

**DNP Communication Driver**

Driver for Serial and Ethernet Communication  
with devices using DNP3 protocol

**Contents**

**CONTENTS .....1**

**INTRODUCTION .....2**

**GENERAL INFORMATION.....3**

    DEVICE SPECIFICATIONS.....3

    NETWORK SPECIFICATIONS.....3

    DRIVER CHARACTERISTICS .....3

    CONFORMANCE TESTING .....3

**SELECTING THE DRIVER .....6**

**CONFIGURING THE DRIVER .....7**

    CONFIGURING THE COMMUNICATION SETTINGS .....7

    CONFIGURING THE DRIVER WORKSHEETS .....8

**EXECUTING THE DRIVER ..... 18**

**TROUBLESHOOTING ..... 20**

**SAMPLE APPLICATION ..... 24**

**REVISION HISTORY..... 25**

## Introduction

This document will help you to select, configure and execute the DNP driver, and it is organized as follows:

- **Introduction:** This section, which provides an overview of the document.
- **General Information:** Identifies all of the hardware and software components required to implement communication between the Studio system and the target device.
- **Selecting the Driver:** Explains how to select the DNP driver in the Studio system.
- **Configuring the Driver:** Explains how to configure the DNP driver in the Studio system, including how to associate database tags with device registers.
- **Executing the Driver:** Explains how to execute the DNP driver during application runtime.
- **Troubleshooting:** Lists the most common errors for this driver, their probable causes, and basic procedures to resolve them.
- **Sample Application:** Explains how to use a sample application to test the DNP driver configuration
- **Revision History:** Provides a log of all changes made to the driver and this documentation.

### **Notes:**

- This document assumes that you have read the “Development Environment” chapter in Studio’s *Technical Reference Manual*.
- This document also assumes that you are familiar with the Microsoft Windows 7/XP/Vista environment. If you are not familiar with Windows, then we suggest using the **Help** feature (available from the Windows desktop **Start** menu) as you work through this guide.

## **General Information**

This chapter identifies all of the hardware and software components required to implement communication between the DNP driver in Studio and remote devices.

The information is organized into the following sections:

- Device Specifications
- Network Specifications
- Driver Characteristics
- Conformance Testing

### **Device Specifications**

You can use this driver to communicate with any device using the DNP3 protocol. (The devices used for conformance testing are listed on the next page.)

### **Network Specifications**

To establish communication, your device network must meet the following specifications:

- **Device Communication Port:** Serial/Ethernet
- **Physical Protocol:** Serial/UDP/TCP/IP
- **Logic Protocol:** DNP3
- **Device Runtime Software:** None
- **Specific PC Board:** None
- **Adapters/Converters:** None
- **Cable Wiring Scheme:** None

### **Driver Characteristics**

The DNP driver package consists of the following files, which are automatically installed in the `\DRV` subdirectory of Studio:

- **DNP.INI:** Internal driver file. *You must not modify this file.*
- **DNP.MSG:** Internal driver file containing error messages for each error code. *You must not modify this file.*
- **DNP.PDF:** This document, which provides detailed information about the DNP driver.
- **DNP.DLL:** Compiled driver.

You can use the DNP driver on the following operating systems:

- Windows 7/XP/Vista
- Windows CE 4.x, 5.x, 6.x

For a description of the operating systems used to test driver conformance, see “Conformance Testing” below.

### **Conformance Testing**

The following hardware/software was used for conformance testing:

For Ethernet Tests

- **TCP/IP Port:** 20000
- **Protocol:** DNP3 over TCP/IP – Serial encapsulation enabled
- **Station:** Device Address
- **Cable:** Ethernet Cable

Driver Version	Studio Version	Operating System	Equipment
1.3	8.0	Windows 8.1	

The DNP driver supports the following services:

- **Read request**
- **Data polling**
- **Unsolicited responses**
- **Write**
- **Select before Operate**
- **Direct Operate**
- **Freeze and Freeze and Clear**
- **Enable/Disable Unsolicited Responses**
- **Cold/Warm restart**

The following groups and variations are supported:

Group	Variation	Group Description
0	242, 243, 246, 248, 250, 252, 255	Device Attributes
1	1, 2	Binary Inputs
2	1 – 3	
3	1, 2	
4	1 – 3	
10	1, 2	Binary Outputs
11	1, 2	
12	1	
13	1, 2	
20	1, 2, 5, 6	Counters
21	1, 2, 5, 6, 9, 10	
22	1, 2, 5, 6	
23	1, 2, 5, 6	
30	1 – 6	Analog Inputs
31	1 – 8	
32	1 – 8	
33	1 – 8	
34	1 – 3	Analog Outputs
40	1 – 4	
41	1 – 4	
42	1 – 8	
43	1 – 8	Time Objects
51	1, 2	
52	1, 2	Class poll
60	1, 2, 3, 4	
80	1	Internal Indications

For most requests issued by the driver, the qualifier 0x01 (2-byte start and stop indexes) is used. However, special qualifiers of the DNP3 protocol are used on some services and groups:

Group	Service	Qualifier
12	Select, Operate, Direct Operate	0x28: 2-byte count of objects, prefixed with a 2-byte index
41		
13	Read	0x06: All objects
60	Enable/Disable Unsolicited Response and Poll	

The following qualifiers are supported on responses:

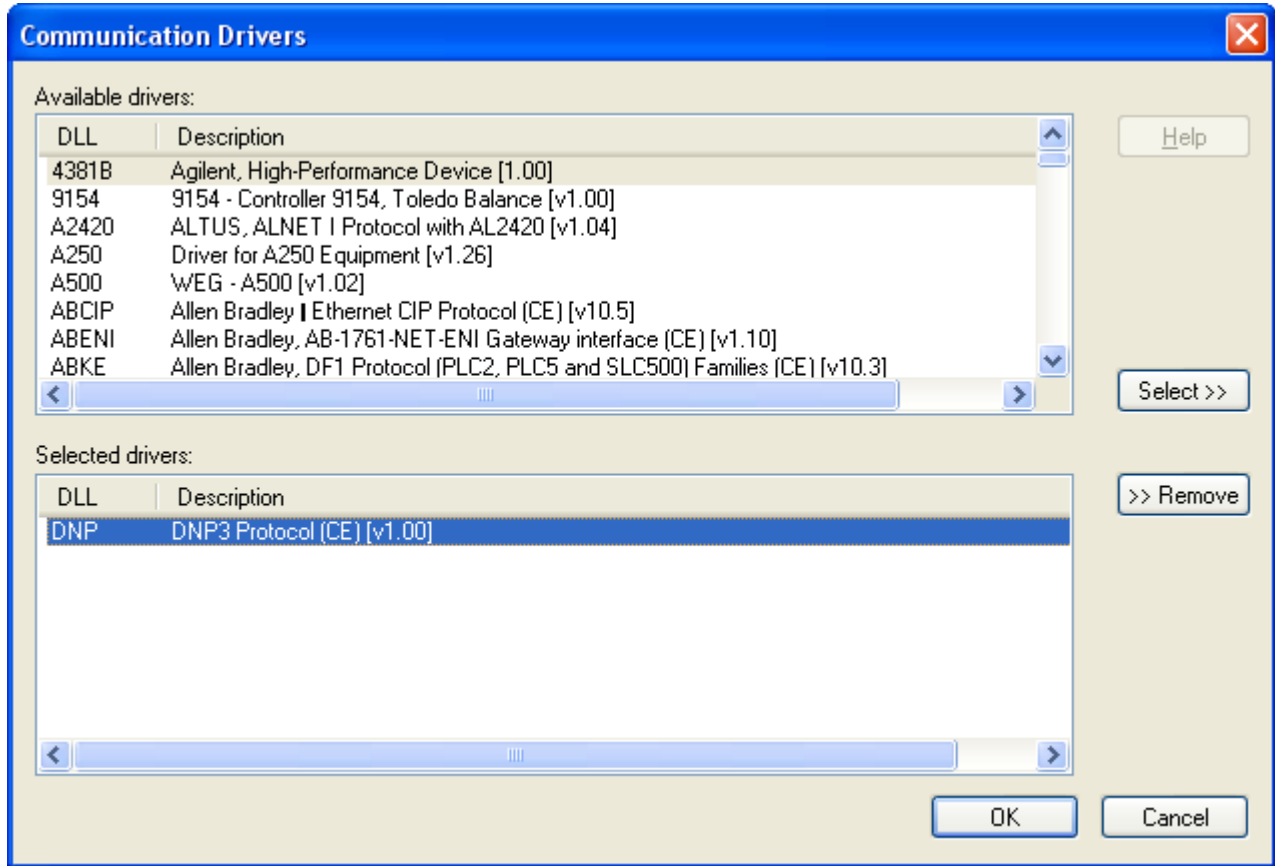
Qualifier	Description
0x00	1-byte start and stop indexes
0x01	2-byte start and stop indexes
0x07	1-byte count of object
0x08	2-byte count of object
0x17	1-byte count of objects, prefixed with a 1-byte index
0x28	2-byte count of objects, prefixed with a 2-byte index
0x5B	1-byte count of objects, prefixed with 2-byte size

The DNP driver operates exclusively as a master and will not answer to messages sent to it from another master.

## Selecting the Driver

When you install Studio, all of the communication drivers are automatically installed in the `\DRV` subdirectory but they remain dormant until manually selected for specific applications. To select the DNP driver for your Studio application:

1. From the main menu bar, select **Insert** → **Driver** to open the *Communication Drivers* dialog.
2. Select the **DNP** driver from the *Available Drivers* list, and then click the **Select** button.



*Communication Drivers Dialog*

3. When the **DNP** driver is displayed in the **Selected Drivers** list, click the **OK** button to close the dialog. The driver is added to the *Drivers* folder, in the *Comm* tab of the Workspace.

**Attention:**  
For safety reasons, you must take special precautions when installing any physical hardware. Please consult the manufacturer's documentation for specific instructions.

## Configuring the Driver

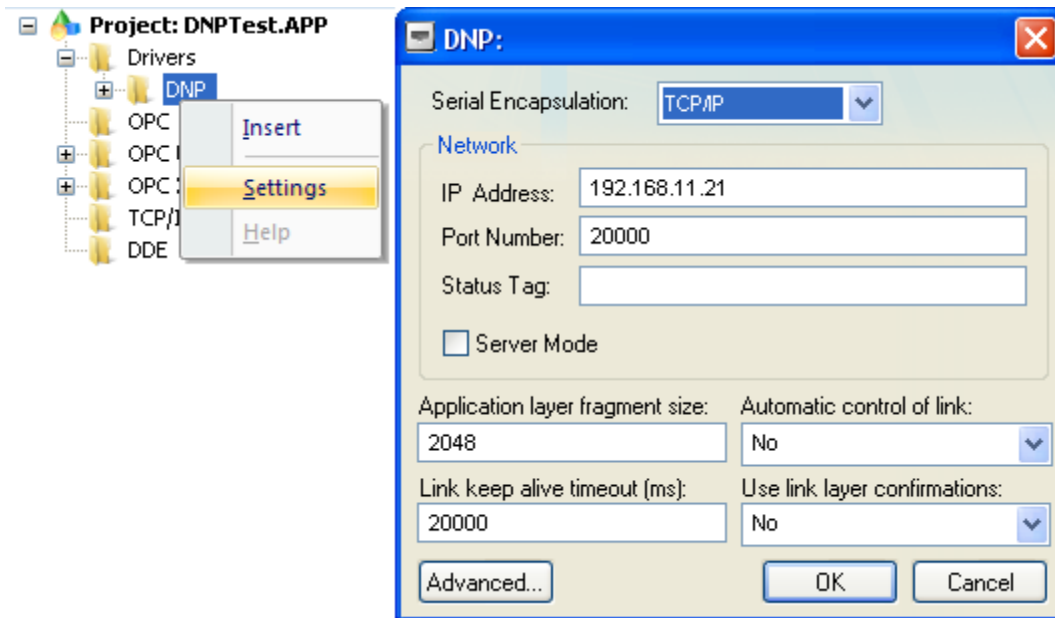
Once you have selected the DNP driver in Studio, you must properly configure it to communicate with your target device.

