



Application Note for u-OS CODESYS NVL in Node-RED

Hardware reference

No.	Component name	Article No.	Hardware / Firmware version
1	UC20-WL2000-AC	1334950000	u-OS 2.0.0
2	IOT-GW30	2682620000	u-OS 2.0.0

Software reference

No.	Software name	Article No.	Software version
1	CODESYS Development System		3.5 SP19
2	CODESYS Runtime App		4.7.0.0-2
3	Node-RED (host network) App		3.0.2-1
4	node-red-contrib-nvl		1.2.0

Contact

Weidmüller Interface GmbH & Co. KG
Klingenbergstraße 26
32758 Detmold, Germany
www.weidmüller.com

For any further support please contact your
local sales representative:
<https://www.weidmüller.com/countries>

Content

1	Warning and Disclaimer.....	4
2	Introduction.....	5
3	Install required software and applications	6
3.1	CODESYS.....	6
3.2	Node-RED	6
3.3	NVL nodes.....	6
4	Set up NVLs	9
4.1	CODESYS as sender	9
4.2	Node-RED as receiver.....	12
4.3	Node-RED as sender.....	15
4.4	CODESYS as receiver.....	18
5	Test the communication.....	21
6	Example Code	22
6.1	CODESYS example sender Network Variable List	22
6.2	CODESYS example program	22
6.3	Node-RED flow.....	22

1 Warning and Disclaimer

Warning

Controls may fail in unsafe operating conditions, causing uncontrolled operation of the controlled devices. Such hazardous events can result in death and / or serious injury and / or property damage. Therefore, there must be safety equipment provided / electrical safety design or other redundant safety features that are independent from the automation system.

Disclaimer

This Application note / Quick Start Guide / Example Program does not relieve you of the obligation to handle it safely during use, installation, operation and maintenance. Each user is responsible for the correct operation of his control system. By using this Application note / Quick Start Guide / Example Program prepared by Weidmüller, you accept that Weidmüller cannot be held liable for any damage to property and / or personal injury that may occur because of the use.

Note

The given descriptions and examples do not represent any customer-specific solutions, they are simply intended to help for typical tasks. The user is responsible for the proper operation of the described products. Application notes / Quick Start Guides / Example Programs are not binding and do not claim to be complete in terms of configuration as well as any contingencies. By using this Application note / Quick Start Guide / Example Program, you acknowledge that we cannot be held liable for any damages beyond the described liability regime. We reserve the right to make changes to this application note / quick start guide / example at any time without notice. In case of discrepancies between the proposals Application notes / Quick Start Guides / Program Examples and other Weidmüller publications, like manuals, such contents have always more priority to the examples. We assume no liability for the information contained in this document. Our liability, for whatever legal reason, for damages caused using the examples, instructions, programs, project planning and performance data, etc. described in this Application note / Quick Start Guide / Example is excluded.

Security notes

In order to protect equipment, systems, machines and networks against cyber threats, it is necessary to implement (and maintain) a complete state-of-the-art industrial security concept. The customer is responsible for preventing unauthorized access to his equipment, systems, machines and networks. Systems, machines and components should only be connected to the corporate network or the Internet if necessary and appropriate safeguards (such as firewalls and network segmentation) have been taken.

2 Introduction

This application note addresses developers with basic familiarity with u-OS, Node-RED and the CODESYS development system.

The application note explains how to set up a network variable exchange between the CODESYS runtime on the one hand and Node-RED on the other hand on a Weidmüller u-OS device. The application note discusses how to set up both Node-RED and a CODESYS application to act as both receiver and sender of network variables.

The example projects shown run the CODESYS runtime and Node-RED on the same device, a UC20-WL2000-AC with u-OS.

The default UDP port for NVL data is 1202. Because the example projects shown in this application note act as sender and receiver simultaneously in both the CODESYS application and the Node-RED flow, we need to configure one network variable connection to use another port than 1202. Accordingly, the examples “Node-RED as sender” and “CODESYS as receiver” use port 1203.

3 Install required software and applications

Make sure that your device runs the latest u-OS. Update if needed. Then install the CODESYS development system on your PC, the CODESYS runtime on your u-OS device, Node-RED on your u-OS device and the network variable list nodes in Node-RED.

3.1 CODESYS

Follow the Weidmüller application note AN0099 to install the CODESYS runtime on your Weidmüller u-OS device and the CODESYS development system on your PC.

3.2 Node-RED

Follow the Weidmüller application note AN0100 to install the Node-RED (host network) app on your Weidmüller u-OS device.

3.3 NVL nodes

1. Open Node-RED on your u-OS device.
2. In Node-RED's burger menu, click "manage palette", see Figure 1: Node-RED's burger menu. This opens the palette view in Node-RED's user settings.

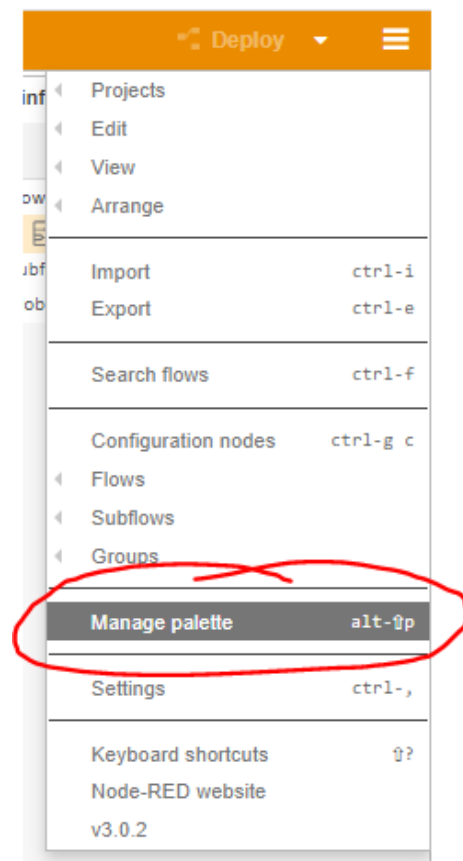


Figure 1: Node-RED's burger menu

3. In the palette view, open the “Install” tab, see Figure 2: Node-RED's palette view.
4. Enter “NVL” into the search field.
5. Click “Install” of the node-red-contrib-nvl node.

