

How to...

## echochange

Setting up data exchange between a Siemens S7-300 and an Allen-Bradley ControlLogix, using Siemens PUT and GET functions



Version: E-032014-01

## **Disclaimer of liability**

The information contained in these instructions corresponds to the technical status at the time of printing of it and is passed on with the best of our knowledge. The information in these instructions is in no event a basis for warranty claims or contractual agreements concerning the described products, and may especially not be deemed as warranty concerning the quality and durability pursuant to Sec. 443 German Civil Code. We reserve the right to make any alterations or improvements to these instructions without prior notice. The actual design of products may deviate from the information contained in the instructions if technical alterations and product improvements so require.

It may not, in part or in its entirety, be reproduced, copied, or transferred into electronic media.

## **Softing Industrial Automation GmbH**

Richard-Reitzner-Allee 6  
85540 Haar / Germany  
Tel: + 49 89 4 56 56-0  
Fax: + 49 89 4 56 56-488  
Internet: <http://industrial.softing.com>  
Email: [info.automation@softing.com](mailto:info.automation@softing.com)  
Support: [support.automation@softing.com](mailto:support.automation@softing.com)

The latest version of this manual is available in the Softing download area at: <http://industrial.softing.com>.

# Table of Contents

|                  |  |           |
|------------------|--|-----------|
| <b>Chapter 1</b> | <b>Introduction.....</b>                                       | <b>5</b>  |
| <b>Chapter 2</b> | <b>RSLogix 5000- create tags to send and receive data.....</b> | <b>6</b>  |
| <b>Chapter 3</b> | <b>NetCon echo – echochange network settings .....</b>         | <b>8</b>  |
| <b>Chapter 4</b> | <b>STEP 7 setup.....</b>                                       | <b>9</b>  |
| <b>Chapter 5</b> | <b>NetCon - establish the connections.....</b>                 | <b>19</b> |

This page is intentionally left blank.

# 1 Introduction

This document provides step by step instructions on how to establish data exchange between a Siemens S7-300 PLC and an Allen-Bradley ControlLogix PLC, using a Softing echochange protocol converter.

General description of the presented example:

- Both PLCs and the echochange are in the same network
- The S7-300 controls the flow of communications, using PUT and GET functions
- The PUT function is used to write data from a S7-300 data block (**Data\_to\_CLX**, 1 byte) onto a ControlLogix tag (**Data\_from\_S7**)
- The GET function is used to read data from a ControlLogix tag (**Data\_to\_S7**, integer array of 50 elements) and store it onto an S7-300 data block (**Data\_from\_CLX**)

Software used:

- Siemens SIMATIC Step 7 V5.4 SP5
- Allen-Bradley RSLogix 5000 V20.01.00 (CPR 9 SR 5)
- Allen-Bradley RSLinx Classic Lite V2.59.02 (CPR 9 SR 5)
- Softing NetCon echo V4.33

## Conventions

The following conventions are used throughout Softing customer documentation:

Keys, buttons, menu items, commands and other elements involving user interaction are set in bold font and menu sequences are separated by an arrow

Open **Start** → **Control Panel** → **Programs**

Buttons from the user interface are enclosed in brackets and set to bold typeface

Press **[Start]** to start the application

Coding samples, file extracts and screen output is set in Courier font type

MaxDlsapAddressSupported=23

Filenames and directories are written in italic

Device description files are located in C:  
*\StarterKit\delivery\software\Device  
Description files*

