1746-BAS Basic Module 5/02 Configuration	8	8
1747-DCM Direct Communication Module (1/4 Rack)	2	2
1747-DCM Direct Communication Module (1/2 Rack)	4	4
1747-DCM Direct Communication Module (3/4Rack)	6	6
1747-DCM Direct Communication Module (Full Rack)	8	8
1747-SN Remote I/O Scanner	32	32
1747-DSN Distributed I/O Scanner 7 Blocks	8	8
1747-DSN Distributed I/O Scanner 30 Blocks	32	32
1747-KE Interface Module, Series A	1	0
1747-KE Interface Module, Series B	8	8
1746-NI8 8 Ch Analog Input, Class 1	8	8
1746-NI8 8 Ch Analog Input, Class 3	16	12
1746-IC16 16 Input (Sink) 48 VDC	1	0
1746-IH16 16 Input [Trans] (Sink) 125 VDC	1	0
1746-OAP12 12 Output [Triac] 120/240 VDC	0	1
1746-OB6EI 6 Output [Trans] (Source) 24 VDC	0	1
1746-OB16E 16 Output [Trans] (Source) Protected	0	1
1746-OB32E 32 Output [Trans] (Source) 10/50 VDC	0	2
1746-OBP8 8 Output [Trans 2 amp] (Source) 24 VDC	0	1
1746-IO12DC 6 Input 12 VDC, 6 Output [Rly]	1	1
1746-INI4I Analog 4 Ch. Isol. Current Input	8	8
1746-INI4VI Analog 4 Ch. Isol. Volt./Current Input	8	8
1746-INT4 4 Ch. Isolated Thermocouple Input	8	8
1746-NT8 Analog 8 Ch Thermocouple Input	8	8
1746-HSRV Motion Control Module	12	8
1746-HSTP1 Stepper Controller Module	8	8

1747-MNET MNET Network Comm Module	0	0
1747-QS Synchronized Axes Module	32	32
1747-QV Open Loop Velocity Control	8	8
1747-RCIF Robot Control Interface Module	32	32
1747-SCNR ControlNet SLC Scanner	32	32
1747-SDN DeviceNet Scanner Module	32	32
1394-SJT GMC Turbo System	32	32
1203-SM1 SCANport Comm Module - Basic	8	8
1203-SM1 SCANport Comm Module - Enhanced	32	32
AMCI-1561 AMCI Series 1561 Resolver Module	8	8

## Optimizing Your AB Ethernet Communications

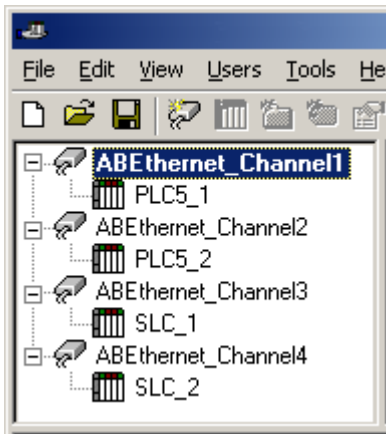
The Allen-Bradley Ethernet driver has been designed to provide the best performance with the least amount of impact on the system's overall performance. While the Allen-Bradley Ethernet driver is fast, there are a couple of guidelines that can be used in order to control and optimize the application and gain maximum performance.

Our server refers to communications protocols like Allen-Bradley Ethernet as a channel. Each channel defined in the application represents a separate path of execution in the server. Once a channel has been defined, a series of devices must then be defined under that channel. Each of these devices represents a single Allen-Bradley PLC from which data will be collected. While this approach to defining the application will provide a high level of performance, it won't take full advantage of the Allen-Bradley Ethernet driver or the network. An example of how the application may appear when configured using a single channel is shown below.



Each device appears under a single Allen-Bradley Ethernet channel. In this configuration, the driver must move from one device to the next as quickly as possible in order to gather information at an effective rate. As more devices are added or more information is requested from a single device, the overall update rate begins to suffer.

If the Allen-Bradley Ethernet driver could only define one single channel, then the example shown above would be the only option available; however, the Allen-Bradley Ethernet driver can define up to 256 channels. Using multiple channels distributes the data collection workload by simultaneously issuing multiple requests to the network. An example of how the same application may appear when configured using multiple channels to improve performance is shown below.



Each device has now been defined under its own channel. In this new configuration, a single path of execution is dedicated to the task of gathering data from each device. If the application has 256 or fewer devices, it can be optimized exactly how it is shown here.

The performance will improve even if the application has more than 256 devices. While 256 or fewer devices may be ideal, the application will still benefit from additional channels. Although by spreading the device load across all channels will cause the server to move from device to device again, it can now do so with far less devices to process on a single channel.

## Data Types Description

Data Type	Description
Boolean	Single bit
Byte	Unsigned 8 bit value
Char	Signed 8 bit value
Word	Unsigned 16 bit value
Short	Signed 16 bit value
DWord	Unsigned 32 bit value
Long	Signed 32 bit value
BCD	Two byte packed BCD, four decimal digits
LBCD	Four byte packed BCD, eight decimal digits
Float	32 bit IEEE Floating point
String	Null terminated character array

The DWord, Long and LBCD data types are not native to any of the PLC models. When referencing a 16 bit location as a 32 bit value, the location referenced will be the low word, and the next successive location will be the high word. For example: If N7:10 is selected as a DWord data type, N7:10 would be the low word and N7:11 the high word.