### Configuring the Communication Settings

The communication settings are described in detail in the “Communication” chapter of the Studio *Technical Reference Manual*, and the same general procedures are used for all drivers. Please review those procedures before continuing.

For the purposes of this document, only DNP driver-specific settings and procedures will be discussed here. To configure the communication settings for the DNP driver:

1. In the *Workspace* pane, select the *Comm* tab and then expand the *Drivers* folder. The DNP driver is listed here as a subfolder.
2. Right-click on the *DNP* subfolder and then select the **Settings** option from the pop-up menu:



**DNP: Communication Settings Dialog**

3. In the *Communication Settings* dialog, configure the driver settings to enable communication with your target device. To ensure error-free communication, the driver settings must *exactly match* the corresponding settings on the device. Please consult the manufacturer’s documentation for instructions how to configure the device and for complete descriptions of the settings.

Depending on your circumstances, you may need to configure the driver *before* you have configured your target device. If this is the case, then take note of the driver settings and have them ready when you later configure the device.

**Attention:**  
For safety reasons, you **must** take special precautions when connecting and configuring new equipment. Please consult the manufacturer’s documentation for specific instructions.

The communication settings and their possible values are described in the following table:

Parameters	Default Value	Valid Values	Description
Application layer fragment size	2048	Integer	This setting defines the maximum expected size, in bytes, of application layer fragments. If a fragment is received with a size longer than configured, an error is returned. Fragments sent by the driver are all limited on 249 bytes.
Link keep alive timeout (ms)	20000	Integer	This setting affect the time that must elapse before the driver sends a test frame, to check if connection is alive. This is used only if Automatic control of link is on.
Automatic control of link	No	Yes/No	This setting defines if the driver will control the link layer automatically, sending the keep alive messages and automatically resetting it. If the driver is under TCP/IP the recommended choice is "No".
Use link layer confirmations	No	Yes/No	This setting defines if the driver must request data link layer confirmations. If the driver is under TCP/IP the recommended choice is "No". Note: Application layer confirmations are never requested.
Station	0	Integer	The address of the master on the DNP3 network. Messages sent to this master must match this address.
Retries	0	Integer	The number of retries allowed to the driver, both on operations triggered by the user and by internal operations, such as reset of link and confirmations of data link layer.
Serial Encapsulation	None	None or TCP/IP	Select the encapsulation to use. To communicate via serial, select None. To use TCP/IP, select TCP/IP. Other encapsulations are not supported for this driver.

### **Configuring the Driver Worksheets**

This driver currently does not support Main Driver Sheet. Standard Driver Worksheets must be inserted to define tag/register associations to be monitored, that are triggered by specific application behaviors.

The configuration of these worksheets is described in detail in the "Communication" chapter of the Studio *Technical Reference Manual*, and the same general procedures are used for all drivers. Please review those procedures before continuing.

For the purposes of this document, only DNP driver-specific parameters and procedures will be discussed here.

### **STANDARD DRIVER WORKSHEET**

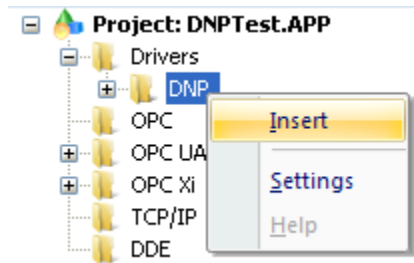
When you select the DNP driver and add it to your application, it does not have any Driver Sheet added. To start communicating, you must insert Standard Driver Worksheets to define the tags/registers to be monitored and commands to be written. These services are specified by the header used on the driver sheet and the addresses.



The configuration of these worksheets is described in detail in the “Communication” chapter of the Studio *Technical Reference Manual*, and the same general procedures are used for all drivers. Please review those procedures before continuing.

To insert a new driver worksheet:

1. In the *Comm* tab, open the *Drivers* folder and locate the *DNP* subfolder.
2. Right-click on the *DNP* subfolder, and then select **Insert** from the pop-up menu:



***Inserting a New Worksheet***

A new *DNP* driver worksheet is inserted into the *DNP* subfolder, and the worksheet is opened for configuration:

**Header**

Description: Internal Indications  Increase priority

Read Trigger:  Enable Read when Idle:  Read Completed:  Read Status:

Write Trigger:  Enable Write on Tag Change: 1 Write Completed:  Write Status:

Station: {station} Header: IIN  Min:   
 Max:

**Body**

	Tag Name	Address	Div	Add
1	IIN[1]	1.0		
2	IIN[2]	1.1		
3	IIN[3]	1.2		
4	IIN[4]	1.3		
5	IIN[5]	1.4		
6	IIN[6]	1.5		
7	IIN[7]	1.6		

**DNP Driver Worksheet**

**Note:**  
 Worksheets are numbered in order of creation, so the first worksheet is **DNP001.drv**.

Most of the fields on this worksheet are standard for all drivers; see the “Communication” chapter of the *Technical Reference Manual* for more information on configuring these fields. However **Header** and **Body** (as noted on the above picture) fields use syntax that is specific to the DNP driver.