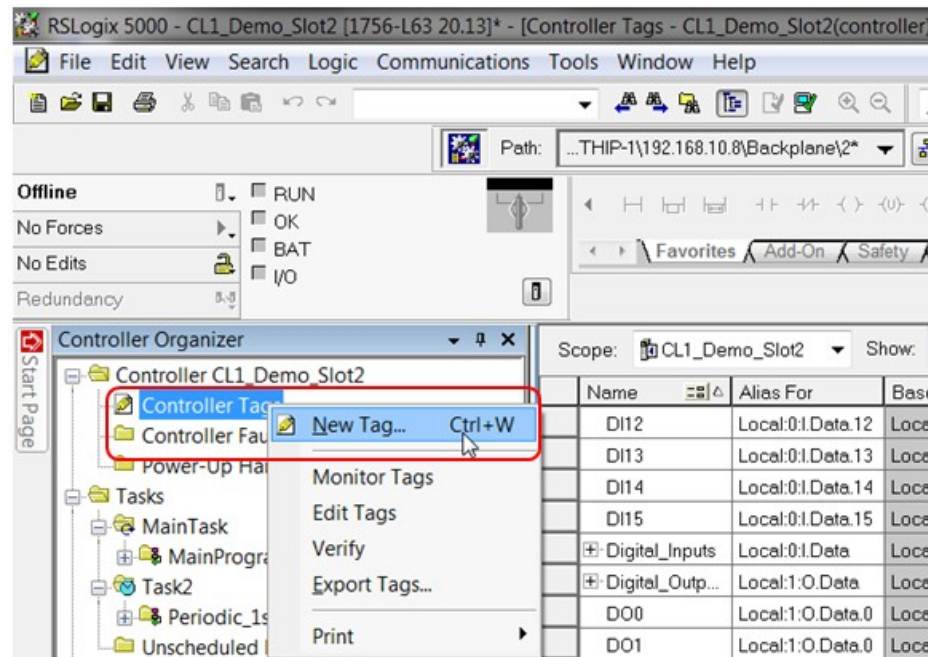


### Note

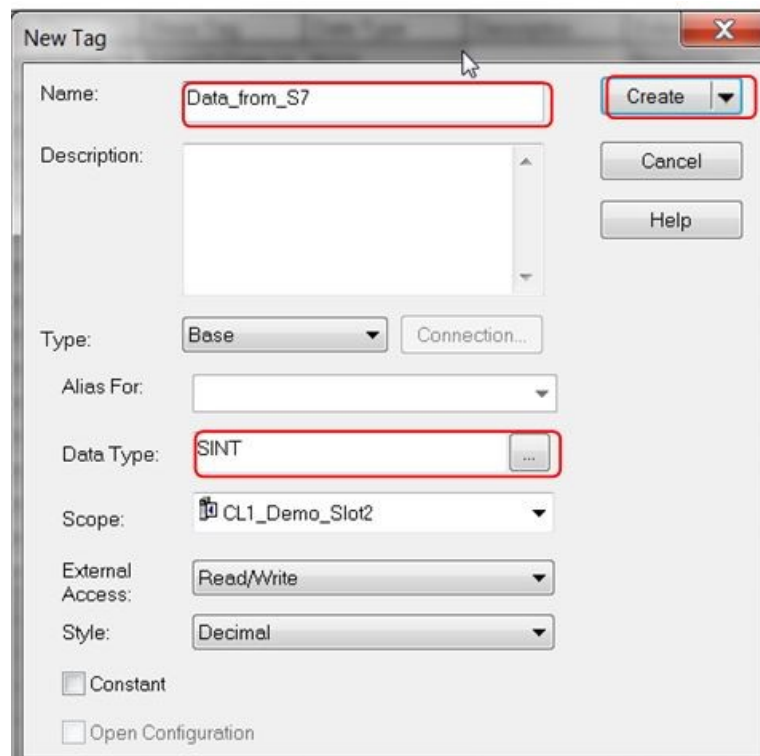
This symbol is used to call attention to notable information that should be followed during installation, use, or servicing of this device.

