

## Application Note – Reading and Writing ISDU from Modbus TCP PLC (Simulator)

Detail: Scheme and Method for using two sessions of MB Poll to read/edit ISDU in any IOL device

### Prerequisites:

Any model of IOLM that has Modbus TCP enabled

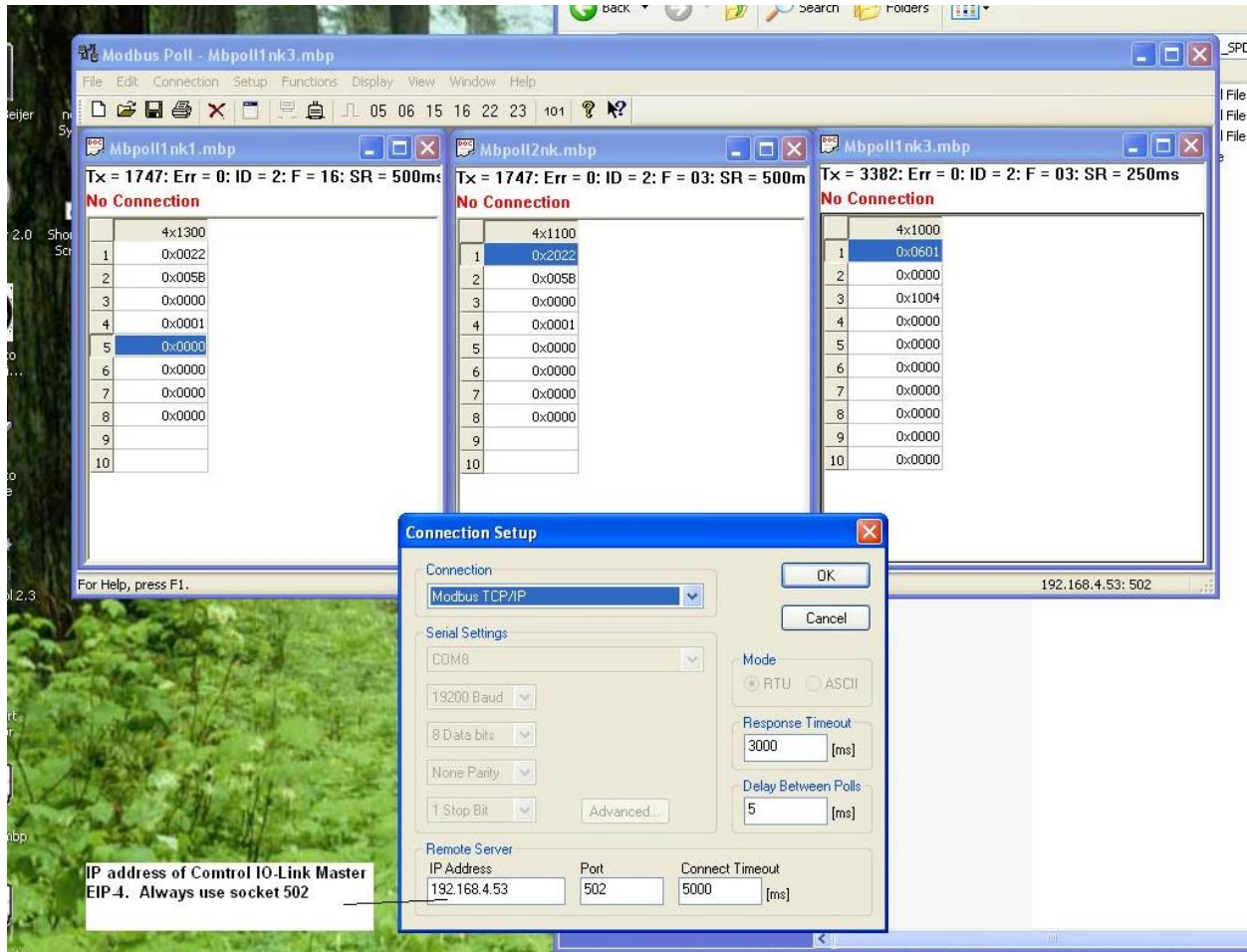
All necessary cables and power supplies and sensors

Powered up with appropriate IP address

No network/security/rules interference between PLC/NIC/ModbusPoll

Date of Note (implies era of relevance): 5/16/19

- 1) Open Modbus Poll and then open three separate windows within. (Actually the far right one is not needed.) You must already be familiar with Modbus Poll. See Application Note "PDI\_to\_ModbusTCP" if you have not done that yet.



2) Note that under setup "Read write Definition," you will need the following settings:

The screenshot shows the Modbus Poll application interface. Two windows, 'Mbpoll1' and 'Mbpoll2', are open, both displaying 'No Connection' and a table with columns 'Alias' and '00000'. A green arrow points from the 'No Connection' text in the 'Mbpoll1' window to the 'Read/Write Definition' dialog box.

The 'Read/Write Definition' dialog box contains the following settings:

- Slave ID: 1
- Function: 16 Write Multiple Registers
- Address: 1301
- Quantity: 8
- Scan Rate: 500 ms
- Read/Write Enabled
- View:
  - Rows:  10  20  50  100
  - Hide Alias Columns
  - Address in Cell
  - PLC Addresses (Base 1)
- Buttons: OK, Cancel, Apply, Read/Write Once

3) Under the window immediately to its right, use these settings (for a device on port X, the first digit should be X.) For port 1, 1101.

The screenshot shows the Modbus Poll application interface. At the top, there is a menu bar (File, Edit, Connection, Setup, Functions, Display, View, Window, Help) and a toolbar. Below the toolbar, there are two poll windows and a configuration dialog.