3. Configure the **Header** fields as follows:

- **Station** field

- Use the following syntax:

*<Device Address>:<Level of Interoperability>*

If the serial encapsulation is being used, the DNP driver supports the specification of the IP address on the station field, using the syntax described on Studio manual. However, the devices are identified by its address and not its IP address or port number, and thus the addresses must be unique on the DNP3 network.

Where:

<Device Address> is the address configured for the outstation.

**Optional** <Level of Interoperability> is the subset of commands/objects that this station supports. This is denoted by a number in the range 1 to 4, respectively to the subsets 1 to 4. If this field is not supplied, it is assumed level 1. Notice that the level informed must be consistent across driver sheets.

Examples:

- 1:1 → Device 1 with level of interoperability 1
- 192.168.11.21|2:3 → Device at 192.168.11.21, with address 2 and level 3
- 192.168.11.21:50000|3 → Device at 192.168.11.21:50000, with address 3 and default level (1)

➔ **Attention:**

- You cannot leave the Station field **blank**
- Even though the station field might be specified by a tag (e.g. {station}), the tag value must be specified before starting the driver task, for example on a startup script.

- **Header** field: Provides access to the services supported on the driver. The header specification is explained depending on the tasks to be performed:

### (a) Group/Variation services

The syntax used for the header field for this task is: <Service>:<Group>.<Variation>:<Offset>

The valid services and Studio operations associated with them are shown on the following table:

Service	Usage	Read	Write
STATIC	Read static data	•	-
MONITOR	Monitor of unsolicited or polled data (events)	-	-
FRZ	Freeze immediately	-	•
FRZNR	Freeze immediately without response	-	•
FRZCLR	Freeze and clear	-	•
FRZCLRNR	Freeze and clear without response	-	•

The Group and Variation parts of the header syntax are one of the pairs presented on the table of page 4.

For the **STATIC** service, it is valid to use 0 (zero) as the Variation, indicating any variation received is valid. If any of the addresses are not found on the response from the device, the tag associated receives quality BAD.

For the **MONITOR** service, the variation must be set with the exact variation that is sent from the device for unsolicited responses or polled data, associated with events. **MONITOR** headers receive data of unsolicited responses or from polls of the device.

For the freezing services (FRZ, FRZNR, FRZCLR and FRZCLRNR), it is important to note that the whole range of addresses on the driver sheet is requested to freeze.

- **Address syntax:** <Index>.<Qualifier>

Where:

<Index> is the index of the element on the device.

<Qualifier> indicates a quality field associated with the index, denoted by one of the following: OL, RS, CL, RF, LF, CF, RO, OR, DC, RE, STATUS

These quality fields are associated with the groups and variations by the following table, denoted as g#v#:

Qualifier	Groups/Variations
OL, RS, CL, RF, LF	g1v2, g2v1, g2v2, g2v3, g3v2, g4v1, g4v2, g4v3, g10v2, g11v1, g11v2, g20v1, g20v2, g21v1, g21v2, g21v5, g21v6, g22v1, g22v2, g22v5, g22v6, g23v1, g23v2, g23v5, g23v6, g30v1, g30v2, g30v5, g30v6, g31v1, g31v2, g31v3, g31v4, g31v7, g31v8, g32v1, g32v2, g32v3, g32v4, g32v5, g32v6, g32v7, g32v8, g33v1, g33v2, g33v3, g33v4, g33v5, g33v6, g33v7, g33v8, g40v1, g40v2, g40v3, g40v4, g42v1, g42v2, g42v3, g42v4, g42v5, g42v6, g42v7, g42v8
CF	g1v2, g2v1, g2v2, g2v3, g3v2, g4v1, g4v2, g4v3
RO, DC	g20v1, g20v2, g21v1, g21v2, g21v5, g21v6, g22v1, g22v2, g22v5, g22v6, g23v1, g23v2, g23v5, g23v6
OR, RE	g30v1, g30v2, g30v5, g30v6, g31v1, g31v2, g31v3, g31v4, g31v7, g31v8, g32v1, g32v2, g32v3, g32v4, g32v5, g32v6, g32v7, g32v8, g33v1, g33v2, g33v3, g33v4, g33v5, g33v6, g33v7, g33v8, g40v1, g40v2, g40v3, g40v4, g42v1, g42v2, g42v3, g42v4, g42v5, g42v6, g42v7, g42v8
STATUS	g13v1, g13v2, g43v1, g43v2, g43v3, g43v4, g43v5, g43v6, g43v7, g43v8

### (b) Data Polling

The data polling for integrity of events is performed by the header `POLL`. This header sends a read request for class information.

- **Address syntax:** `<ClassNumber>`

Where: `<ClassNumber>` Is the class to be polled, in the range 0 – 3.

Polling is performed by a write operation, and read operations are invalid for this header. The data returned by the poll is received by `MONITOR` headers.

### (c) Internal Indications (IIN) monitoring

Internal indications are included within every message received from the device. The bits of IIN contain important information regarding the status of the device and the last request issued. The header is configured as `IIN`.

- **Address syntax:** `<Byte>.<Bit>`

The IIN are bits within a word, so the `<Byte>` part is in the range 1 – 2 and the `<Bit>` part is in the range 0 – 7. The DNP driver does not accept read operations on the IIN header. However, the user can write to the bits of IIN, to inform the device a condition, for example that the “Need time” flag (1.4) shall be off.

Some IIN are associated with error conditions of requests. These bits are indicated on the table below with the associated error code. The detailed descriptions of each IIN may be found on the DNP documentation. The table below shows the short descriptions of each bit:

IIN (Address)	Description	Error Code
1.0	All Stations	
1.1	Class 1 events	
1.2	Class 2 events	
1.3	Class 3 events	
1.4	Need time	
1.5	Local control	
1.6	Device trouble	
1.7	Device restart	
2.0	Function code not supported	13
2.1	Object unknown	14
2.2	Parameter error	15
2.3	Event buffer overflow	
2.4	Already executing	
2.5	Configuration corrupt	

### (d) Control Relay Output Block (CROB) commands

This header is used to send commands to change binary outputs. The DNP3 protocol specifies many parameters for a CROB command. On the DNP driver, those are specified on the addresses, and the header is filled with **CROB**. For that, only Write Group operations are supported for this header, using write trigger on Studio. The parameters are specified below, along with the error code returned in case of invalid values.