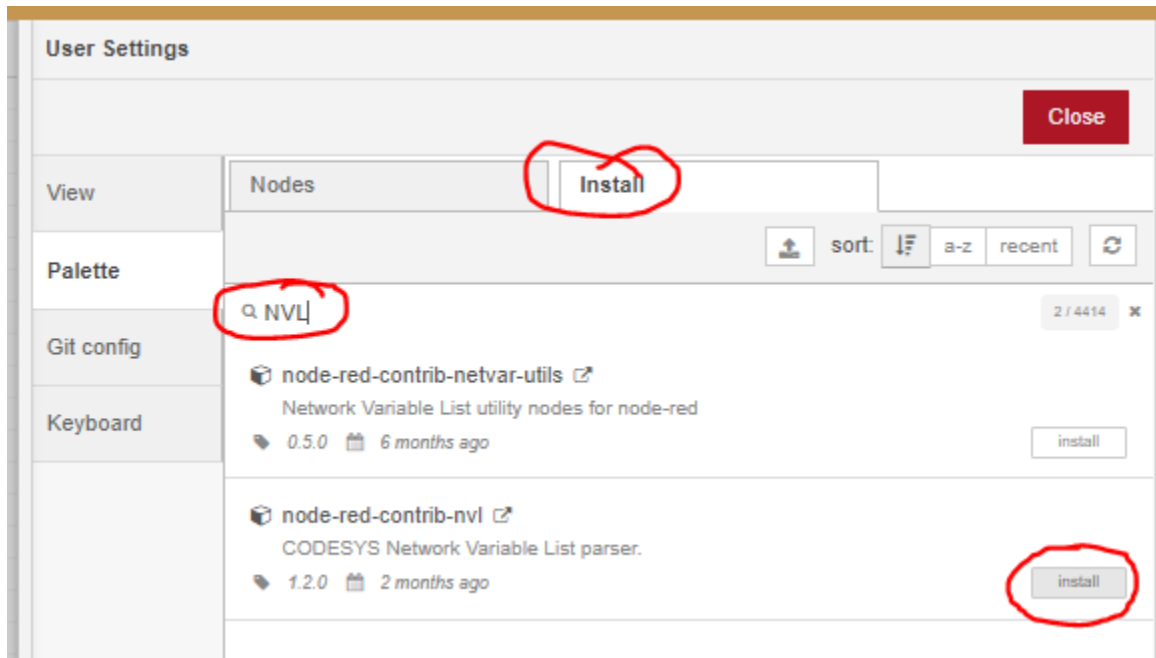


Figure 2: Node-RED's palette view

Node-RED pops up a dialog box, see Figure 3: Installing 'node-red-contrib-nvl'.

6. Click “Install”, in that dialog box.
7. Wait, the installation takes some time.

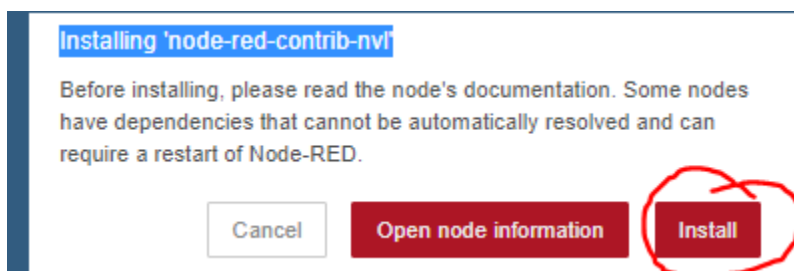


Figure 3: Installing 'node-red-contrib-nvl'

8. After the installation has completed, check that the NVL nodes exist in the palette in the category “parser”, see Figure 4: NVL - receive and NVL - send.

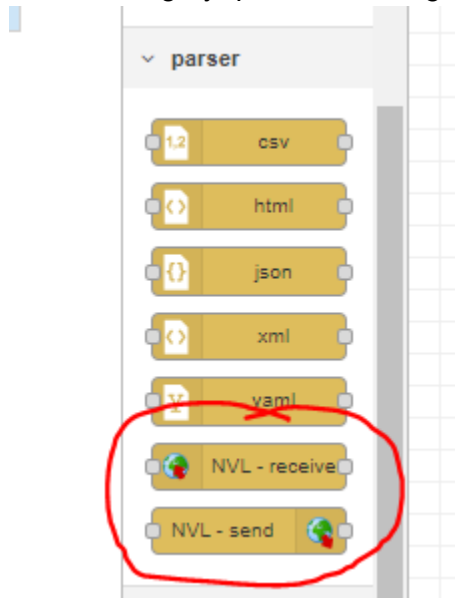


Figure 4: NVL - receive and NVL - send

4 Set up NVLs

4.1 CODESYS as sender

CODESYS configures Network Variable transmission via a Network Variable List (Sender) child object of the application in the Devices tree. To add and configure such an object, follow the steps described below. The CODESYS Development system offers an export of the sender NVL into a file. Import this file into the receiver to keep it coordinated with your NVL sender CODESYS project.

1. In your CODESYS project, right-click on the application, select Add Object -> Network Variable List (Sender)..., as shown in Figure 5: add NVL (Sender).