## Address Descriptions

Address specifications vary depending on the model in use. Select a link from the following list to obtain specific address information for the model of interest.

[SLC5/05 Open Addressing](#)

[PLC-5 Family Addressing](#)

[General Addressing](#)

## General Addressing

These addresses are general addresses that pertain to both the SLC50/5 and PLC-5.

[Output Files](#)

[Input Files](#)

[Status Files](#)

[Binary Files](#)

[Timer Files](#)

[Counter Files](#)

[Control Files](#)

[Integer Files](#)

[Float Files](#)

[ASCII Files](#)

[String Files](#)

See Also: [SLC5/05 Open Addressing](#), [PLC-5 Family Addressing](#)

## Output Files

The syntax for accessing data in the output file differs depending on the PLC model. Arrays are not supported for output files. Data locations are Read/Write for PLC-5 models and Read Only for all other models. The default data type for all syntax is shown in **bold**.

### PLC-5 Model Syntax

Syntax	Data Type
O:<word>	Short, <b>Word</b> , BCD
O:<word>/<bit>	<b>Boolean</b>
O:<word>/<bit>[rows][cols]	<b>Boolean</b> (array types)
O:<word>/<bit>[cols]	<b>Boolean</b> (array types)
O/bit	<b>Boolean</b>
O/bit[rows][cols]	<b>Boolean</b> (array types)
O/bit[cols]	<b>Boolean</b> (array types)

**Note:** Word and bit address information is in octal for PLC-5 models. This follows the convention of the programming software.

### SLC 5/05 Open Models (Modular I/O) Syntax

Syntax	Data Type
O:<slot>	Short, <b>Word</b> , BCD
O:<slot>.<word>	Short, <b>Word</b> , BCD
O:<slot>/<bit>	<b>Boolean</b>
O:<slot>/<bit>[rows][cols]	<b>Boolean</b> (array types)
O:<slot>/<bit>[cols]	<b>Boolean</b> (array types)
O:<slot>.<word>/<bit>	<b>Boolean</b>
O:<slot>.<word>/<bit>[rows][cols]	<b>Boolean</b> (array types)
O:<slot>.<word>/<bit>[cols]	<b>Boolean</b> (array types)

The following slot and word locations are allowed for each model.

PLC Model	Min Slot	Max Slot	Max Word
SLC 5/05 Open	1	30	*
PLC-5 Family	NA	NA	277 (octal)

\*The number of Input or Output words available for each I/O module can be found in the [Modular I/O Selection Guide](#).

For information on slot configuration, refer to [Device Setup](#).

### Examples

PLC-5	Addresses (Octal)
O:0	word 0
O:37	word 31 (37 octal=31 decimal)
O/42	bit 34 (42 octal=34 decimal)

O:2/2	bit 2 word 2 (same as O/42)
O/20[9]	9 element Boolean array starting at bit 16 (20 octal=16 decimal)
O/37[8][11]	8 by 11 element Boolean array starting at bit 31 (37 octal=31 decimal)
O:47/5[3]	3 element Boolean array starting at bit 5 word 39 (47 octal=39 decimal)
O:11/13[3][7]	3 by 7 element Boolean array starting at bit 11 (13 octal=11 decimal) word 9 (11 octal=9 decimal)

SLC 5/05	Addresses (Octal)
O:1	word 0 slot 1
O:1.0	word 0 slot 1 (same as O:1)
O:12	word 0 slot 12
O:12.2	word 2 slot 12
O:4.0/0	bit 0 word 0 slot 4
O:4/0	bit 0 slot 4 (same as O:4.0/0)
O:4.2/0	bit 0 word 2 slot 4
O:4/32	bit 32 slot 4 (same as O:4.2/0)
O:2.12/3[17]	17 element Boolean array starting at bit 3 word 12 slot 2
O:2.2/0[12][12]	12 by 12 element Boolean array starting at bit 0 word 2 slot 2
O:2/43[5]	5 element Boolean array starting at bit 43 slot 2
O:2/11[6][12]	6 by 12 element Boolean array starting at bit 11 slot 2

## Input Files

The syntax for accessing data in the input file differs depending on the PLC model. Arrays are not supported for input files. Data locations are Read/Write for PLC-5 models, Read Only for all other models. The default data type for all syntax is shown in **bold**.

### PLC-5 Model Syntax

Syntax	Data Type
I:<word>	Short, <b>Word</b> , BCD
I:<word>/<bit>	<b>Boolean</b>
I:<word>/<bit>[rows][cols]	<b>Boolean</b> (array types)
I:<word>/<bit>[cols]	<b>Boolean</b> (array types)
I/bit	<b>Boolean</b>
I/bit[rows][cols]	<b>Boolean</b> (array types)
I/bit[cols]	<b>Boolean</b> (array types)

**Note:** Word and bit address information is in octal for PLC-5 models. This follows the convention of the programming software.

