

# NR500 Series Industrial Cellular VPN Router

## Application Note 061

### MQTT to DO and Modbus

**Version:** V1.0.0  
**Date:** Sep. 2021  
**Status:** Confidential



## Directory

1. Introduction.....	3
1.1 Overview.....	3
1.2 Compatibility.....	3
1.3 Version.....	3
1.4 Corrections.....	3
2. Topology.....	4
3. MQTT to Digital Output.....	5
3.1 Configuration on NR500.....	5
3.2 Test.....	6
4. MQTT to Modbus.....	7
4.1 Configuration on Modbus Slave.....	7
4.2 Configuration on NR500.....	7
4.3 Test.....	9

# 1. Introduction

## 1.1 Overview

This document contains information regarding the configuration and use of MQTT to DO and Modbus.

This guide has been written for use by technically competent personnel with a good understanding of the communications technologies used in the product, and of the requirements for their specific application.

## 1.2 Compatibility

This application note applies to:

**Models Shown:** NR500 series.

**Firmware Version:** V1.1.6(2e031bb) or newer

**Other Compatible Models:** None

## 1.3 Version

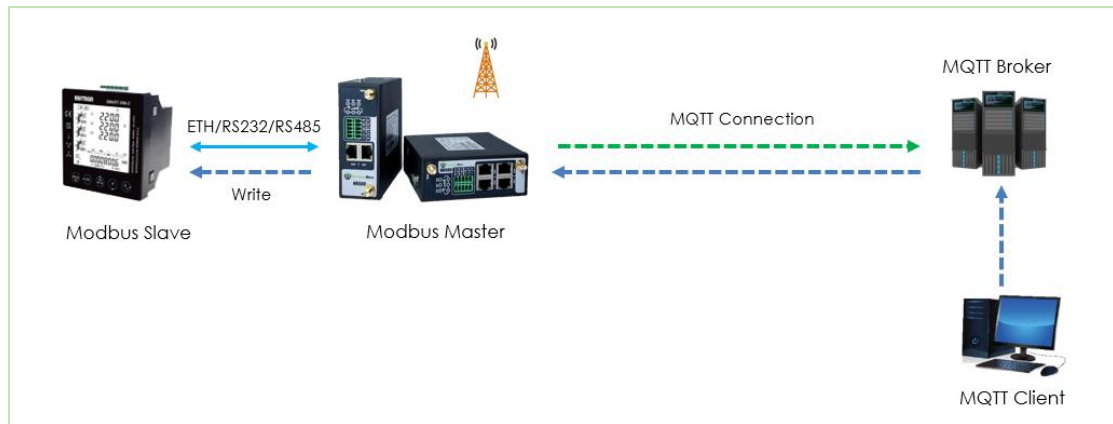
Updates between document versions are cumulative. Therefore, the latest document will include all the content of previous versions.

Release Date	Doc. Version	Firmware Version	Change Description
2021/09/30	V1.0.0	V1.1.6(2e031bb)	First released

## 1.4 Corrections

Appreciate for corrections or rectifications to this application note, and if any request for new application notes please email to: **support@navigateworx.com**

## 2. Topology



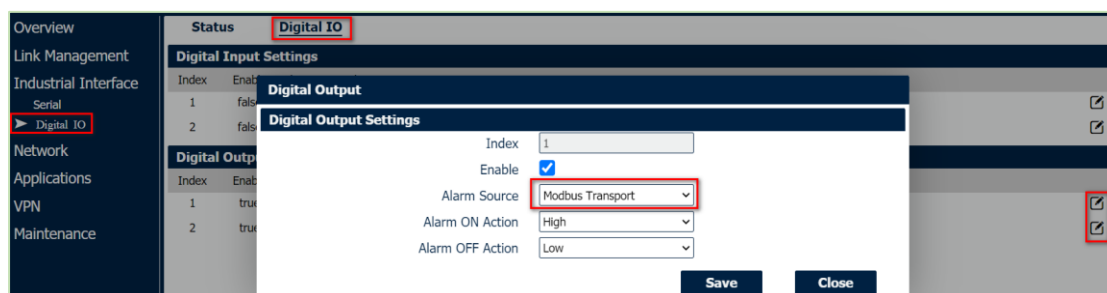
1. NR500 Router runs as Modbus Master and connect to Modbus Slave via Ethernet, RS232 or RS485 interface.
2. NR500 runs as MQTT Client and connect to the MQTT Broker
3. Another MQTT Client connect to the MQTT Broker and send the commands to control the Digital Output and write the Slave device