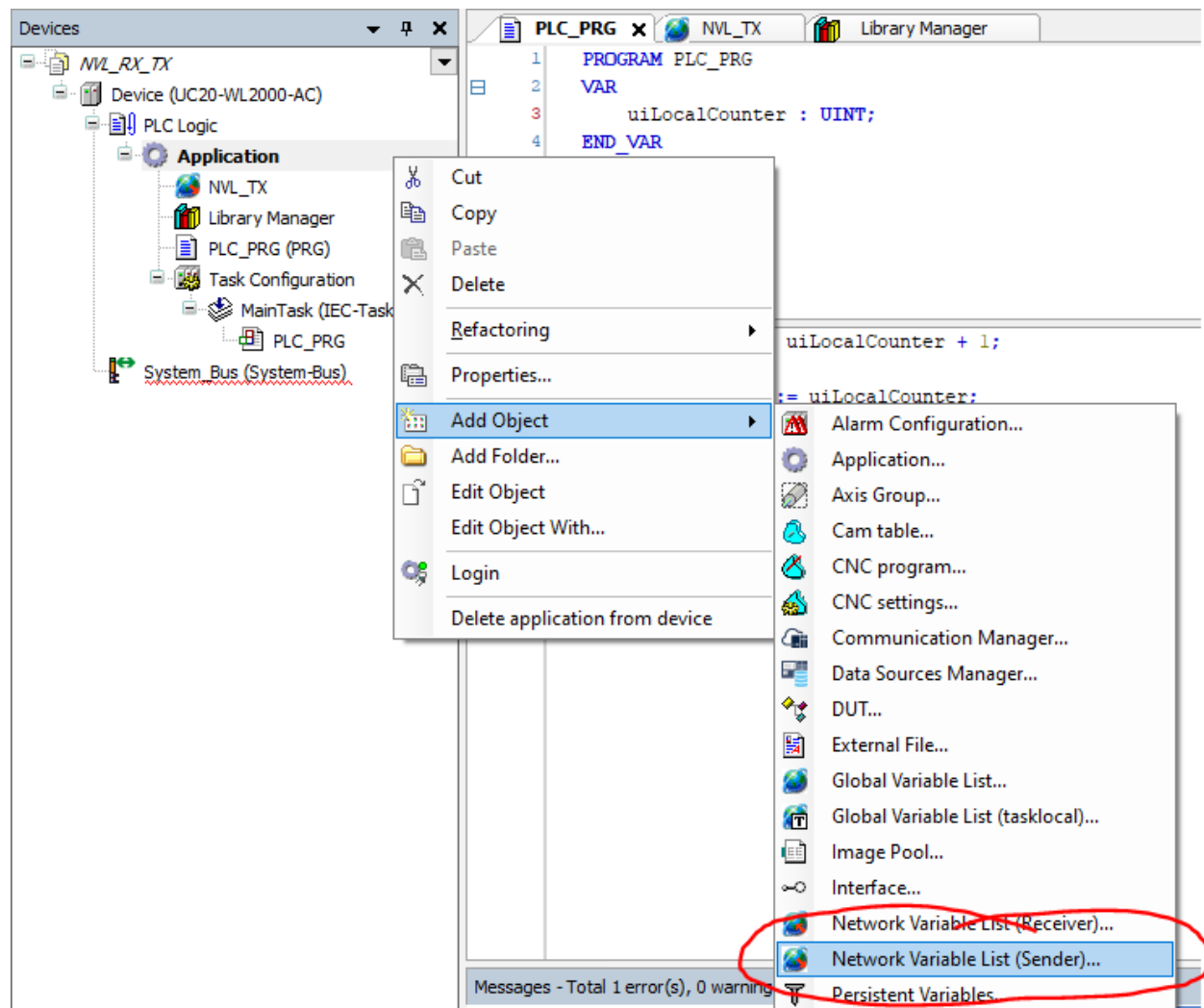


Figure 5: add NVL (Sender)

2. In the “Add Network Variable List (Sender)” dialogue,
 - a. enter a name for the Network Variable List (Sender) object, e.g. “NVL_TX”.
 - b. Choose “UDP” as network type.
 - c. Click “Settings”
 - d. Change the Broadcast address from 255.255.255.255 to 127.0.0.1
By doing this, CODESYS will broadcast the NVL only locally on the device, not in the entire network.
 - e. Click “Add”.

Figure 6: Add Network Variable List (Sender) and Figure 7: Properties NVL_TX local broadcast address show this.

The screenshot shows the 'Add Network Variable List (Sender)' dialog box. At the top, it says 'Create a global variable list to send via a network (Use object properties to edit settings)'. Below this, the 'Name' field is filled with 'NVL_TX'. The 'Network type' dropdown is set to 'UDP'. The 'Task' dropdown is set to 'MainTask'. The 'Listidentifier' field contains the number '1'. There are three unchecked checkboxes: 'Pack variables', 'Transmit checksum', and 'Acknowledgement'. There are three checked/unchecked options for transmission: 'Cyclictransmission' is checked with an interval of 'T#50ms', 'Transmit on change' is unchecked with a minimum gap of 'T#20ms', and 'Transmit on event' is unchecked. At the bottom, there are 'Add' and 'Cancel' buttons. The 'Add' button is highlighted with a blue border.

