

ASCII 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
Station Configuration	3
Section 4 – Point Configuration	4
Section 5 – Access Point.....	4
Section 6 – Example	4
Section 7 – Troubleshoot	5
Revision History	5

Section 1 – Summary Information

Communication Driver Name: ASCII

Implementation DLL: T.ProtocolDriver.ASCII.dll

Protocol: Generic ASCII

Interface: TCP/IP and Serial

Description: ASCII driver implements communication with any device using ASCII protocol on TCP/IP or serial networks. The communications blocks are dynamically created according the pooling cycle defined on the AccessType for each Device Point.

PLC types supported: Any PLC compatible with ASCII Protocol

Section 2 – Channel Configuration

Protocol Options

BlockSize: Defines the maximum amount of characters, the default value is **250**.

StartChar: Defines the start character of coming message.

EndChar: Defines the end character of coming message.

Settings

Serial and MultiSerial channels:

Default configuration:

BaudRate: 9600

DataBits: 8

StopBits: 1

Parity: None

TCP/IP channels:

- **NodeConnections:** Defines the maximum number of parallel requests that will be sent to each node (asynchronous communication)

Section 3 – Node Configuration

Station Configuration

Serial channels:

- Nothing

TCP/IP channels:

- Station syntax: <IP address> ; <Port number>

Where : <IP address> = IP address of the slave device in the ASCII network

< Port number > = TCP port where the slave device is listening (default is 502)

Ex: 192.168.1.101 ; 502

Section 4 – Point Configuration

The syntax for the ASCII communication points is: <SequenceId>:<SizeOf>

Where:

< SequenceId > indicates the data order where the TX message will be created:

<SizeOf> indicates the characters amount from the configured tag

Example:

Section 5 – Access Point

- Read and Write commands will generate the same TX message.
- **In coming messages from device will be treated as an Unsolicited Message.**

Section 6 – Example

Point 1

TagName: TagA (value equal ABCD)

Address: 0:3

Point 2

TagName: TagB (value equal 1234)

Address: 2:3

Point 3

TagName: TagC (value equal abcd)

Address: 1:4

The TX message created is: ABCabcd123

Section 7 – Troubleshoot

The status of the driver execution can be observed through the Factory Studio diagnostic tools, that are:

- 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	August, 2015
B	Added the Start and End character	September, 2016