Address	Description	Accepted Values		Default	Error code
<b>FUNCTION</b>	The function associated with the command	<b>SELECT</b>	Select	Not applicable	20
		<b>OPERATE</b>	Operate		
		<b>DIRECTOP</b>	Direct Operate		
		<b>DIRECTOPNR</b>	Direct Operate without Response		
<b>INDEX</b>	The point index to be operated	Integer greater or equal than 0		Not applicable	21
<b>OPTYPE</b>	The operation to be performed on the point	<b>NUL</b>	Nul. Device specified.	<b>NUL</b>	22
		<b>PULSEON</b>	Pulse On		
		<b>PULSEOFF</b>	Pulse Off		
		<b>LATCHON</b>	Latch On		
		<b>LATCHOFF</b>	Latch Off		
<b>CLEAR</b>	Clear flag.	0 or 1		0	23
<b>TRIPCODE</b>	Used with <b>OPTYPE</b> to describe the operation.	<b>NUL</b>	Nul	<b>NUL</b>	24
		<b>CLOSE</b>	Close		
		<b>TRIP</b>	Trip		
<b>COUNT</b>	Number of times to execute the operation.	Integer greater or equal than 1 and less than or equal to 255		1	25
<b>ONTIME</b>	Duration in milliseconds to remain active.	Integer greater or equal than 0		Not applicable	26
<b>OFFTIME</b>	Duration in milliseconds to remain non-active.	Integer greater or equal than 0		0	27

For the parameters where the defaults are not applicable, not informing the value results on error code 12, meaning not enough parameters. Write Item and read operations are not allowed. Only one point can be operated on the same request to the device, indicated by the **INDEX** parameter. For further information on each parameter, consult the DNP3 protocol documentation on Group 12 Variation 1, and your device documentation regarding binary outputs.

### (e) Analog Output Block (AOB) commands

This header is used to send commands to change analog outputs. The DNP3 protocol specifies many parameters for an AOB command. On the DNP driver, those are specified on the addresses, and the header is filled with **AOB**. For that, only Write Group operations are supported for this header, using write trigger on Studio. The parameters are specified below, along with the error code returned in case of invalid values.

Address	Description	Accepted Values		Default	Error code
<b>FUNCTION</b>	The function associated with the command	<b>SELECT</b>	Select	Not applicable	20
		<b>OPERATE</b>	Operate		
		<b>DIRECTOP</b>	Direct Operate		
		<b>DIRECTOPNR</b>	Direct Operate without Response		
<b>INDEX</b>	The point index to be operated	Integer greater or equal than 0		Not applicable	21
<b>VALUE</b>	The value to change the analog output to.	Any numeric value.		Not applicable	28,30
<b>TYPE</b>	The type variation associated with the analog output point.	<b>INT</b>	32-bit signed integer	<b>SINT</b>	29
		<b>SINT</b>	16-bit signed integer		
		<b>FLOAT</b>	32-bit IEEE-754 floating point		
		<b>DOUBLE</b>	64-bit IEEE-754 floating point		

For the parameters where the defaults are not applicable, not informing the value results on error code 12, meaning not enough parameters. Write Item and read operations are not allowed. Only one point can be operated on the same request to the device, indicated by the **INDEX** parameter. For further information on each parameter, consult the DNP3 protocol documentation on Group 41 and its variations, and your device documentation regarding binary outputs.

### (f) Configuration commands

This header is used to change configuration of the device or send to it other commands associated with its configuration. The header is filled with **CMD**, and the commands are specified on the addresses. This command does not support read nor group write operations. Commands are issued on a per address basis to avoid multiple commands to be issued undesirably, using write on tag change on Studio. The following table describes the addresses supported by this header.

Address	Description	Accepted Values	Error code
<b>COLDRESTART</b>	Sends a cold restart request to the device.	Any.	
<b>WARMRESTART</b>	Sends a warm restart request to the device.	Any.	
<b>ENABLEUNSOL</b>	Send an enable unsolicited responses request to the device for a specific class.	1 – 3	10
<b>DISABLEUNSOL</b>	Send a disable unsolicited responses request to the device for a specific class.	1 – 3	10

The cold and warm restarts are triggered by having any value on a tag associated with the address. After sending these commands, the driver waits for a number of seconds determined by the device on a reply message.

The enable and disable unsolicited responses requests are associated with class numbers. The class number is determined on the tag value sent to the DNP driver, on the range 1 – 3, respectively to the classes 1 – 3 of the DNP3 protocol and device configuration. If a value outside this range is informed, the driver returns error code 10, meaning invalid choice and the command is not sent.

### (g) Status monitoring

This header is used to inform errors to the user, such as malformed packets and timeouts on connection. The **STATUS** header does not require the station address to be configured. The addresses are configured with the device address. Neither read nor write operations are allowed on this header.

#### 4. Examples

Header	Addresses (Values)	Description
STATIC:1.2	1 2 10	Reads binary inputs with variation 2 on the range 1 – 10.
STATIC:10.2	0.OL 0.RS 0.CL 0.RF 0.LF	Reads the quality fields of the binary output point 0.
MONITOR:2.3	0 1 3	Monitor the values of unsolicited responses or polled values of binary input on variation 3.
FRZCLRNR:20.0	0 4	Freeze and clear without response the counters in range 0 – 4.
POLL	0 1 2	Perform class polling on classes 0, 1 and 2.
IIN	1.1 1.2 1.3	Receives the flags for each received message for the class 1, 2 and 3 events.