Figure 6: Add Network Variable List (Sender)

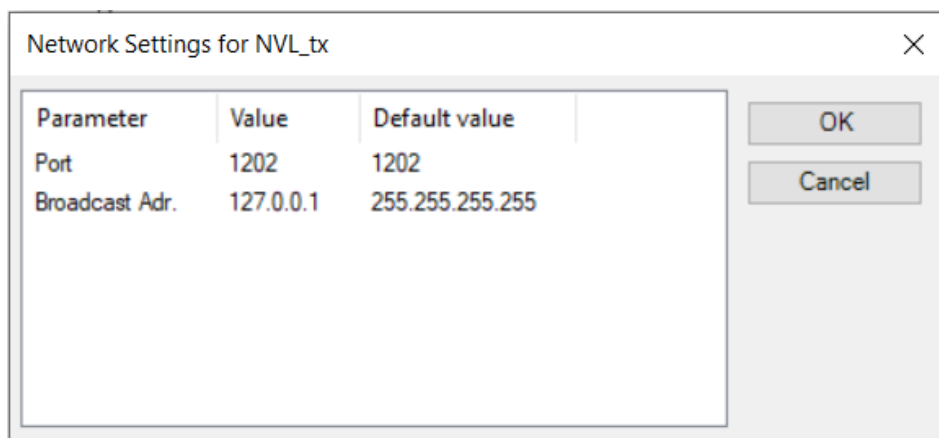


Figure 7: Properties NVL_TX local broadcast address

3. Double-click the Network Variable List (Sender) object in your device tree.
4. In the object's editor tab, add some variables.

Figure 8: add variables to NVL shows this.

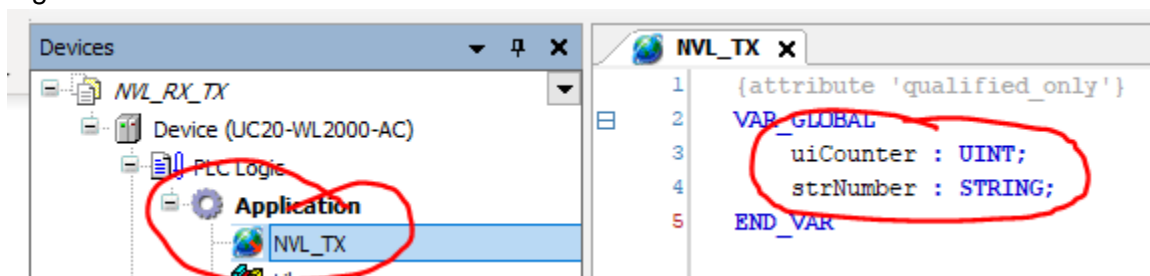


Figure 8: add variables to NVL

5. Right-click on the Network Variable List (Sender) object in your device tree and select “Properties”.
6. Open the “Link To File” tab and enter a file name.
7. Click on “OK”.

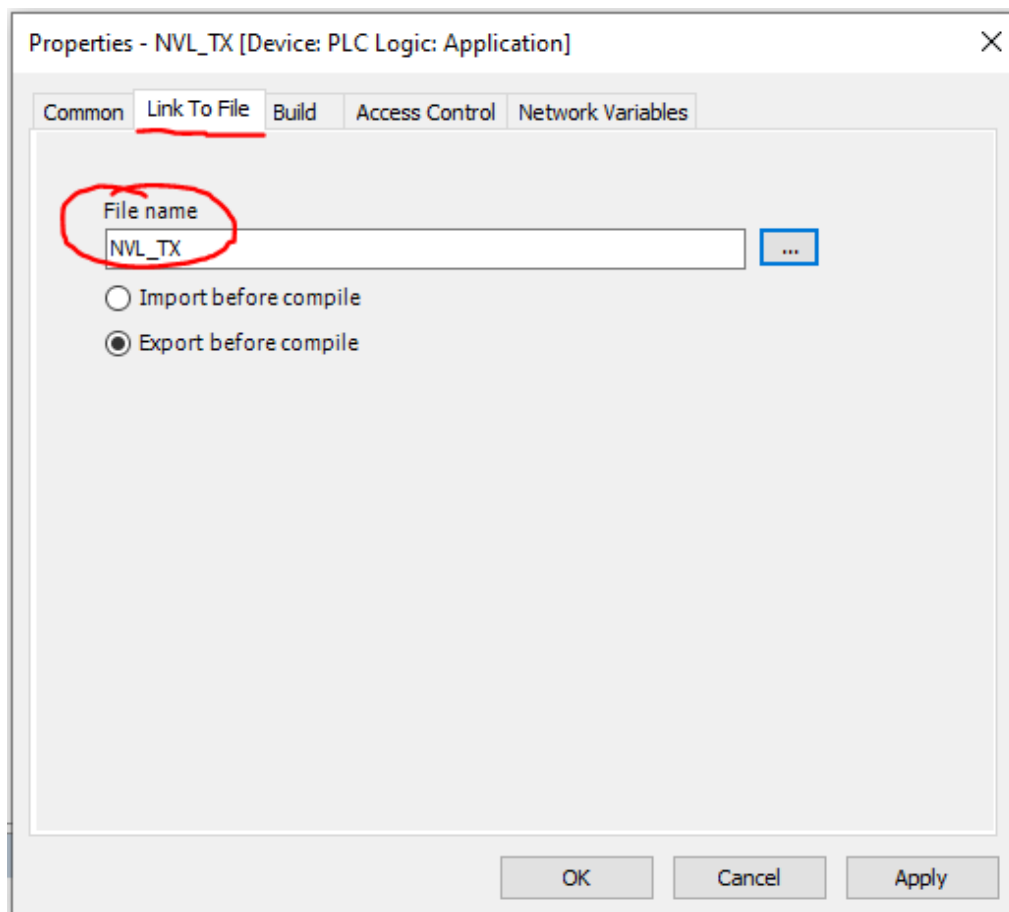


Figure 9: Properties NVL_TX

4.2 Node-RED as receiver

Node-RED receives the NVL data via a udp in node and passes this data as a buffer into the NVL-receive node. The NVL-receive node expects this buffer as input and outputs a JSON object with the received network variables.