### SLC 5/05 Open Models (Modular I/O) Syntax

Syntax	Data Type
I:<slot>	Short, <b>Word</b> , BCD
I:<slot>.<word>	Short, <b>Word</b> , BCD
I:<slot>/<bit>	<b>Boolean</b>
I:<slot>/<bit>[rows][cols]	<b>Boolean</b> (array types)
I:<slot>/<bit>[cols]	<b>Boolean</b> (array types)
I:<slot>.<word>/<bit>	<b>Boolean</b>
I:<slot>.<word>/<bit>[rows][cols]	<b>Boolean</b> (array types)
I:<slot>.<word>/<bit>[cols]	<b>Boolean</b> (array types)

The following slot and word locations are allowed for each model.

PLC Model	Min Slot	Max Slot	Max Word
SLC 5/05 Open	1	30	*
PLC5 Family	NA	NA	277 (octal)

\*The number of Input or Output words available for each I/O module can be found in the [Modular I/O Selection Guide](#).

For information on slot configuration, refer to [Device Setup](#).

### Examples

PLC-5	Addresses (Octal)
I:0	word 0
I:10	word 8 (10 octal = 8 decimal)
I/20	bit 16 (20 octal = 16 decimal)
I:1/0	bit 0 word 1 (same as I/20)
I/20[9]	9 element Boolean array starting at bit 16 (20 octal = 16 decimal)
I/37[8][11]	8 by 11 element Boolean array starting at bit 31 (37 octal = 31 decimal)
I:47/5[3]	3 element Boolean array starting at bit 5 word 39 (47 octal = 39 decimal)
I:11/13[3][7]	3 by 7 element Boolean array starting at bit 11 (13 octal = 11 decimal) word 9 (11 octal = 9 decimal)

SLC 5/05	Addresses (Octal)
I:1	word 0 slot 1
I:1.0	word 0 slot 1 (same as I:1)
I:12	word 0 slot 12
I:12.2	word 2 slot 12
I:4.0/0	bit 0 word 0 slot 4
I:4/0	bit 0 slot 4 (same as I:4.0/0)
I:4.2/0	bit 0 word 2 slot 4
I:4/32	bit 32 slot 4 (same as I:4.2/0)
I:2.12/3[17]	17 element Boolean array starting at bit 3 word 12 slot 2
I:2.2/0[12][12]	12 by 12 element Boolean array starting at bit 0 word 2 slot 2
I:2/43[5]	5 element Boolean array starting at bit 43 slot 2
I:2/11[6][12]	6 by 12 element Boolean array starting at bit 11 slot 2

### Status Files

Specifying a word (and optionally, a bit in the word) accesses status files. The default data type for all syntax is shown in **bold**.

Syntax	Data Type
S:<word>	Short, <b>Word</b> , BCD, DWord, Long, LBCD
S:<word> [rows][cols]	Short, <b>Word</b> , BCD, DWord, Long, LBCD (array types)
S:<word> [cols]	Short, <b>Word</b> , BCD, DWord, Long, LBCD (array types)
S:<word>/<bit>	<b>Boolean</b>
S:<word>/<bit> [rows][cols]	<b>Boolean</b> (array types)
S:<word>/<bit> [cols]	<b>Boolean</b> (array types)
S/bit	<b>Boolean</b>
S/bit [rows][cols]	<b>Boolean</b> (array types)
S/bit [cols]	<b>Boolean</b> (array types)

The number of array elements (in bytes) cannot exceed the [block request size](#) specified. This means that array size cannot exceed 16 words given a block request size of 32 bytes.

The following Word locations are allowed for each model. The maximum word location is one less when accessing as a 32 bit data type (Long, DWord or Long BCD).

PLC Model	Max Word
SLC 5/05 Open	999
PLC-5 Family	999

Example	Description
S:0	word 0
S/26	bit 26
S:4/15	bit 15 word 4
S:10 [16]	16 element array starting at word 10
S:0 [4][8]	4 by 8 element array starting at word 0
S/9 [5]	5 element Boolean array starting at bit 9
S/11 [3][7]	3 by 7 element Boolean array starting at bit 11
S:6/1 [6]	6 element Boolean array starting at bit 1 word 6
S:13/5 [2][3]	2 by 3 element Boolean array starting at bit 5 word 13

## Binary Files

Specifying a file number, a word (and optionally, a bit in the word) accesses binary files. The default data type for all syntax is shown in **bold**.

Syntax	Data Type
B<file>:<word>	Short, <b>Word</b> , BCD, DWord, Long, LBCD
B<file>:<word> [rows][cols]	Short, <b>Word</b> , BCD, DWord, Long, LBCD (array types)
B<file>:<word> [cols]	Short, <b>Word</b> , BCD, DWord, Long, LBCD (array types)
B<file>:<word>/<bit>	<b>Boolean</b>
B<file>:<word>/<bit> [rows][cols]	<b>Boolean</b> (array types)
B<file>:<word>/<bit> [cols]	<b>Boolean</b> (array types)
B<file>/bit	<b>Boolean</b>
B<file>/bit [rows][cols]	<b>Boolean</b> (array types)
B<file>/bit [cols]	<b>Boolean</b> (array types)

The number of array elements (in bytes) cannot exceed the [block request size](#) specified. This means that array size cannot exceed 16 words given a block request size of 32 bytes.

The following file numbers and maximum word locations are allowed for each model. The maximum word location is one less when accessing as a 32 bit data type (Long, DWord or Long BCD).