## 2 RSLogix 5000- create tags to send and receive data

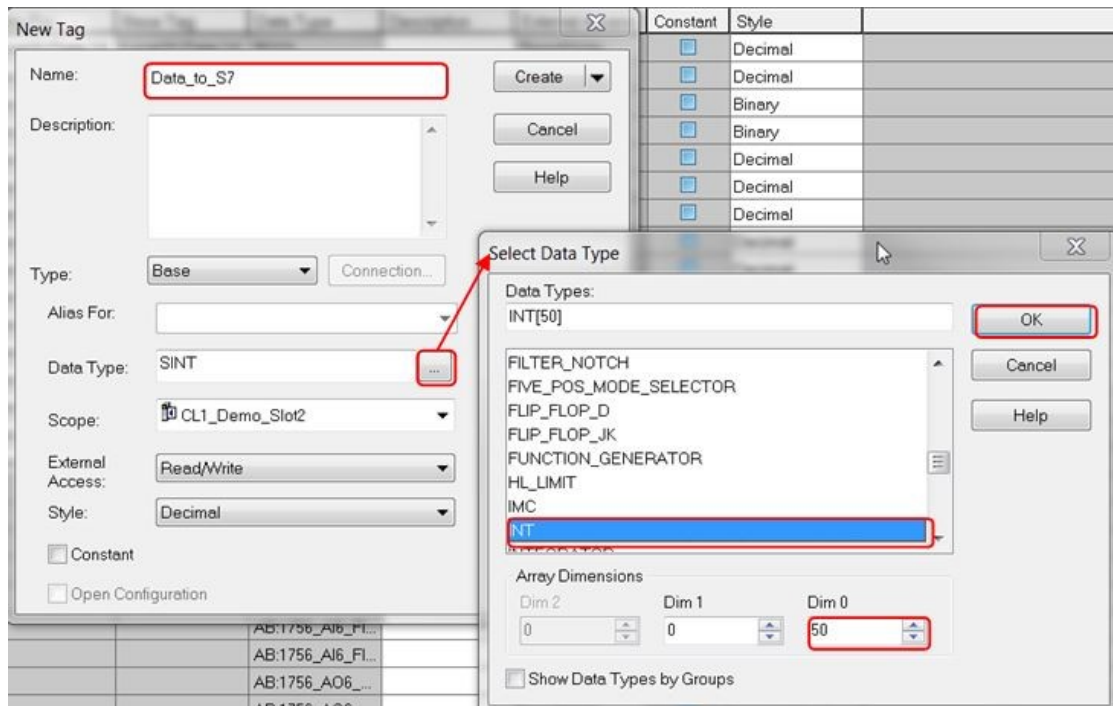
1. Create a tag to receive data from the S7-300 PLC. In this example we will create a SINT tag (short integer, 1 byte).
  - a. In the **Controller Organizer** section, right-click on **Controller Tags** and then click on **New Tag...**



- b. Enter a tag name, select data type **SINT** and click **Create**.



- Use the previous method to create a tag for the S7-300 to read. We will create an INT (integer) array with 50 elements.



- This is how the new tags should look in the **Controller Tags** table:

| Scope: CL1_Demo_Slot2 Show: All Tags |                   |                   |             |             |                 |                                     |         |  |
|--------------------------------------|-------------------|-------------------|-------------|-------------|-----------------|-------------------------------------|---------|--|
| Name                                 | Alias For         | Base Tag          | Data Type   | Description | External Access | Constant                            | Style   |  |
| AI0                                  | Local:3:I.Ch0Data | Local:3:I.Ch0Data | REAL        |             | Read/Write      | <input checked="" type="checkbox"/> | Float   |  |
| AI1                                  | Local:3:I.Ch1Data | Local:3:I.Ch1Data | REAL        |             | Read/Write      | <input checked="" type="checkbox"/> | Float   |  |
| AO0                                  | Local:4:O.Ch0D... | Local:4:O.Ch0D... | REAL        |             | Read/Write      | <input checked="" type="checkbox"/> | Float   |  |
| AO1                                  | Local:4:O.Ch1D... | Local:4:O.Ch1D... | REAL        |             | Read/Write      | <input checked="" type="checkbox"/> | Float   |  |
| BitCounter                           |                   |                   | COUNTER(10) |             | Read/Write      | <input type="checkbox"/>            |         |  |
| Data_from_S7                         |                   |                   | SINT        |             | Read/Write      | <input type="checkbox"/>            | Hex     |  |
| Data_to_S7                           |                   |                   | INT[50]     |             | Read/Write      | <input type="checkbox"/>            | Decimal |  |
| DI0                                  | Local:0:I.Data 0  | Local:0:I.Data 0  | BOOL        |             | Read/Write      | <input checked="" type="checkbox"/> | Decimal |  |