1. Assemble a flow as shown in Figure 10: NVL receive flow in Node-RED.



Figure 10: NVL receive flow in Node-RED

2. Double-click the NVL-receive node and click “import”, as shown in Figure 11: edit NVL-receive node.

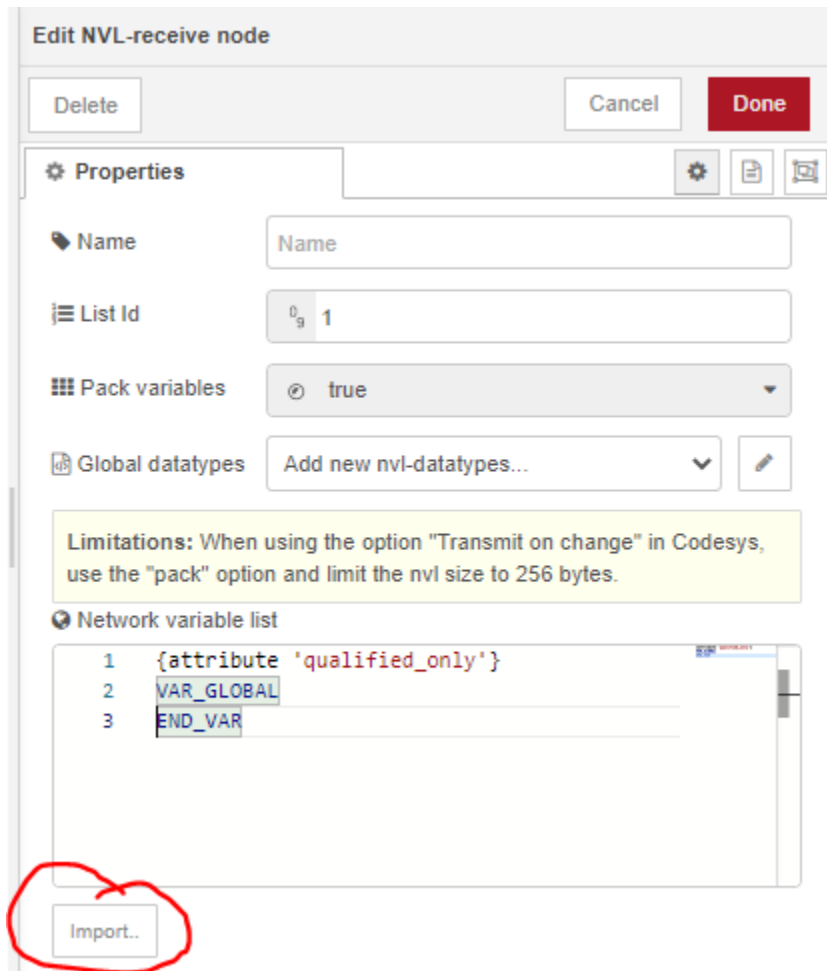


Figure 11: edit NVL-receive node

3. Select and open the NVL export file you created when setting up CODESYS as sender, see Figure 9: Properties NVL_TX.

Node-RED will read and parse the NVL export file and fill-in the properties in the NVL-receive node, accordingly.

4. Double-click the udp in node and enter the port number 1202. See Figure 12: Edit udp in node.

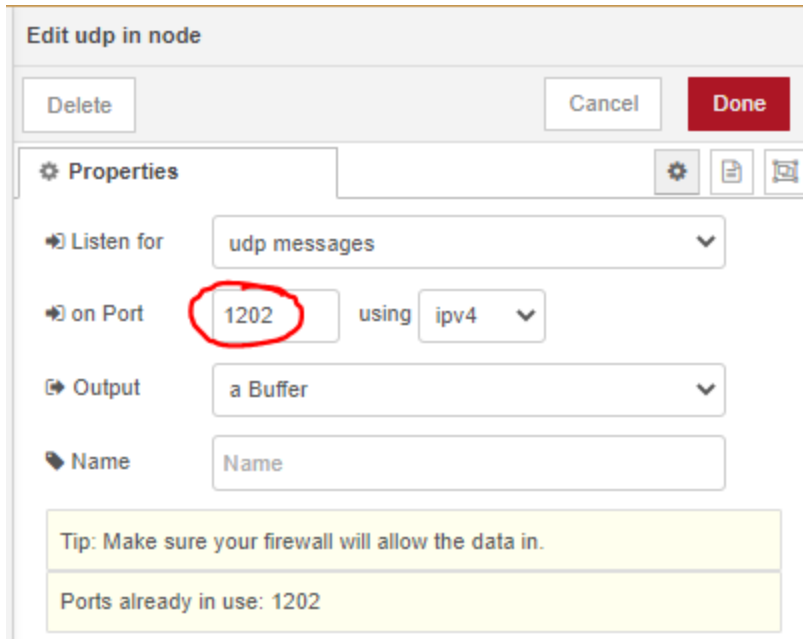


Figure 12: Edit udp in node

5. Deploy the flow
6. Watch the debug output while a CODESYS application on the same u-OS device sends NVL data.

4.3 Node-RED as sender

Node-RED's NVL-send node expects a JSON object with the network variable to send as input. The NVL-send node outputs the NVL content as a buffer. The flow then sends the NVL data in said buffer via a udp out node. The udp out node specifies the target address and the UDP port to use. We use an inject node to generate a JSON object with the network variable "uiDemo" and pass it to the NVL-send node.