*Note: For this Application Note, we will set the Connection Type as "RS232" as an example, which means that NR500(Modbus Master) will connect to the Modbus Slave via RS232. Of course, it also works with RS485 and Ethernet Port.*

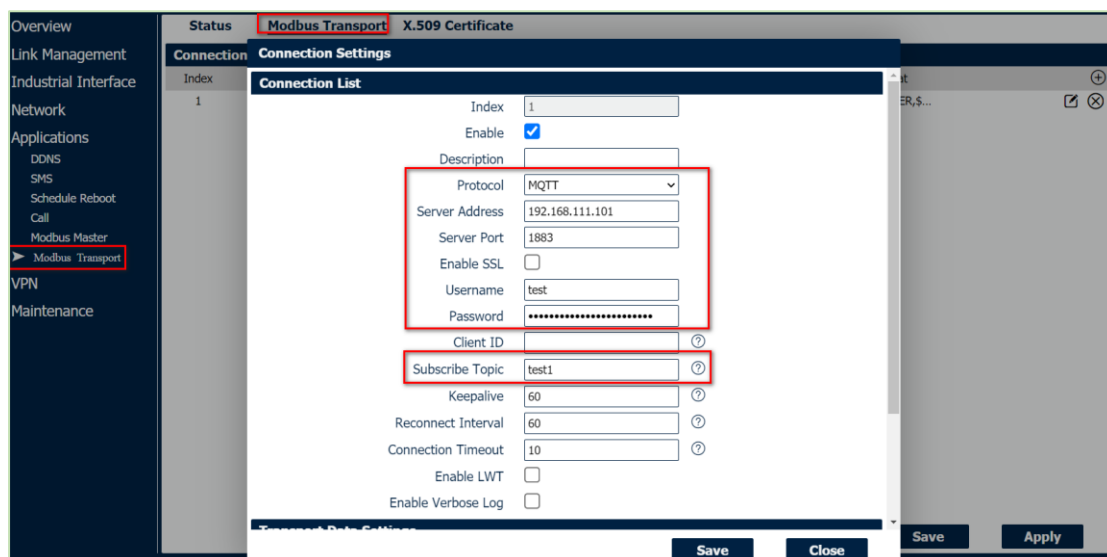
### 3. MQTT to Digital Output

#### 3.1 Configuration on NR500

1. Go to **Industrial Interface>Digital IO**, enable digital output feature and set the Alarm Source as Modbus Transport, as below:



2. Click Save>Apply.
3. Go to **Applications>Modbus Transport**, to specify the parameters for MQTT to make the router connect to MQTT Broker, and also set a Subscribe Topic as "test1", the other settings leave it as default.

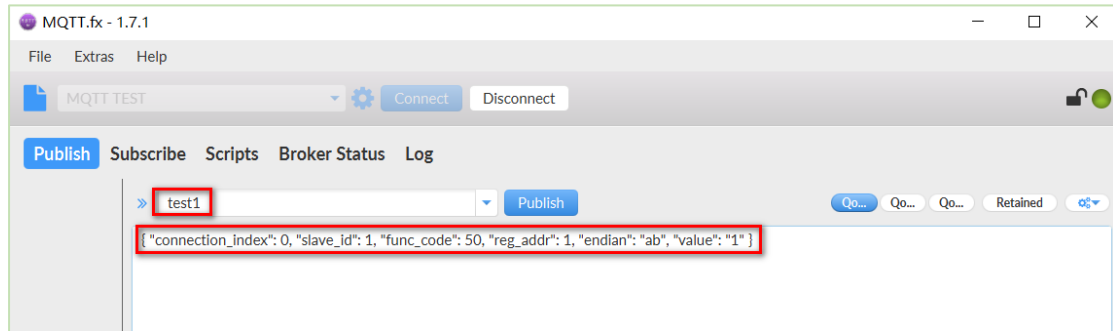


4. Click Save>Apply.
5. Router had connected to MQTT Broker successfully:

Status Modbus Transport X.509 Certificate						
Connection Status						
Index	Enable	Description	Protocol	Status	Uptime	
1	true		MQTT	Connected	00:07:23	