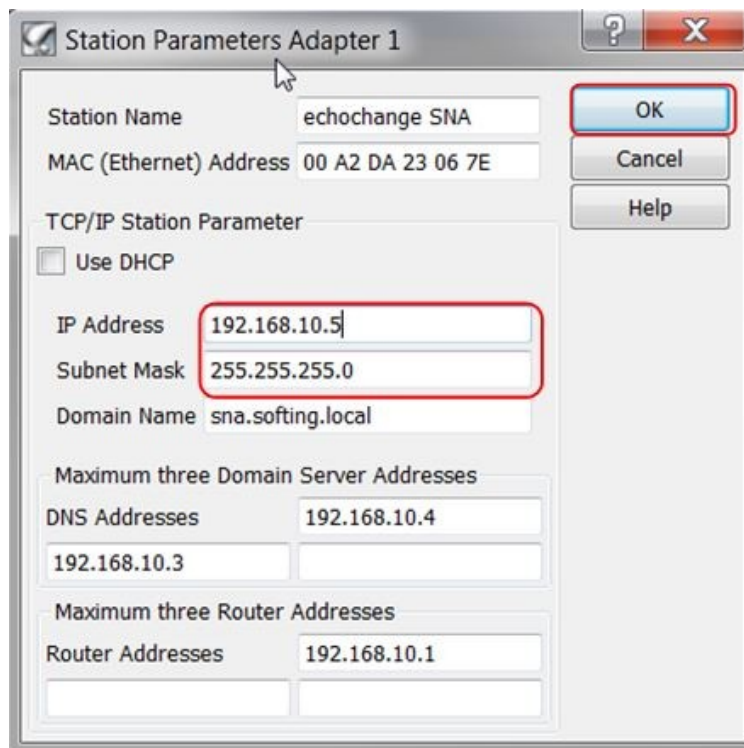
- Download the project to the ControlLogix device.

### 3 NetCon echo – echochange network settings

1. Configure the network settings of the echochange port to be used. We will use port 1 (Eth 1).
  - a. Click on **Station** → **Own Station 1**.



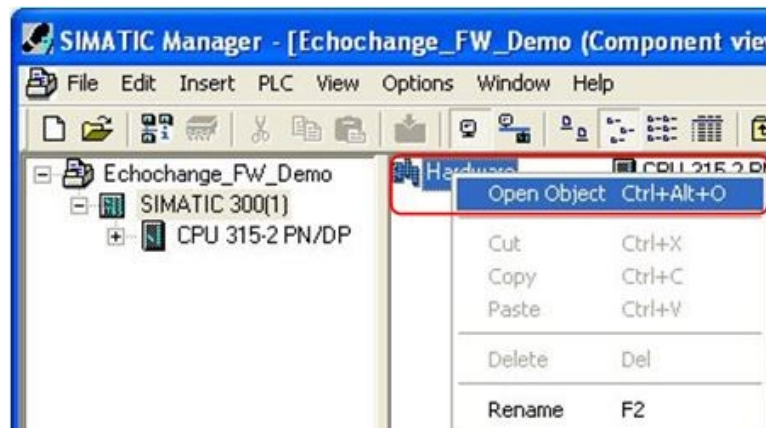
- b. As a minimum, the IP address and subnet mask are required. DNS and Router addresses may be necessary in some cases.