1. Create the flow as shown in Figure 13: Node-RED flow as NVL sender in Node-RED.

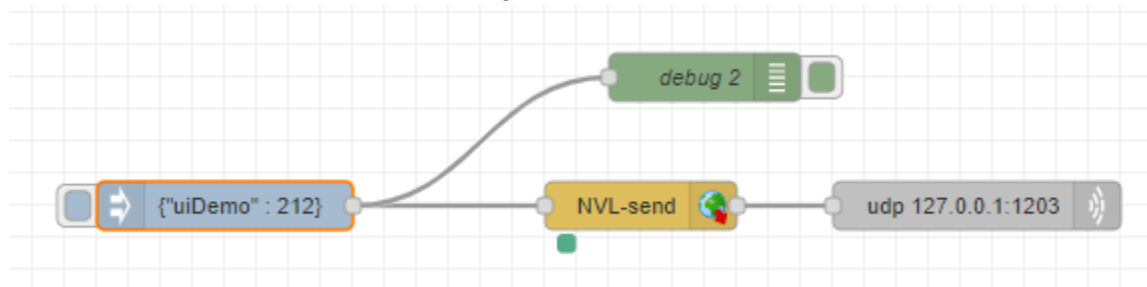


Figure 13: Node-RED flow as NVL sender

2. Double-click the inject node and select JSON instead of timestamp for the msg.payload, as shown in Figure 14: inject payload.

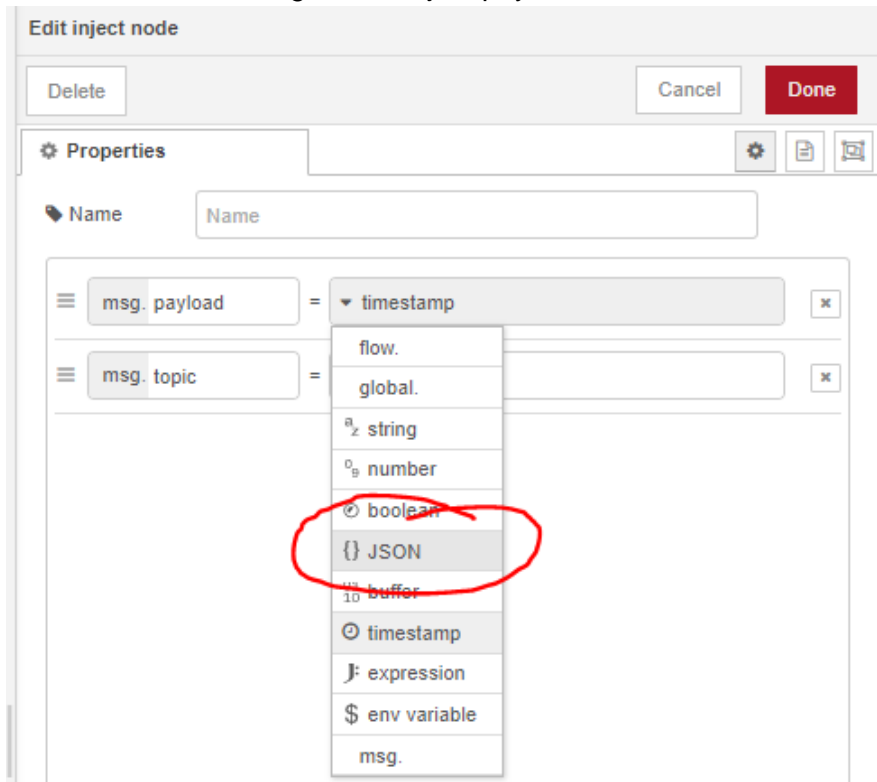


Figure 14: inject payload

3. Input the JSON payload {"uiDemo" : 123} as shown in Figure 15: inject JSON payload.

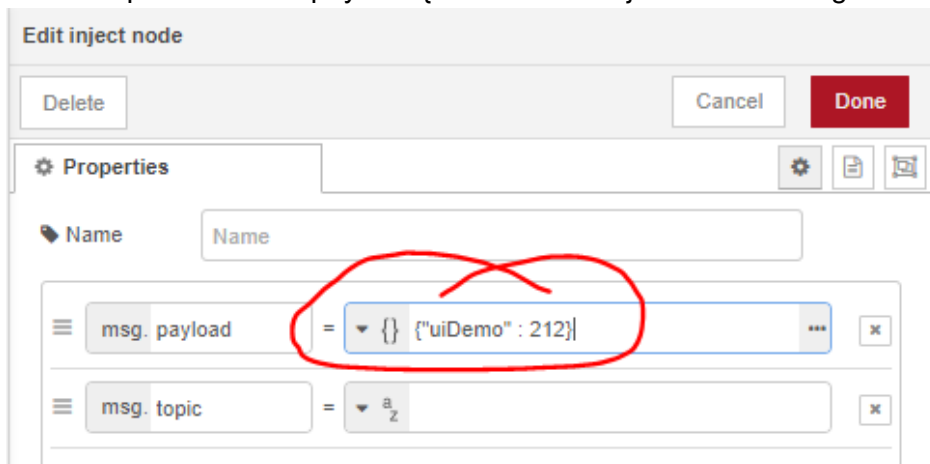


Figure 15: inject JSON payload

4. Click "Done".
5. Double-click the NVL-send node and enter List Id, UDP Port and Network variable list as shown in Figure 16: Edit NVL-send node.
6. Click Export.. and save the .gvl file.
7. Click Done.

