

Barcode Reader Communication Driver

This document has the specific information related to the driver configuration. For a generic explanation on Devices, Channels, Nodes and Points configuration, please refer to reference guide.

Contents

- Section 1 – Summary Information..... 2
- Section 2 – Channel Configuration 2
 - Protocol Options..... 2
 - Settings 2
- Section 3 – Node Configuration 3
- Section 4 – Point Configuration 3
- Section 5 – Troubleshoot 3
- Revision History..... 4

Section 1 – Summary Information

Communication Driver Name: Barcode

Implementation DLL: T.ProtocolDriver. Barcode.dll

Protocol: ASCII (American Standard Code for Information Interchange)

Interface: TCP/IP and Serial

Description: Barcode is able to receive messages from any equipment, since the message is ASCII encoded. It operates as a slave over Serial or TCP/IP networks.

Equipments types supported: Any equipment compatible with ASCII encoding

Protocol Options: BlockSize, Start Char and End Char

Multi-threading: only one instance for channel is allowed

Max number of nodes: only one Node should be used for each channel

PC Hardware requirements: Standard PC Ethernet interface board, RS485 or RS232 port

Section 2 – Channel Configuration

Protocol Options

BlockSize: Defines the maximum length of a single item, the default value is **256**.

StartChar: Determines the character that indicates the message start. When empty, starts receiving the message with any character.

EndChar: Determines the character that indicates the message end. When empty, stops receiving the message when it reaches the block size length.

Settings

Serial and MultiSerial channels:

- Consult the Serial Port configuration from your equipment

TCP/IP channels:

- **ListeningPort:** Defines the TCP/IP port that will be listening for the connections.

Section 3 – Node Configuration

The PrimaryStation and the BackupStation should be blank

Section 4 – Point Configuration

The syntax for the Barcode communication points is: <Number of Bytes>

Number of Bytes: means how many bytes (characters) the Tag will receive.

The value 0 (zero) means that the Tag will receive the whole message

Example: The message received was “ABCDXYZ”

Case1:

TagName	Address	Value
TagA	4	ABCD
TagB	3	XYZ

Case2:

TagName	Address	Value
TagC	0	ABCDXYZ

Section 5 – Troubleshoot

The status of the driver execution can be observed through the diagnostic tools:

- Trace window
- Property Watch
- Module Information

The above tools indicate if the operations have succeeded or have failed where the status 0 (zero) means success. Negative values are internal error codes and positive values are protocol error codes.

Revision History

Revision	Description	Date
A	Initial Revision	July, 5 th 2012
B	Accept empty StartChar or empty EndChar	July, 12 th 2012