PLC Model	File Number	Max Word
SLC 5/05 Open	3, 9-999	999
PLC-5 Family	3-999	1999

Example	Description
B3:0	word 0
B3/26	bit 26
B12:4/15	bit 15 word 4
B3:10 [20]	20 element array starting at word 10

B15:0 [6][6]	6 by 6 element array starting at word 0
B3/7 [8]	8 element Boolean array starting at bit 7
B3/32 [6][9]	6 by 9 element Boolean array starting at bit 32
B3:11/2 [12]	12 element Boolean array starting at bit 2 word 11
B3:23/4 [5][8]	5 by 8 element Boolean array starting at bit 4 word 23

## Timer Files

Timer files are a structured type whose data is accessed by specifying a file number, an element and a field. The default data type depends on the field being accessed. Integer fields receive a default data type of Word.

Syntax	Data Type
T<file>:<element>.<field>	Depends on field

The following file numbers and maximum element are allowed for each model.

PLC Model	File Number	Max Element
SLC 5/05 Open	4, 9-999	999
PLC-5 Family	3-999	1999

The following fields are allowed for each element. Refer to the PLC documentation for the meaning of each field.

Element Field	Data Type	Access
ACC	Short, <b>Word</b>	Read/Write
PRE	Short, <b>Word</b>	Read/Write
DN	<b>Boolean</b>	Read Only
TT	<b>Boolean</b>	Read Only
EN	<b>Boolean</b>	Read Only

Example	Description
T4:0.ACC	accumulator of timer 0 file 4
T4:10.DN	done bit of timer 10 file 4
T15:0.PRE	preset of timer 0 file15

## Counter Files

Counter files are a structured type whose data is accessed by specifying a file number, an element and a field. The default data type depends on the field being accessed. Integer fields receive a default data type of Word.

Syntax	Data Type
C<file>:<element>.<field>	Depends on field

The following file numbers and maximum element are allowed for each model.

PLC Model	File Number	Max Element
SLC 5/05 Open	5, 9-999	999
PLC-5 Family	3-999	1999

The following fields are allowed for each element. Refer to the PLC documentation for the meaning of each field.

Element Field	Data Type	Access
ACC	Short, <b>Word</b>	Read/Write
PRE	Short, <b>Word</b>	Read/Write
UA	<b>Boolean</b>	Read Only

UN	<b>Boolean</b>	Read Only
OV	<b>Boolean</b>	Read Only
DN	<b>Boolean</b>	Read Only
CD	<b>Boolean</b>	Read Only
CU	<b>Boolean</b>	Read Only

Example	Description
C5:0.ACC	accumulator of counter 0 file 5
C5:10.DN	done bit of counter 10 file 5
C15:0.PRE	preset of counter 0 file 15

## Control Files

Control files are a structured type whose data is accessed by specifying a file number, an element and a field. The default data type depends on the field being accessed. Integer fields receive a default data type of Word.

Syntax	Data Type
R<file> : <element> . <field>	Depends on field

The following file numbers and maximum element are allowed for each model.

PLC Model	File Number	Max Element
SLC 5/05 Open	6, 9-999	999
PLC-5 Family	3-999	1999

The following fields are allowed for each element. Refer to the PLC documentation for the meaning of each field.

Element Field	Data Type	Access
LEN	Short, <b>Word</b>	Read/Write
POS	Short, <b>Word</b>	Read/Write
FD	<b>Boolean</b>	Read Only
IN	<b>Boolean</b>	Read Only
UL	<b>Boolean</b>	Read Only
ER	<b>Boolean</b>	Read Only
EM	<b>Boolean</b>	Read Only
DN	<b>Boolean</b>	Read Only
EU	<b>Boolean</b>	Read Only
EN	<b>Boolean</b>	Read Only

Examples	Description
R6:0.LEN	length field of control 0 file 6
R6:10.DN	done bit of control 10 file 6
R15:18.POS	position field of control 18 file 15

## Integer Files

Specifying a file number, a word (and optionally, a bit in the word) accesses integer files. The default data type for all syntax is shown in **bold**.

Syntax	Data Type
N<file> : <word>	Short, <b>Word</b> , BCD, DWord, Long, LBCD
N<file> : <word> [rows][cols]	Short, <b>Word</b> , BCD, DWord, Long, LBCD

	(array types)
N<file>:<word> [cols]	Short, <b>Word</b> , BCD, DWord, Long, LBCD (array types)
N<file>:<word>/<bit>	<b>Boolean</b>
N<file>:<word>/<bit> [rows] [cols]	<b>Boolean</b> (array types)
N<file>:<word>/<bit> [cols]	<b>Boolean</b> (array types)
N<file>/bit	<b>Boolean</b>
N<file>/bit [rows][cols]	<b>Boolean</b> (array types)
N<file>/bit [cols]	<b>Boolean</b> (array types)

The number of array elements (in bytes) cannot exceed the [block request size](#) specified. This means that array size cannot exceed 16 words given a block request size of 32 bytes.

The following file numbers and maximum word locations are allowed for each model. The maximum word location is one less when accessing as a 32 bit data type (Long, DWord or Long BCD).