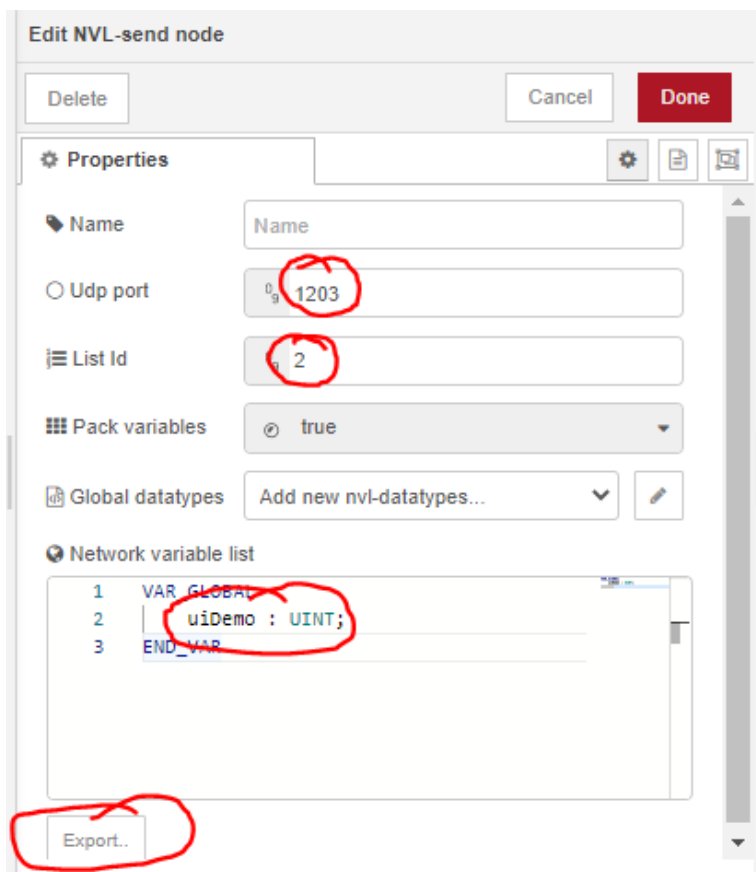


Figure 16: Edit NVL-send node

8. Double-click the udp out node and enter the target address and UDP port as shown in Figure 17: Edit udp out node.

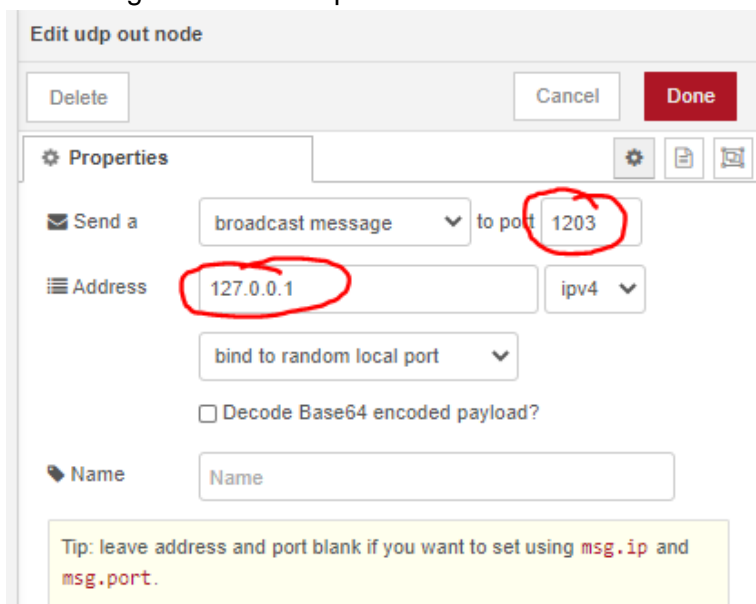


Figure 17: Edit udp out node

9. Click Done.

4.4 CODESYS as receiver

CODESYS configures Network Variable reception via a Network Variable List (Receiver) child object of the application in the Devices tree. The Network Variable List (Receiver) object expects to import a .gvl file created by a Network Variable sender. To add and configure a Network Variable List (Receiver) object to your CODESYS project, follow these steps:

1. In your CODESYS project, right-click on the application and select Add Object -> Network Variable List (Receiver)..., as shown in Figure 18: CODESYS Network Variable List (Receiver).