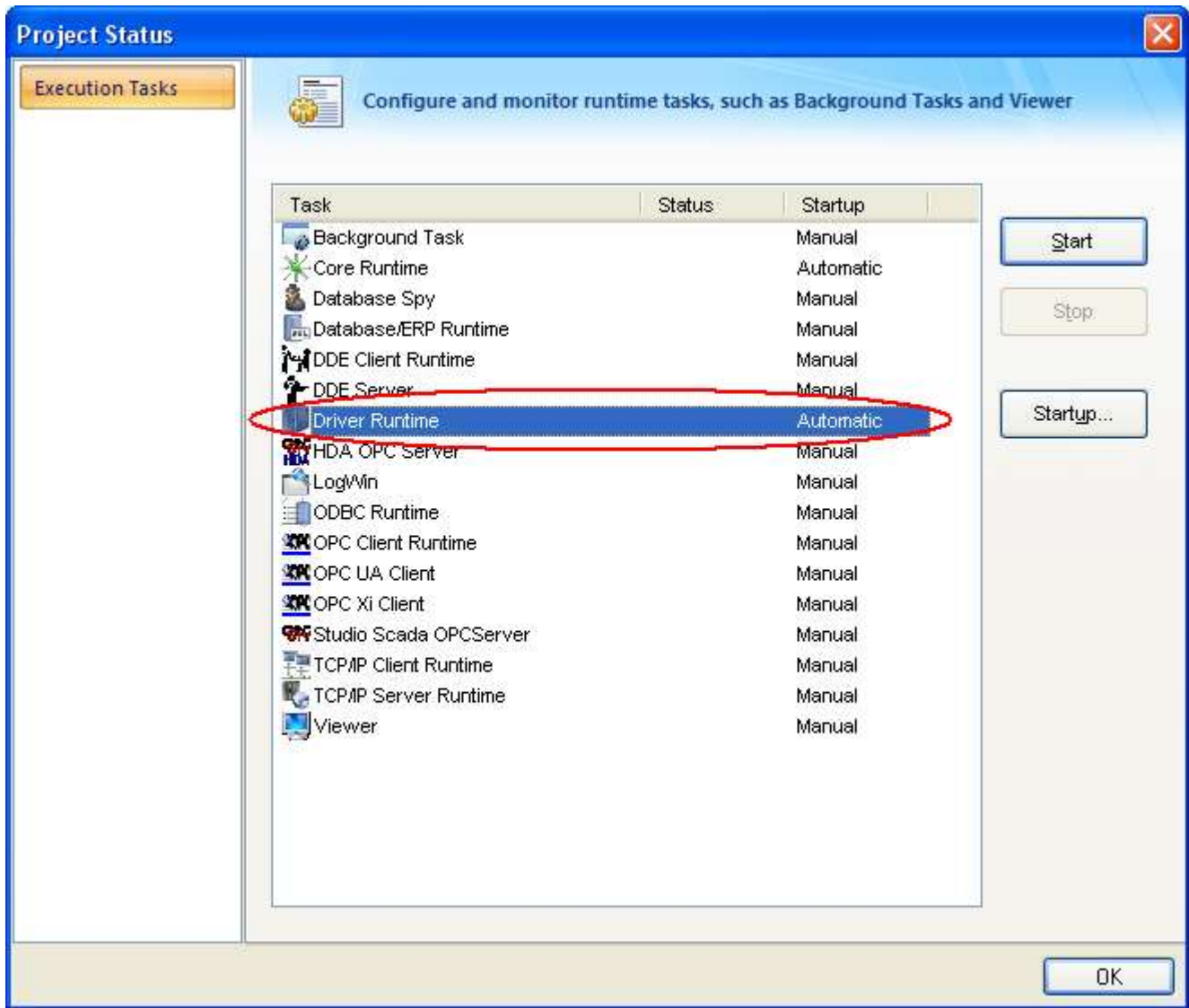
Header	Addresses (Values)	Description
CROB	FUNCTION (DIRECTOP) INDEX (40) ONTIME (1000) OPTYPE (PULSEON)	Send a request to change a binary output, with a direct operation, on point 40, to remain active for 1 second, with a pulse on operation.
AOB	FUNCTION (SELECT) INDEX (10) TYPE (DOUBLE) VALUE (152.35)	Send a request to change an analog output, with select, on point 10, with value 152.35 and type double.
CMD	ENABLEUNSOL (1)	Send a request to enable unsolicited responses of class 1 events.
CMD	COLDRESTART	Send a request to do a cold restart on the device.
STATUS	4	Monitor the status of the device with address 4.

## Executing the Driver

By default, Studio will automatically execute your selected communication driver(s) during application runtime. However, you may verify your application’s runtime execution settings by checking the *Project Status* dialog.

To verify that the communication driver(s) will execute correctly:

1. From the main menu bar, select **Project** → **Status**. The *Project Status* dialog displays:



*Project Status Dialog*

2. Verify that the *Driver Runtime* task is set to **Automatic**.
  - If the setting is correct, then proceed to step 3 below.
  - If the **Driver Runtime** task is set to **Manual**, then select the task and click the **Startup** button to toggle the task’s *Startup* mode to **Automatic**.

3. Click **OK** to close the *Project Status* dialog.
4. Start the application to run the driver.

## Troubleshooting

If the DNP driver fails to communicate with the target device, then the database tag(s) that you configured for the **Read Status** or **Write Status** fields of the Standard Driver Sheet will receive an error code. Use this error code and the following table to identify what kind of failure occurred.