PLC Model	File Number	Max Word
SLC 5/05 Open	7, 9-999	999
PLC-5 Family	3-999	1999

Example	Description
N7:0	word 0
N7/26	bit 26
N12:4/15	bit 15 word 4
N7:10 [8]	8 element array starting at word 10
N15:0 [4][5]	4 by 5 element array starting at word 0
N7/12 [9]	9 element Boolean array starting at bit 12
N7/19 [3][11]	3 by 11 element Boolean array starting at bit 19
N7:7/0 [10]	10 element Boolean array starting at bit 0 word 7
N7:29/13 [2][15]	2 by 15 element Boolean array starting at bit 13 word 29

## Float Files

Specifying a file number and an element accesses float files. The only data type allowed is float.

Syntax	Data Type
F<file>:<element>	Float
F<file>:<element> [rows][cols]	Float array
F<file>:<element> [cols]	Float array

The number of array elements (in bytes) cannot exceed the [block request size](#) specified. This means array size cannot exceed 8 Floats given a block request size of 32 bytes.

The following file numbers and maximum word locations are allowed for each model.

PLC Model	File Number	Max Word
SLC 5/05 Open	8-999	999
PLC-5 Family	3-999	1999

Example	Description
F8:0	Float 0
F8:10 [16]	16 element array starting at word 10

F15:0 [4][4] 16 element array starting at word 0

## ASCII Files

Specifying a file number and character location accesses ASCII file data. The default data type for all syntax is shown in **bold**.

Syntax	Data Type
A<file>:<char>	<b>Char</b> , Byte (1)
A<file>:<char> [rows][cols]	<b>Char</b> , Byte (1)
A<file>:<char> [cols]	<b>Char</b> , Byte (1)
A<file>:<word offset>/length	<b>String</b> (2)

1. The number of array elements cannot exceed the [block request size](#) specified.

Internally, the PLC packs two characters per word in the file, with the high byte containing the first character and the low byte containing the second character. The PLC programming software allows access at the word level or two-character level. The AB Ethernet driver allows accessing to the character level.

Using the programming software A10:0=AB would result in 'A' being stored in the high byte of A10:0 and 'B' being stored in the low byte.

Using the AB Ethernet driver you would make two assignments, A10:0=A and A10:1=B, which would result in the same data being stored in the PLC memory.

2. Referencing this file as string data allows access to data at word boundaries like the programming software. The length can be up to 236 characters. If a string that is sent to the device is smaller in length than the length specified by the address, the driver null terminates the string before sending it down to the controller.

The following file numbers and maximum character locations are allowed for each model.

PLC Model	File Number	Max Character
SLC 5/05 Open	9-999	1999
PLC-5 Family	3-999	1999

All SLC 500 PLCs do not support ASCII file types. For more information, refer to the PLC's documentation.

Example	Description
A9:0	Character 0 (high byte of word 0)
A27:10 [80]	80 character array starting at character 10
A15:0 [4][16]	4 by 16 character array starting at character 0
A62:0/32	32 character string starting at word offset 0

## String Files

Specifying a file number and an element accesses data in a string file. The only data type allowed is string, which are 82 character null terminated arrays. The driver places the null terminator based on the string length returned by the PLC.

Syntax	Data Type
ST<file>:<element>.<field>	String

**Note:** Arrays of strings are not supported.

The following file numbers and maximum word locations are allowed for each model.

PLC Model	File Number	Max Word
SLC 5/05 Open	9-999	999

PLC-5 Family	3-999	999
--------------	-------	-----

Example	Description
ST9:0	string 0
ST18:10	string 10

## SLC5/05 Open Addressing

---

### Open Addressing

The actual number of addresses available is dependent on the model of the PLC. The ranges have been opened up to allow for maximum flexibility with future models. If at runtime the driver finds that an address is not present in the device, the driver will post an error message and remove the tag from its scan list.

**Note:** This model has no specific addressing.

**See Also:** [General Addressing](#)

## PLC-5 Family Addressing

---

### General Addressing

[General Addressing](#)

### Model Specific Addressing

[BCD Files](#)

[PID Files](#)

[Message Files](#)

[Block Transfer Files](#)

## BCD Files

---

Specifying a file number and a word accesses BCD files. The only data types allowed are BCD and long BCD. The default data type is always BCD.

Syntax	Data Type
D<file>:<word>	<b>BCD</b> , LBCD
D<file>:<word> [rows][cols]	<b>BCD</b> , LBCD (array types)
D<file>:<word> [cols]	<b>BCD</b> , LBCD (array types)

The number of array elements (in bytes) cannot exceed the [block request size](#) specified. This means array size cannot exceed 16 BCDs given a block request size of 32 bytes.

The following file numbers and maximum word locations are allowed for each model.

PLC Model	File Number	Max Word
SLC 5/05 Open	NA	NA
PLC-5 Family	3-999	999

Example	Description
D9:0	word 0
D27:10 [16]	16 element array starting at word 10
D15:0 [4][8]	32 element array starting at word 0

## PID Files

---

PID files are a structured type whose data is accessed by specifying a file number, an element and a field. The default

data type depends on the field being accessed. Integer fields receive a default data type of Word.

Syntax	Data Type
PD<file>:<element>.<field>	Depends on field

The following file numbers and maximum element are allowed for each model.

PLC Model	File Number	Max Element
SLC 5/05 Open	NA	NA
PLC-5 Family	3-999	999

The following fields are allowed for each element. Refer to the PLC documentation for the meaning of each field.