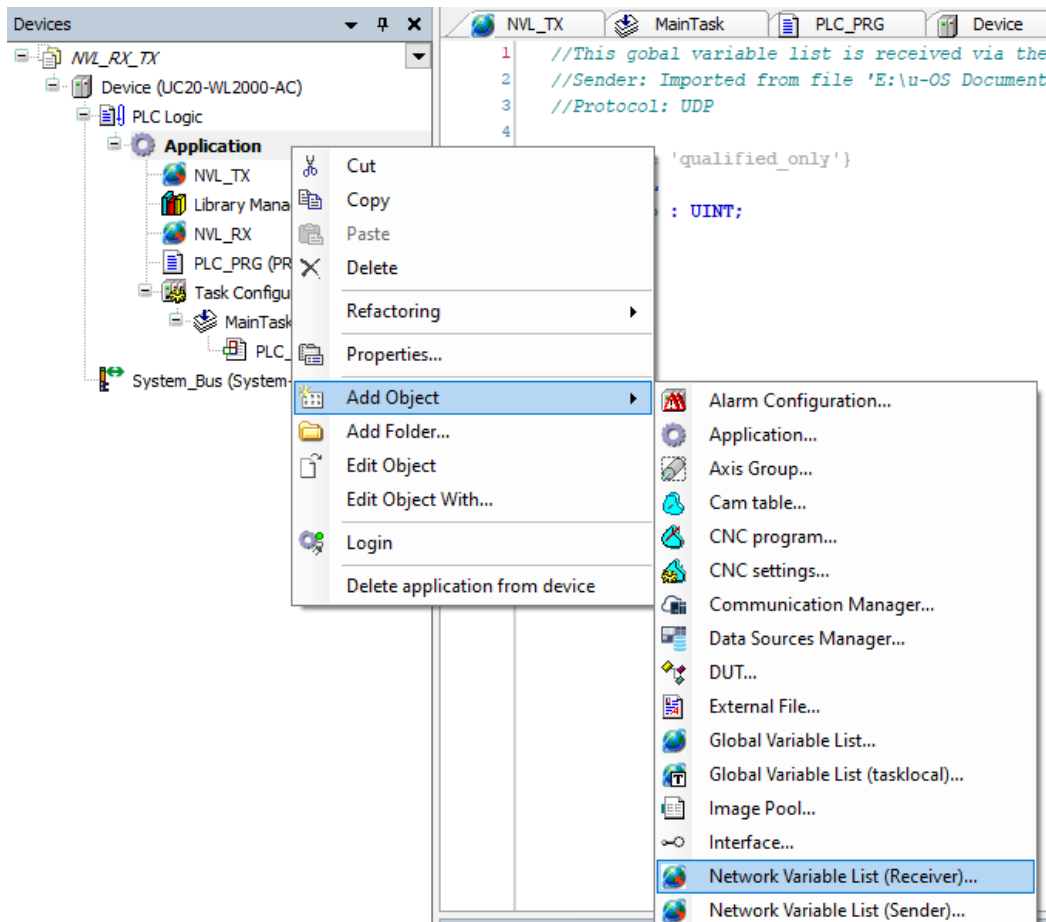


Figure 18: CODESYS Network Variable List (Receiver)

2. In the Network Variable List (Receiver) dialogue, enter a name for the Network Variable List (Receiver) Object.
3. In the Network Variable List (Receiver) dialogue, choose the file to import from.

See Figure 19: Add Network Variable List (Receiver).

Hint: For this example, the file to import from is the file you exported from Node-RED in the chapter “Node-RED as sender”.

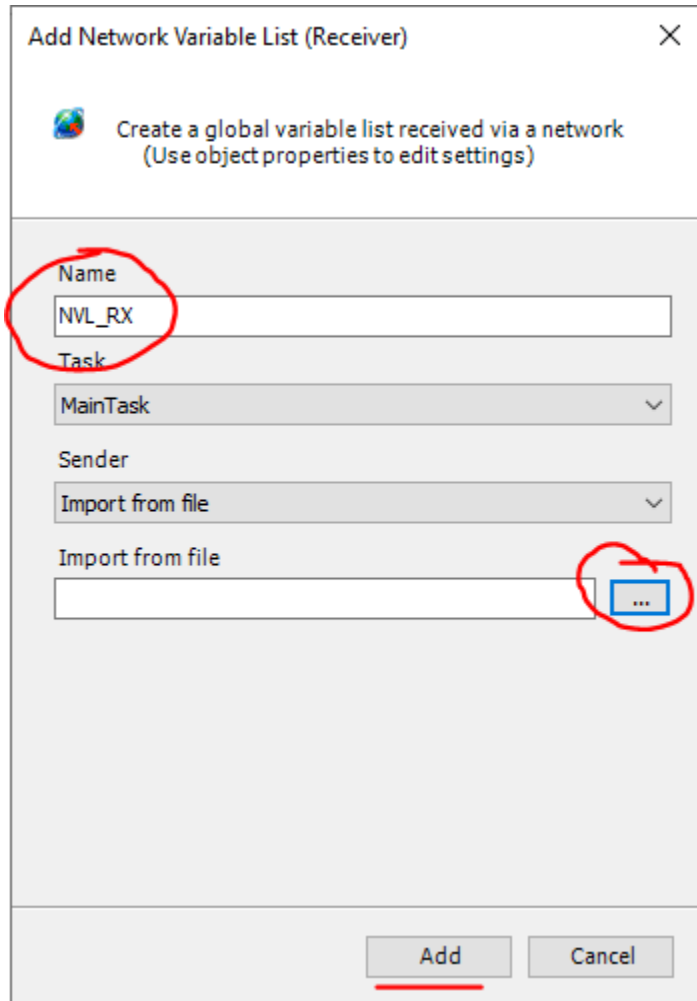


Figure 19: Add Network Variable List (Receiver)

4. Reference the received Network Variable in your application by preceding its name with the name of the Network Variable List (Receiver) object, separated by a dot. E.g. for the “uiDemo” Network Variable exported from Node-RED this is NVL_RX.uiDemo, as shown in Figure 20: example code.

```
1  ● uiLocalCounter 664 := uiLocalCounter 664 + 1;  
2  
3  ● NVL_TX.uiCounter 664 := uiLocalCounter 664;  
4  ● NVL_TX.strNumber 664 := ANY_TO_STRING(uiLocalCounter 664);  
5  
6  ● uiLocalDemo 212 := NVL_RX.uiDemo 212;  
7  ● RETURN
```

Figure 20: example code

5 Test the communication

To transmit and receive network variables with Node-RED, it is sufficient to set up the udp in/out - and NVL-send / NVL-receive nodes in Node-RED and deploy the flow.

Similarly, it is sufficient to add and configure the Network Variable List (Sender) and Network Variable List (Receiver) objects in the CODESYS application. After you have built and downloaded it, the CODESYS application will transmit and receive network variables. For a simple test, follow the steps described in chapters 4.1 - 4.4. Deploy the Node-RED flow. Generate, build and download the CODESYS application. Log in into the CODESYS runtime on your u-OS device and start the application.

In Node-RED, switch on the Debug nodes and observe the incoming network variable data.

In CODESYS, watch the received network variable, then toggle the inject node in the Node-RED-as-sender-flow in Node-RED and observe the value change in the CODESYS application.

6 Example Code

6.1 CODESYS example sender Network Variable List

```
{attribute 'qualified_only'}  
VAR_GLOBAL  
    uiCounter : UINT;  
    strNumber : STRING;  
END_VAR
```

6.2 CODESYS example program

```
PROGRAM PLC_PRG  
VAR  
    uiLocalCounter : UINT;  
    uiLocalDemo : UINT;  
END_VAR  
  
uiLocalCounter := uiLocalCounter + 1;  
  
NVL_TX.uiCounter := uiLocalCounter;  
NVL_TX.strNumber := ANY_TO_STRING(uiLocalCounter);  
  
uiLocalDemo := NVL_RX.uiDemo;
```

6.3 Node-RED flow

You should have received the Node-RED flow together with this document as a .json file