## 3.2 Test

1. Run the MQTT Client on the PC and connect to MQTT Broker, Publish a topic as "test1", and send the command to the router to control the Digital Output:



2. Test successfully:

Overview	Status	Digital IO		
Link Management	<b>Digital Input Information</b>			
Industrial Interface	Index	Enable	Logic Level	Status
Serial	1	false	High	Alarm OFF
▶ Digital IO	2	false	High	Alarm OFF
Network	<b>Digital Output Information</b>			
Applications	Index	Enable	Logic Level	Status
VPN	1	true	High	Alarm ON
Maintenance	2	true	High	Alarm ON

**Note: Command explanation as below:**

1. **Command to control the Digital Output 1 ON:**

```
{ "connection_index": 0, "slave_id": 1, "func_code": 50, "reg_addr": 1, "endian": "ab", "value": "1" }
```

2. **Command to control the Digital Output 1 OFF:**

```
{ "connection_index": 0, "slave_id": 1, "func_code": 50, "reg_addr": 1, "endian": "ab", "value": "0" }
```

3. **Command to control the Digital Output 2 ON:**

```
{ "connection_index": 0, "slave_id": 1, "func_code": 50, "reg_addr": 2, "endian": "ab", "value": "1" }
```

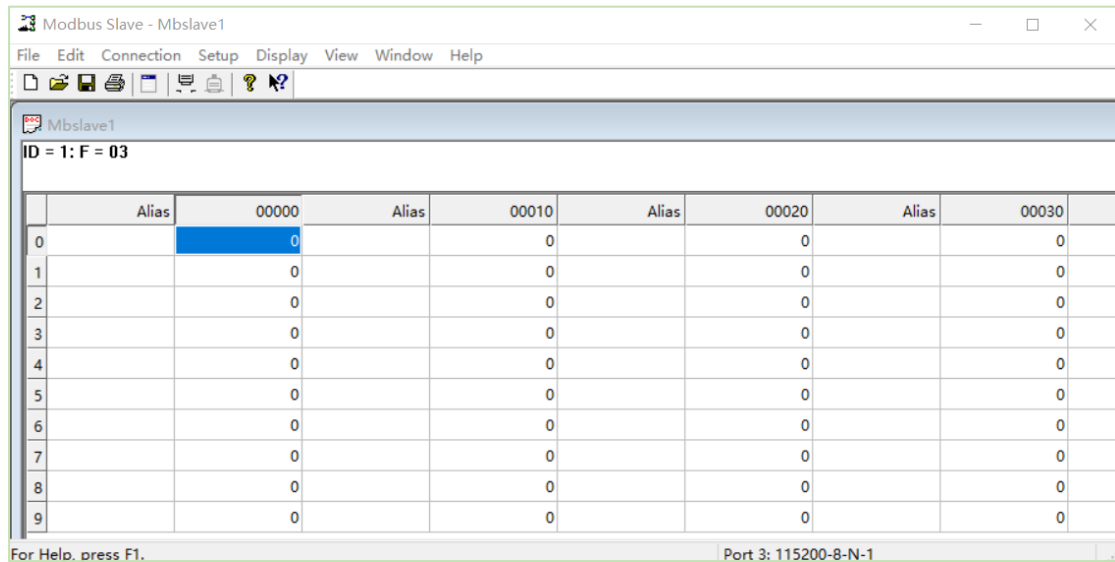
4. **Command to control the Digital Output 2 OFF:**

```
{ "connection_index": 0, "slave_id": 1, "func_code": 50, "reg_addr": 2, "endian": "ab", "value": "0" }
```