Element Field	Data Type	Access
SP	Real	Read/Write
KP	Real	Read/Write
KI	Real	Read/Write
KD	Real	Read/Write
BIAS	Real	Read/Write
MAXS	Real	Read/Write
MINS	Real	Read/Write
DB	Real	Read/Write
SO	Real	Read/Write
MAXO	Real	Read/Write
MINO	Real	Read/Write
UPD	Real	Read/Write
PV	Real	Read/Write
ERR	Real	Read/Write
OUT	Real	Read/Write
PVH	Real	Read/Write
PVL	Real	Read/Write
DVP	Real	Read/Write
DVN	Real	Read/Write
PVDB	Real	Read/Write
DVDB	Real	Read/Write
MAXI	Real	Read/Write
MINI	Real	Read/Write
TIE	Real	Read/Write
FILE	Short, <b>Word</b>	Read/Write
ELEM	Short, <b>Word</b>	Read/Write
EN	<b>Boolean</b>	Read/Write
CT	<b>Boolean</b>	Read/Write
CL	<b>Boolean</b>	Read/Write
PVT	<b>Boolean</b>	Read/Write
DO	<b>Boolean</b>	Read/Write
SWM	<b>Boolean</b>	Read/Write
CA	<b>Boolean</b>	Read/Write
MO	<b>Boolean</b>	Read/Write
PE	<b>Boolean</b>	Read/Write
INI	<b>Boolean</b>	Read/Write

SPOR	<b>Boolean</b>	Read/Write
OLL	<b>Boolean</b>	Read/Write
OLH	<b>Boolean</b>	Read/Write
EWD	<b>Boolean</b>	Read/Write
DVNA	<b>Boolean</b>	Read/Write
DVHA	<b>Boolean</b>	Read/Write
PVLA	<b>Boolean</b>	Read/Write
PVHA	<b>Boolean</b>	Read/Write

<b>Example</b>	<b>Description</b>
PD14:0.SP	setpoint field of PD 0 file 14
PD18:6.EN	status enable bit of PD 6 file 18

## Message Files

Message files are a structured type whose data is accessed by specifying a file number, an element and a field. The default data type depends on the field being accessed. Integer fields receive a default data type of Word.

<b>Syntax</b>	<b>Data Type</b>
MG<file>:<element>.<field>	Depends on field

The following file numbers and maximum element are allowed for each model.

<b>PLC Model</b>	<b>File Number</b>	<b>Max Element</b>
SLC 5/05 Open	NA	NA
PLC-5 Family	3-999	999

The following fields are allowed for each element. Refer to the PLC documentation for the meaning of each field.

<b>Element Field</b>	<b>Data Type</b>	<b>Access</b>
ERR	Short, <b>Word</b>	Read/Write
RLEN	Short, <b>Word</b>	Read/Write
DLEN	Short, <b>Word</b>	Read/Write
EN	<b>Boolean</b>	Read/Write
ST	<b>Boolean</b>	Read/Write
DN	<b>Boolean</b>	Read/Write
ER	<b>Boolean</b>	Read/Write
CO	<b>Boolean</b>	Read/Write
EW	<b>Boolean</b>	Read/Write
NR	<b>Boolean</b>	Read/Write
TO	<b>Boolean</b>	Read/Write

<b>Example</b>	<b>Description</b>
MG14:0.RLEN	requested length field of MG 0 file 14
MG18:6.CO	continue bit of MG 6 file 18

## Block Transfer Files

Block transfer files are a structured type whose data is accessed by specifying a file number, an element and a field. The default data type depends on the field being accessed. Integer fields receive a default data type of Word.

<b>Syntax</b>	<b>Data Type</b>
---------------	------------------

BT<file>:<element>.<field>      Depends on field

The following file numbers and maximum element are allowed for each model.

PLC Model	File Number	Max Element
SLC 5/05 Open	NA	NA
PLC-5 Family	3-999	1999

The following fields are allowed for each element. Refer to the PLC documentation for the meaning of each field.

Element Field	Data Type	Access
RLEN	Short, <b>Word</b>	Read/Write
DLEN	Short, <b>Word</b>	Read/Write
FILE	Short, <b>Word</b>	Read/Write
ELEM	Short, <b>Word</b>	Read/Write
RW	<b>Boolean</b>	Read/Write
ST	<b>Boolean</b>	Read/Write
DN	<b>Boolean</b>	Read/Write
ER	<b>Boolean</b>	Read/Write
CO	<b>Boolean</b>	Read/Write
EW	<b>Boolean</b>	Read/Write
NR	<b>Boolean</b>	Read/Write
TO	<b>Boolean</b>	Read/Write

Example	Description
BT14:0.RLEN	requested length field of BT 0 file 14
BT18:6.CO	continue bit of BT 6 file 18

## Error Descriptions

The following error/warning messages may be generated. Click on the link for a description of the message.