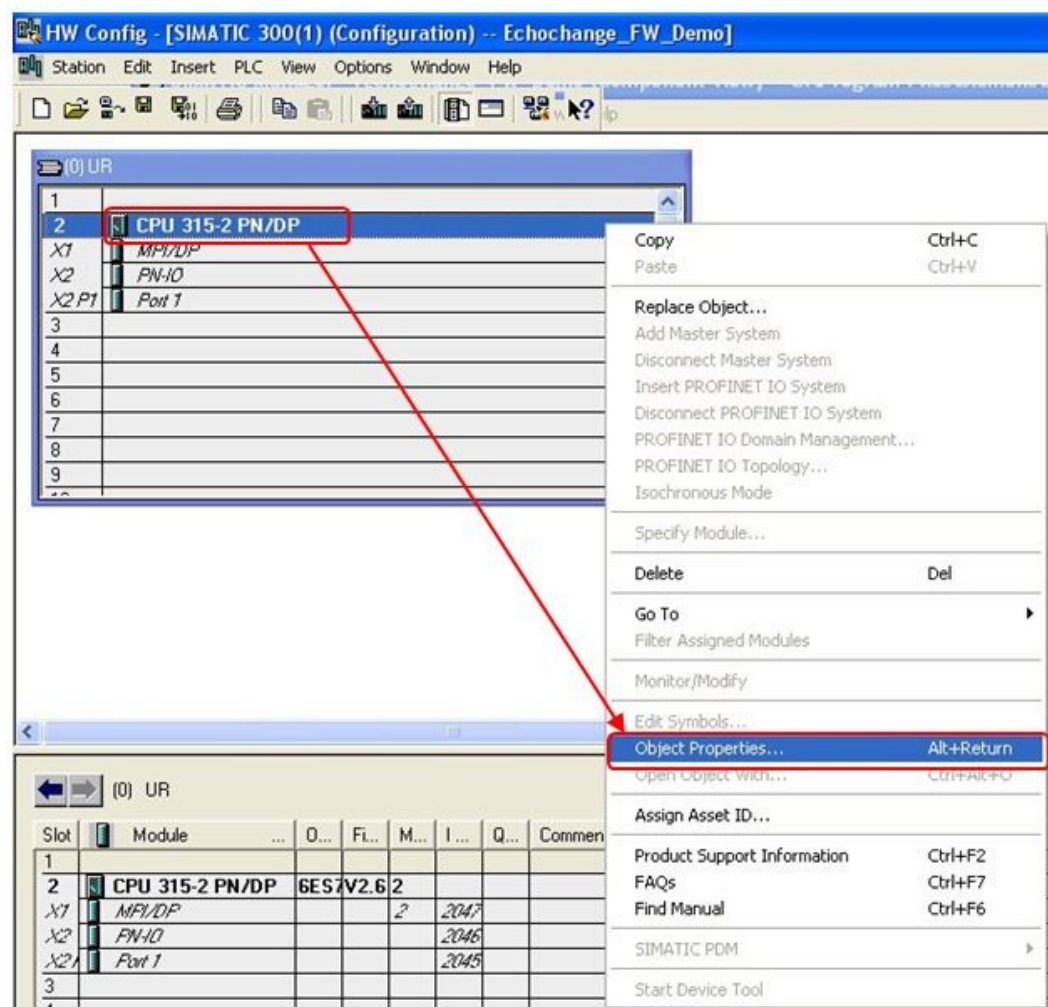


## 4 STEP 7 setup

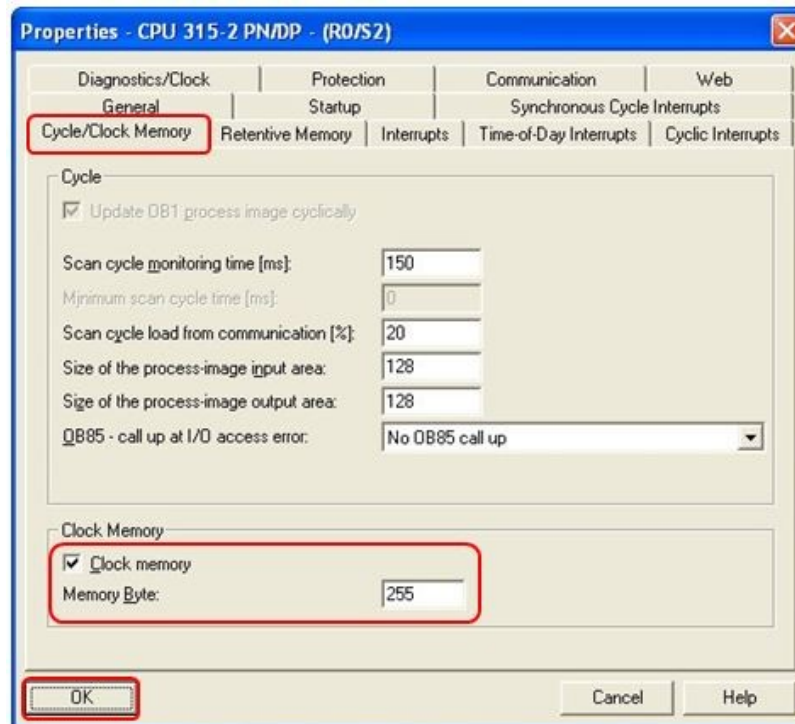
1. Open or create your project.
2. Enable the clock memory byte, as we will need a cyclic trigger.
  - a. Open the hardware configuration.