## 4. MQTT to Modbus

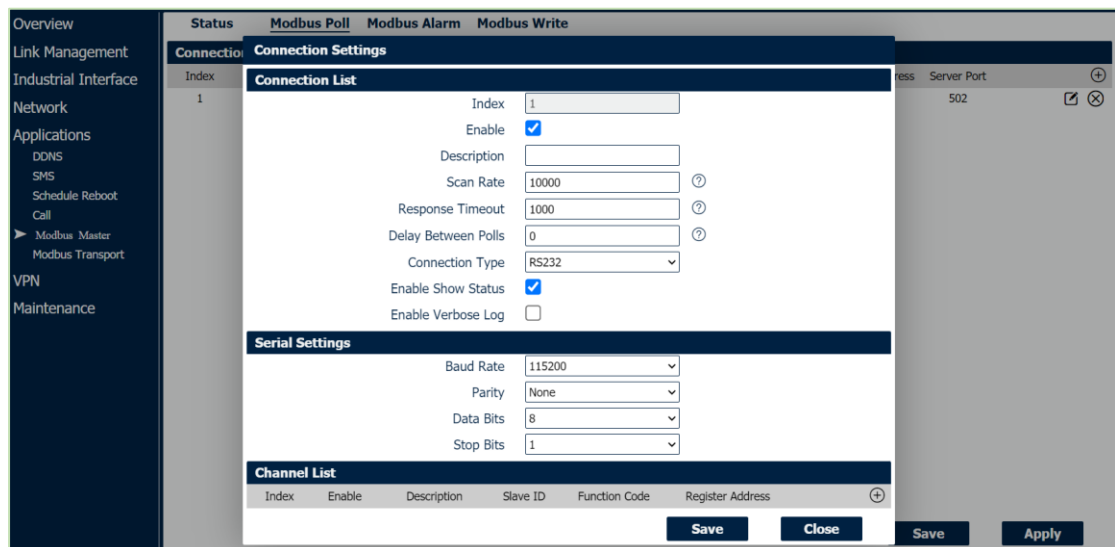
### 4.1 Configuration on Modbus Slave

- Here we use “Modbus Slave” software to simulate the end device (Modbus Slave device), and specify the **Slave ID: 1, Function Code: 03-Holding-Register:**

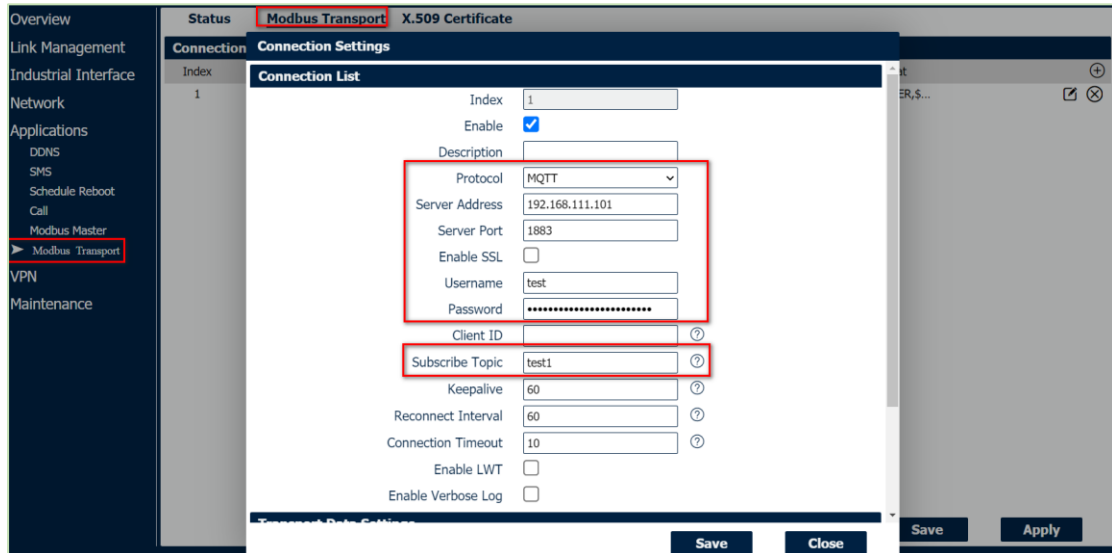


### 4.2 Configuration on NR500

- Go to **Applications>Modbus Master>Modbus Poll**, specify the Modbus setting to make it connect to the slave, as below:



2. Click Save>Apply.
3. Go to **Applications>Modbus Transport**, to specify the parameters for MQTT to make the router connect to MQTT Broker, and also set a Subscribe Topic as "test1", the other settings leave it as default.



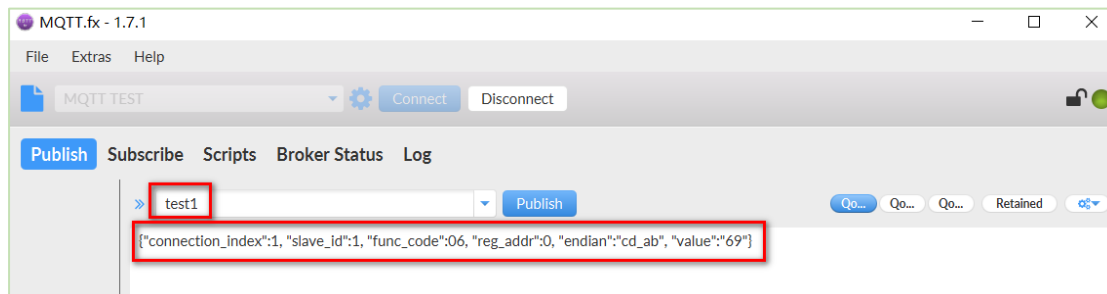
4. Click Save>Apply.
5. Router had connected to MQTT Broker successfully:

Status Modbus Transport X.509 Certificate						
Connection Status						
Index	Enable	Description	Protocol	Status	Uptime	
1	true		MQTT	Connected	00:07:23	

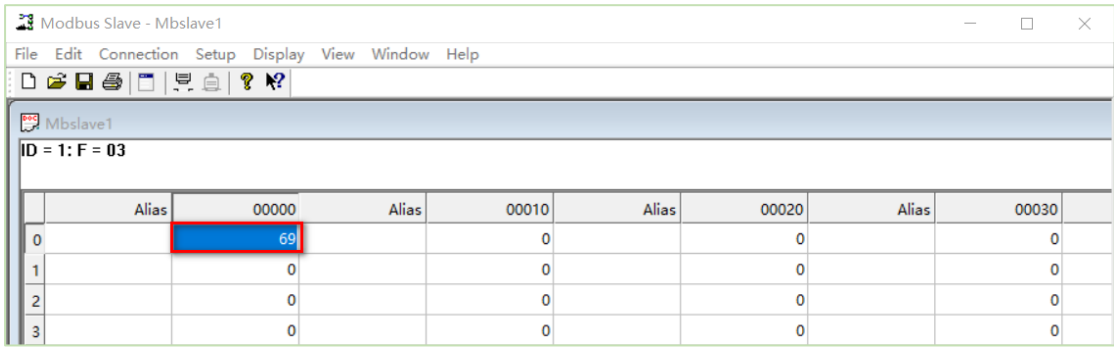


## 4.3 Test

1. Run the MQTT Client on the PC and connect to MQTT Broker, Publish a topic as "test1", and send the command to the router to control the slave device:



2. Test successfully, finally we are able to send the command via MQTT to control the slave device:



	Alias	00000	Alias	00010	Alias	00020	Alias	00030
0		69		0		0		0
1		0		0		0		0
2		0		0		0		0
3		0		0		0		0

### Note:

Control command to set the value as 69 to the slave 1 and register address 0 as an example:

```
{"connection_index":1, "slave_id":1, "func_code":06, "reg_addr":0, "endian":"cd_ab", "value":"69"}
```