### Address Validation

#### [Missing address](#)

[Device address '<address>' contains a syntax error](#)

[Address '<address>' is out of range for the specified device or register](#)

[Device address '<address>' is not supported by model '<model name>'](#)

[Data Type '<type>' is not valid for device address '<address>'](#)

[Device address '<address>' is Read Only](#)

[Array size is out of range for address '<address>'](#)

[Array support is not available for the specified address: '<address>'](#)

### Communications Error Messages

[Winsock initialization failed \(OS Error = n\)](#)

[Winsock V1.1 or higher must be installed to use the Allen-Bradley Ethernet device driver](#)

### Device Status Messages

[Device '<device name>' is not responding](#)

[Unable to write to '<address>' on device '<device name>'](#)

### Device Specific Messages

[Unable to read data starting at address <address> on device '<device name>'. Frame received contains errors](#)

[Unable to read data starting at address <address> on device '<device name>'. \[STS=<value>, EXT STS=<value>\]](#)

[Unable to write to address <address> on device '<device name>'. Frame received contains errors](#)  
[Unable to write to address <address> on device '<device name>'. \[STS=<value>, EXT STS=<value>\]](#)

## **Address Validation**

---

The following error/warning messages may be generated. Click on the link for a description of the message.

### **Address Validation**

#### **[Missing address](#)**

[Device address '<address>' contains a syntax error](#)

[Address '<address>' is out of range for the specified device or register](#)

[Device address '<address>' is not supported by model '<model name>'](#)

[Data Type '<type>' is not valid for device address '<address>'](#)

[Device address '<address>' is Read Only](#)

[Array size is out of range for address '<address>'](#)

[Array support is not available for the specified address: '<address>'](#)

### **Missing address**

---

#### **Error Type:**

Warning

#### **Possible Cause:**

A tag address that has been specified statically has no length.

#### **Solution:**

Re-enter the address in the client application.

### **Device address '<address>' contains a syntax error**

---

#### **Error Type:**

Warning

#### **Possible Cause:**

A tag address that has been specified statically contains one or more invalid characters.

#### **Solution:**

Re-enter the address in the client application.

### **Address '<address>' is out of range for the specified device or register**

---

#### **Error Type:**

Warning

#### **Possible Cause:**

A tag address that has been specified statically references a location that is beyond the range of supported locations for the device.

#### **Solution:**

Verify the address is correct; if it is not, re-enter it in the client application.

### **Device address '<address>' is not supported by model '<model name>'**

---

#### **Error Type:**

Warning

**Possible Cause:**

A tag address that has been specified statically references a location that is valid for the communications protocol but not supported by the target device.

**Solution:**

Verify the address is correct; if it is not, re-enter it in the client application. Also verify that the selected model name for the device is correct.

---

**Data Type '<type>' is not valid for device address '<address>'**

---

**Error Type:**

Warning

**Possible Cause:**

A tag address that has been specified statically has been assigned an invalid data type.

**Solution:**

Modify the requested data type in the client application.

---

**Device address '<address>' is Read Only**

---

**Error Type:**

Warning

**Possible Cause:**

A tag address that has been specified statically has a requested access mode that is not compatible with what the device supports for that address.

**Solution:**

Change the access mode in the client application.

---

**Array size is out of range for address '<address>'**

---

**Error Type:**

Warning

**Possible Cause:**

A tag address that has been specified statically is requesting an array size that is too large for the address type or block size of the driver.

**Solution:**

Re-enter the address in the client application to specify a smaller value for the array or a different starting point.

---

**Array support is not available for the specified address: '<address>'**

---

**Error Type:**

Warning

**Possible Cause:**

A tag address that has been specified statically contains an array reference for an address type that doesn't support arrays.

**Solution:**

Re-enter the address in the client application to remove the array reference or correct the address type.

## **Communications Error Messages**

---

The following error/warning messages may be generated. Click on the link for a description of the message.

### **Communications Error Messages**

[Winsock initialization failed \(OS Error = n\)](#)

[Winsock V1.1 or higher must be installed to use the Allen-Bradley Ethernet device driver](#)

### **Winsock initialization failed (OS Error = n)**

---

#### **Error Type:**

Fatal

<b>OS Error:</b>	<b>Indication</b>	<b>Possible Solution</b>
10091	Indicates that the underlying network subsystem is not ready for network communication.	Wait a few seconds and restart the driver.
10067	Limit on the number of tasks supported by the Windows Sockets implementation has been reached.	Close one or more applications that may be using Winsock and restart the driver.

### **Winsock V1.1 or higher must be installed to use the Allen-Bradley Ethernet device driver**

---

#### **Error Type:**

Fatal

#### **Possible Cause:**

The version number of the Winsock DLL found on your system is less than 1.1.

#### **Solution:**

Upgrade your version of Winsock to 1.1 or higher.

## **Device Status Messages**

---

The following error/warning messages may be generated. Click on the link for a description of the message.

### **Device Status Messages**

[Device '<device name>' is not responding](#)

[Unable to write to '<address>' on device '<device name>'](#)

### **Device '<device name>' is not responding**

---

#### **Error Type:**

Warning

#### **Possible Cause:**

1. The Ethernet connection between the device and the host PC is broken.
2. The communications parameters for the Ethernet connection are incorrect.
3. The named device may have been assigned an incorrect IP address.

#### **Solution:**

1. Verify the cabling between the PC and the device.
2. Verify the correct port is specified for the named device.
3. Verify the IP address given to the named device matches that of the actual device.

---

**Unable to write to '<address>' on device '<device name>'**

---

**Error Type:**

Warning

**Possible Cause:**

1. The Ethernet connection between the device and the host PC is broken.
2. The communication parameters for the Ethernet connection are incorrect.
3. The named device may have been assigned an incorrect IP address.