Error Code	Description	Possible Causes	Procedure to Solve
0	Ok	N/A	N/A
1	Wrong address	<ul style="list-style-type: none"> <li>- The device address does not match the address configured on any of the stations.</li> <li>- The destination address of the message is not the address configured for the master.</li> </ul>	<ul style="list-style-type: none"> <li>- Check the device configuration for the address</li> <li>- Check the station configurations</li> <li>- Check the station configured on the advanced settings to match the master address configured on the device</li> </ul>
2	CRC error	The message contained a malformed CRC	<ul style="list-style-type: none"> <li>- Check the network for noise</li> <li>- Check the connections for possible network errors</li> </ul>
3	Invalid data link layer function	Indicates an invalid function was received from the device	<ul style="list-style-type: none"> <li>- Check device configurations</li> </ul>
4	Data link layer wrong direction	Received a message from another master	<ul style="list-style-type: none"> <li>- Check device configurations</li> <li>- Check station address on advanced settings</li> </ul>
5	Data link layer timeout for reset	The driver attempted to reset the number of times configured on retries, and no response was received	<ul style="list-style-type: none"> <li>- Check network configurations for noise and errors</li> <li>- Check device configurations for replies of Data Link Layer commands</li> </ul>
6	Data link layer timeout for confirmation	The driver has not received the confirmation for data link layer messages	<ul style="list-style-type: none"> <li>- Check network configurations for noise and errors</li> <li>- Check device configurations for replies of Data Link Layer commands</li> </ul>
7	Data link layer timeout for status	The driver has not received the response for a request for link status	<ul style="list-style-type: none"> <li>- Check network configurations for noise and errors</li> <li>- Check device configurations for replies of Data Link Layer commands</li> </ul>
8	Application layer timeout for receive	A response for a request from application layer was not received before timeout	<ul style="list-style-type: none"> <li>- Check network configurations for noise and errors</li> <li>- Check device configurations for responses</li> <li>- Check configurations of commands for requesting responses</li> </ul>
9	Invalid qualifier	The message contained an invalid qualifier, not described on interoperable table of DNP3 protocol or not supported for responses.	<ul style="list-style-type: none"> <li>- Check device configurations for responses</li> </ul>
10	Invalid choice	<ul style="list-style-type: none"> <li>- The configuration value on one of the settings is invalid</li> <li>- The class number requested for enable/disable unsolicited responses is invalid</li> </ul>	<ul style="list-style-type: none"> <li>- Fill the configuration with a valid value (see the section configuration of this document)</li> <li>- If unexpected, check the class number set on the tag value for enabling/disabling unsolicited responses</li> </ul>
11	Invalid operation	A read or write was requested for a header that does not support the operation	<ul style="list-style-type: none"> <li>- Check read/write triggers</li> <li>- Check headers configuration</li> </ul>
12	Not enough parameters	The minimum parameters were not supplied for headers CROB or AOB	<ul style="list-style-type: none"> <li>- Check the configuration of addresses for headers CROB and AOB to have at least the parameters that do not have default values. Check the table on the proper section of this document.</li> </ul>
13	Function not supported	Returned from the device, indicates the operation requested on the object is invalid, for example freezing binary inputs.	<ul style="list-style-type: none"> <li>- Check if the object supports the operation requested</li> <li>- Check header configurations</li> <li>- See DNP3 protocol documentation for supported functions and objects</li> </ul>

14	Object unknown	The object (group and/or variation) is invalid for the device.	- Check if the device has support for the requested object - See DNP3 protocol documentation for supported functions and objects
15	Parameter error	A parameter on the message is incorrect, such as the valid range of addresses, or the qualifier of addresses.	- Check if the range of addresses requested is valid on the device - Contact support to check for problems with the qualifiers.
20	Invalid choice of function	The value set on the address for a CROB or AOB function is invalid.	- Check that the value for the address FUNCTION of headers CROB and AOB receive one of the values described on this document on the proper section.
21	Invalid choice of index	The value set on the address for a CROB or AOB index is invalid.	- Check that the value for the address INDEX of headers CROB and AOB receive one of the values described on this document on the proper section.
22	Invalid choice of optype	The value set on the address for a CROB optype is invalid.	- Check that the value for the address OPTYPE of header CROB receive one of the values described on this document on the proper section.
23	Invalid choice of clear	The value set on the address for a CROB clear flag is invalid.	- Check that the value for the address CLEAR of header CROB receive one of the values described on this document on the proper section.
24	Invalid choice of trip code	The value set on the address for a CROB trip code is invalid.	- Check that the value for the address TRIPCODE of header CROB receive one of the values described on this document on the proper section.
25	Invalid choice of count	The value set on the address for a CROB count is invalid.	- Check that the value for the address COUNT of header CROB receive one of the values described on this document on the proper section.
26	Invalid choice of ontime	The value set on the address for a CROB on time is invalid.	- Check that the value for the address ONTIME of header CROB receive one of the values described on this document on the proper section.
27	Invalid choice of offtime	The value set on the address for a CROB off time is invalid.	- Check that the value for the address OFFTIME of header CROB receive one of the values described on this document on the proper section.
28	Invalid choice of value	The value set on the address for a AOB value is invalid.	- Check that the value for the address VALUE of header AOB receive one of the values described on this document on the proper section.
29	Invalid choice of type	The value set on the address for a AOB type is invalid.	- Check that the value for the address TYPE of header AOB receive one of the values described on this document on the proper section.
30	Type overflow	The value set for a AOB on field VALUE overflows the type determined by the field TYPE.	- Check the VALUE and TYPE addresses of header AOB.
31	Invalid address range for the qualifier	The address is incompatible with the level of interoperability of the device.	- Check if the address exists on the device - Check the address configurations
50	Unexpected state	The driver has reached an unexpected state.	- This is a critical error, please contact support.
51	Unsupported	The driver has reached an unsupported functionality.	- Check configurations of headers, addresses and operations - Check device configurations to match those of the driver - Review this document for the supported features
52	Busy	The driver is busy handling messages.	- This error may happen when large batches of messages are being handled, and is not critical in this case. - If the error is happening constantly, this becomes critical, please contact support.