The left window, titled "Mbpoll1", shows the status "Tx = 0: Err = 0: ID = 1: F = 16: SR = 500ms" and "No Connection". It contains a table with 10 rows and 2 columns:

	4x1300
1	0x0000
2	0x0000
3	0x0000
4	0x0000
5	0x0000
6	0x0000
7	0x0000
8	0x0000
9	
10	

The right window, titled "Mbpoll2", shows the status "Tx = 0: Err = 0: ID = 1: F = 03: SR = 1000ms" and "No Connection". It contains a table with 10 rows and 2 columns:

	Alias	00000
0		0
1		0
2		0
3		0
4		0
5		0
6		0
7		0
8		0
9		0

The "Read/Write Definition" dialog box is open, showing the following settings:

- Slave ID: 1
- Function: 03 Read Holding Registers (4x)
- Address: 1101
- Quantity: 8
- Scan Rate: 500 ms
- Read/Write Enabled
- View: Rows (radio buttons for 10, 20, 50, 100; 10 is selected)
- Display: Hex
- Hide Alias Columns
- Address in Cell
- PLC Addresses (Base 1)

Buttons for OK, Cancel, and Apply are visible. A green arrow points from the "03 Read Holding Registers (4x)" function dropdown in the dialog to the "F = 03" status in the Mbpoll2 window.

At the bottom left, it says "For Help, press F1." At the bottom right, the IP address "192.168.16.44: 502" is displayed.

- 4) See the annotated guide below. Important always are opcode (0021h = read, 0022h = write), index, subindex, length, and (if writing) data

The screenshot displays three Modbus Poll connections and their configuration dialog boxes:

- Mbpoll1nk1.mbp:** Tx = 412; Err = 0; ID = 2; F = 16; SR = 500ms. This connection is annotated as "This is writing a 0 to index 91". The data table shows:
 

1	0x0022
2	0x005B
3	0x0000
4	0x0001
5	0x0000
6	0x0000
7	0x0000
8	0x0000
9	
10	
- Mbpoll2nk.mbp:** Tx = 412; Err = 0; ID = 2; F = 03; SR = 500ms. This connection is annotated as "This is reading index 91". The data table shows:
 

1	0x2022
2	0x005B
3	0x0000
4	0x0001
5	0x0000
6	0x0000
7	0x0000
8	0x0000
9	
10	
- Mbpoll1nk3.mbp:** Tx = 798; Err = 0; ID = 2; F = 03; SR = 250ms. This connection is annotated as "This is reading process data from the device plugged into Port 1". The data table shows:
 

1	0x0601
2	0x0000
3	0x2404
4	0x0000
5	0x0000
6	0x0000
7	0x0000
8	0x0000
9	0x0000
10	0x0000

Three "Read/Write Definition" dialog boxes are shown at the bottom, each with a green box around the "Address" field:

- Left Dialog:** Slave ID: 2, Function: 16 Write Multiple Registers, Address: 1301, Quantity: 8, Scan Rate: 500 ms.
- Middle Dialog:** Slave ID: 2, Function: 03 Read Holding Registers (4x), Address: 1101, Quantity: 8, Scan Rate: 500 ms.
- Right Dialog:** Slave ID: 2, Function: 03 Read Holding Registers (4x), Address: 1001, Quantity: 10, Scan Rate: 250 ms.

Annotations include: "op code and format", "index 91 (dec)", "sub index 0", "data length 1 byte", "0 = degC, 1 = degF", "op code, format, cmd status", "status of PDI (600 and 601 are desirable)", and "Process data payload".



5) Please see the additional annotations in this image

The screenshot displays the Modbus Poll software interface with three data windows and three configuration dialog boxes. Annotations explain the data fields and configurations.

**Annotations:**

- Leftmost window (Mbpoll1nk1.mbp):**
  - Row 1: 0x0022 - op code and format
  - Row 2: 0x005B - index 91 (dec)
  - Row 3: 0x0000 - sub index 0
  - Row 4: 0x0001 - data length 1 byte
  - Row 5: 0x0000 - 0 = degC, 1 = degF
- Middle window (Mbpoll2nk.mbp):**
  - Row 1: 0x2022 - op code, format, cmd status
- Rightmost window (Mbpoll1nk3.mbp):**
  - Row 1: 0x0501 - status of PDI (600 and 601 are desirable)
  - Row 3: 0x2404 - Process data payload
- Configuration Dialogs (Read/Write Definition):**
  - Leftmost dialog:** Function: 16 Write Multiple Registers, Address: 1301.
  - Middle dialog:** Function: 03 Read Holding Registers (4x), Address: 1101.
  - Rightmost dialog:** Function: 03 Read Holding Registers (4x), Address: 1001.

**Text boxes in the image:**

- Top left: "This should be 0x0021 for a read and 0x0022 for a write." (with a green arrow pointing to the 0x0022 in the first window)
- Bottom left: "This is writing a 0 to index 91" (with a green arrow pointing to the 0x0001 in the first window)
- Bottom middle: "This is reading index 91" (with a green arrow pointing to the 0x005B in the second window)
- Bottom right: "This is reading process data from the device plugged into Port 1" (with a blue arrow pointing to the 0x0501 in the third window)

**Bottom text box:**

The leftmost window (the ISDU request) and the middle window (ISDU response) are continuously running when pictured here. In an actual installation, various schemes may occur such as: Run the request once, and then immediately poll for the response, check the data against a known placeholder, and retry until it is noted that the data changed. In this particular exercise, changes were made only in the far left window and then could be seen propagating to the middle window in typically < 1.0 sec.

- 6) Once you connect, the left side is used to send an IOlink ISDU command which can be “get” (op code 21) or “set” (op code 22). You can now send any (valid) value to the index.