**Solution:**

1. Verify the cabling between the PC and the device.
2. Verify that the correct port has been specified for the named device.
3. Verify that the IP address given to the named device matches that of the actual device.

---

**Device Specific Messages**

---

The following error/warning messages may be generated. Click on the link for a description of the message.

**Device Specific Messages**

[Unable to read data starting at address <address> on device '<device name>'. Frame received contains errors](#)

[Unable to read data starting at address <address> on device '<device name>'. \[STS=<value>, EXT STS=<value>\]](#)

[Unable to write to address <address> on device '<device name>'. Frame received contains errors](#)

[Unable to write to address <address> on device '<device name>'. \[STS=<value>, EXT STS=<value>\]](#)

---

**Unable to read data starting at address <address> on device '<device name>'. Frame received contains errors**

---

**Error Type:**

Warning

**The type of error could be:**

1. An incorrect frame size received.
2. A TNS mismatch.
3. An invalid response command returned from the device.

**Possible Cause:**

1. Misalignment of packets due to connection/disconnection between the PC and device.
2. Bad cabling connecting the devices that is causing noise.

**Solution:**

The driver will recover from the error without intervention. If this error occurs frequently, there may be an issue with the cabling or the device itself.

---

**Unable to read data starting at address <address> on device '<device name>'. [STS=<value>, EXT STS=<value>]**

---

**Error Type:**

Warning

**Possible Cause:**

1. The address requested does not exist in the PLC.
2. The address requested cannot be accessed because the PLC is in an error state.
3. The communications parameters for the Ethernet connection are incorrect.

**Solution:**

1. Verify the address exists in the PLC.

2. Verify the PLC is not in an error state.
3. Verify the communications parameters for the Ethernet connection are correct.
4. Verify the correct port is specified for the named device.
5. Verify the IP address given to the named device matches that of the actual device.

**Note:**

Check the status and extended status codes that are being returned by the PLC. The extended status code may not always be returned; thus, error information is contained within the status code. The codes are displayed in hexadecimal.

Status code errors in the low nibble of the status code indicate errors found by the local node. The driver will continue to retry reading these blocks of data periodically. Errors found by the local node occur when the KF module cannot see the destination PLC on the network for some reason.

Status code errors in the high nibble of the status code indicate errors found by the PLC. These errors are generated when the block of data the driver is asking for is not available in the PLC. The driver will not ask for these blocks again after receiving this kind of error. This kind of error can be generated if the address does not exist in the PLC.

---

**Unable to write to address <address> on device '<device name>'. Frame received contains errors**

---

**Error Type:**

Warning

**The Error Could Be:**

1. Incorrect frame size received.
2. TNS mismatch.
3. Invalid response command returned from device.

**Possible Cause:**

1. Misalignment of packets due to connection/disconnection between PC and device.
2. There is bad cabling connecting the devices causing noise.

**Solution:**

The driver will recover from this error without intervention. If this error occurs frequently, there may be an issue with the cabling or the device itself.

---

**Unable to write to address <address> on device '<device name>'. [STS=<value>, EXT STS=<value>]**

---

**Error Type:**

Warning

**Possible Cause:**

The address written to does not exist in the PLC.

**Solution:**

Check the status and extended status codes that are being returned by the PLC. Note that an extended status code may not always be returned and thus the error information is contained within the status code. The codes are displayed in hexadecimal.

Status code errors in the low nibble of the status code indicate errors found by the local node. Errors found by the local node occur when the KF module cannot see the destination PLC on the network for some reason.

Status code errors in the high nibble of the status code indicate errors found by the PLC. These errors are generated when the data location is not available in the PLC or is not writeable.

# Index

## - A -

Address '<address>' is out of range for the specified device or register. 21

Address Descriptions 8

ASCII Files 16

## - B -

BCD 8

BCD Files 17

Binary Files 12

Block Transfer Files 19

Boolean 8

Byte 8

## - C -

Cable Connections 3

Char 8

Communications Parameters 4

Control Files 14

Counter Files 13

## - D -

Data Types Description 8

Destination Node Address 4

Device '<device name>' is not responding. 23

Device address '<address>' contains a syntax error. 21

Device address '<address>' is not supported by model '<model name>'. 21

Device ID 3

DWord 8

## - F -

Float 8

Float Files 15

## - G -

General Addressing 8

## - I -

Input Files 10

Integer Files 14

## - L -

LBCD 8

Long 8

## - M -

Message 19

Modular I/O Selection Guide 5

## - N -

Network 3

## - O -

Optimizing Your AB Ethernet Communications 7

Output Files 9

Overview 3

## - P -

PID Files 17

PLC5 Addressing 17

Protocol Parameters 4

## - S -

Short 8

SLC5/05 17

SLC500 Slot Configuration 4

Status Files 11

String 8

String Files 16

**- T -**

Timer Files 13

**- U -**

Unable to read data starting at address <address>  
on device '<device name>'. Frame received  
contains errors. 24

Unable to write tag <tag address>... 25

Unable to write to address <address> on device  
'<device name>'. Frame received contains errors.  
25

**- W -**

Winsock initialization failed (OS Error = n) 23

Winsock V1.1 or higher must be installed to use  
the Allen-Bradley Ethernet device driver. 23

Word 8