100	Allocated frame is too small	The buffer size configured on settings for application layer fragment size is smaller than the fragments received.	- Check the device configurations for the fragment size to match the size configured on the driver - Check the driver configuration to increase the buffer size.
101	Message shorter than expected	The message contains less data than expected	- Check network configurations for noise and errors - Check device configurations for replies of Data Link Layer commands
102	Buffer internal error	The buffer is too small for the data requested.	- This is an internal error, please contact support.
201	Operation Timeout	Error returned from device. Check DNP3 documentation for status code 1.	- Check device and driver configurations. - See DNP3 documentation for further instructions.
202	Operation No select	Error returned from device. Check DNP3 documentation for status code 2.	- Check device and driver configurations. - See DNP3 documentation for further instructions.
203	Operation Format error	Error returned from device. Check DNP3 documentation for status code 3.	- Check device and driver configurations. - See DNP3 documentation for further instructions.
204	Operation Not supported	Error returned from device. Check DNP3 documentation for status code 4.	- Check device and driver configurations. - See DNP3 documentation for further instructions.
205	Operation Already active	Error returned from device. Check DNP3 documentation for status code 5.	- Check device and driver configurations. - See DNP3 documentation for further instructions.
206	Operation Hardware error	Error returned from device. Check DNP3 documentation for status code 6.	- Check device and driver configurations. - See DNP3 documentation for further instructions.
207	Operation Local	Error returned from device. Check DNP3 documentation for status code 7.	- Check device and driver configurations. - See DNP3 documentation for further instructions.
208	Operation Too many objects	Error returned from device. Check DNP3 documentation for status code 8.	- Check device and driver configurations. - See DNP3 documentation for further instructions.
209	Operation Not authorized	Error returned from device. Check DNP3 documentation for status code 9.	- Check device and driver configurations. - See DNP3 documentation for further instructions.
210	Operation Automation inhibit	Error returned from device. Check DNP3 documentation for status code 10.	- Check device and driver configurations. - See DNP3 documentation for further instructions.
211	Operation Processing limited	Error returned from device. Check DNP3 documentation for status code 11.	- Check device and driver configurations. - See DNP3 documentation for further instructions.
212	Operation Out of range	Error returned from device. Check DNP3 documentation for status code 12.	- Check device and driver configurations. - See DNP3 documentation for further instructions.
326	Operation Non participating	Error returned from device. Check DNP3 documentation for status code 126.	- Check device and driver configurations. - See DNP3 documentation for further instructions.
327	Operation Undefined error	Error returned from device. Check DNP3 documentation for status code 127.	- Check device and driver configurations. - See DNP3 documentation for further instructions.
-37	Invalid Header	Header was not recognized	- Check your driver sheet to ensure the requested operation is valid for the header, and that its syntax is correct
-38	Invalid Station	The syntax for the station field is incorrect. This error is shown on the log windows when attempting to create a connection with an invalid station.	- Check for the station syntax: - Check if the station field is filled - Check if device address is in range 0-65519 - If using serial encapsulation, check the value of IP address and port number

⇒ **Tip:**

You can monitor communication status by establishing an event log in Studio's *Output* window (*LogWin* module). To establish a log for **Field Read Commands**, **Field Write Commands** and **Protocol Analyzer**, right-click in the *Output* window and select the desired options from the pop-up menu.

You can also use the *Remote LogWin* module (**Tools** → **LogWin**) to establish an event log on a remote unit that runs Windows CE

If you are unable to establish communication between Studio and the target device, then try instead to establish communication using the device's own programming software. Quite often, communication is interrupted by a hardware or cable problem or by a device configuration error. If you can successfully communicate using the programming software, then recheck the driver's communication settings in Studio.

If you must contact us for technical support, please have the following information available:

- **Operating System and Project Information** (type and version): To find this information, select **Help** → **Support Information**.
- **Driver Version and Communication Log**: Displays in the Studio *Output* window when the driver is running.
- **Device Model and Boards**: Consult the hardware manufacturer's documentation for this information.

## Sample Application

A sample application that employs the DNP driver is provided on the Studio installation CD. We strongly recommend that you use this sample application to test the driver *before* you develop your own applications, for the following reasons:

- To better understand the information and instructions provided in this document;
- To verify that your driver configuration is working satisfactorily with the target device; and
- To ensure that the all of hardware used in the test (i.e. the device, adapter, cable, and PC) is functioning safely and correctly.

 **Note:**

The following instructions assume that you are familiar with developing project applications in Studio. If you are not, then please review the relevant chapters of the Studio *Technical Reference Manual* before proceeding.

To use the sample application:

1. Configure the device's communication settings according to the manufacturer's documentation.
2. Run Studio.
3. From the main menu bar, select **File → Open Project**.
4. Insert the Studio installation CD and browse it to find the sample application. It should be located in the directory `\COMMUNICATION EXAMPLES\DNP`.
5. Select and open the sample application.
6. Configure and test the driver, as described in the rest of this document.

When you have thoroughly tested the driver with your target device, you may proceed with developing your own Studio application projects.

 **Tip:**

You can use the sample application screen as the maintenance screen for your own applications.



## Revision History

Doc. Revision	Driver Version	Author	Date	Description of Changes
A	1.0	André Körbes	Aug. 1 <sup>st</sup> , 2011	Initial version
B	1.1	André Körbes	Oct. 18 <sup>th</sup> , 2011	Changed a parameter on driver INI
C	1.2	André Körbes	Mar. 6, 2013	<ul style="list-style-type: none"><li>• Modified the driver to support more functions.</li><li>• Fixed STATIC read functions to fill the timestamp when it is not received.</li></ul>
D	1.3	Eduardo Castro	Jan. 29, 2016	<ul style="list-style-type: none"><li>• Fixed time zone for pooled tags timestamps</li><li>• Increased number of supported data inputs</li><li>• Fixed Application Layer Timeout problems</li></ul>