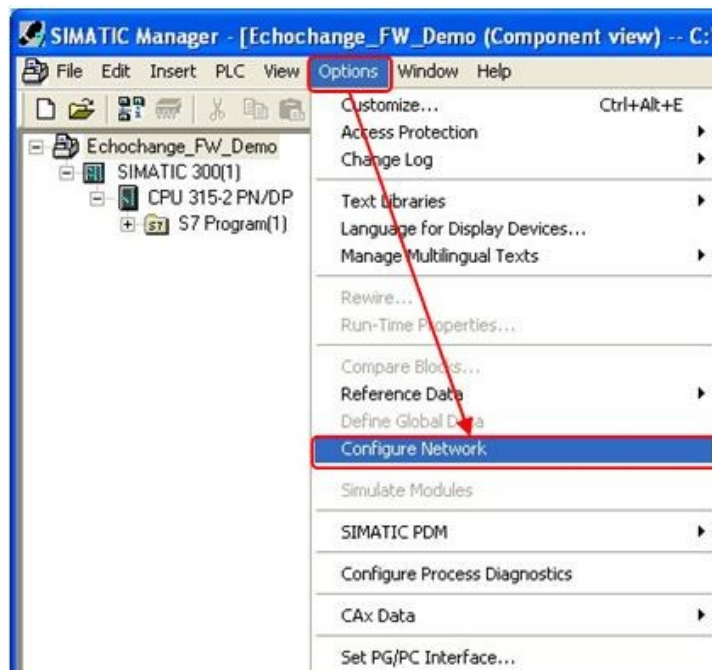
- b. Right-click on the CPU and select **Object Properties...**



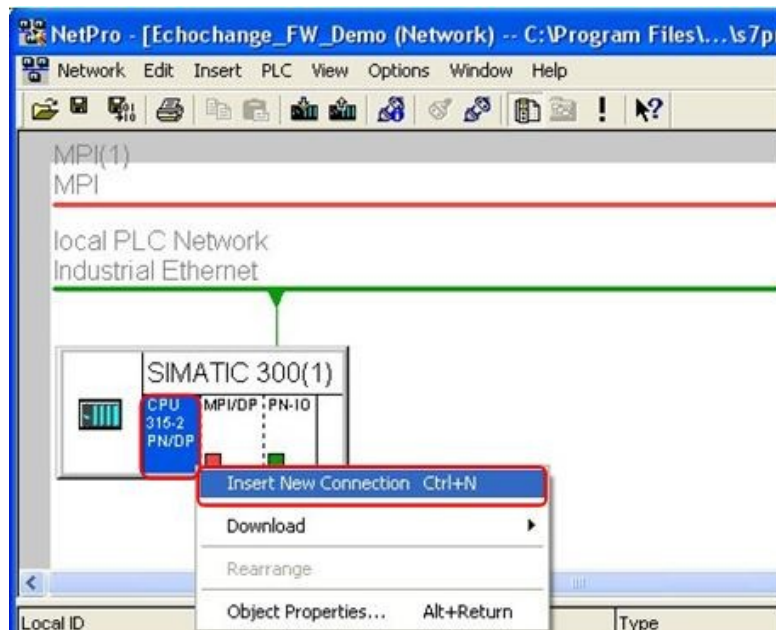
- c. Select the **Cycle/Clock Memory** tab to make sure **Clock memory** is checked, and choose an unused memory byte. We use MB255 in this example.



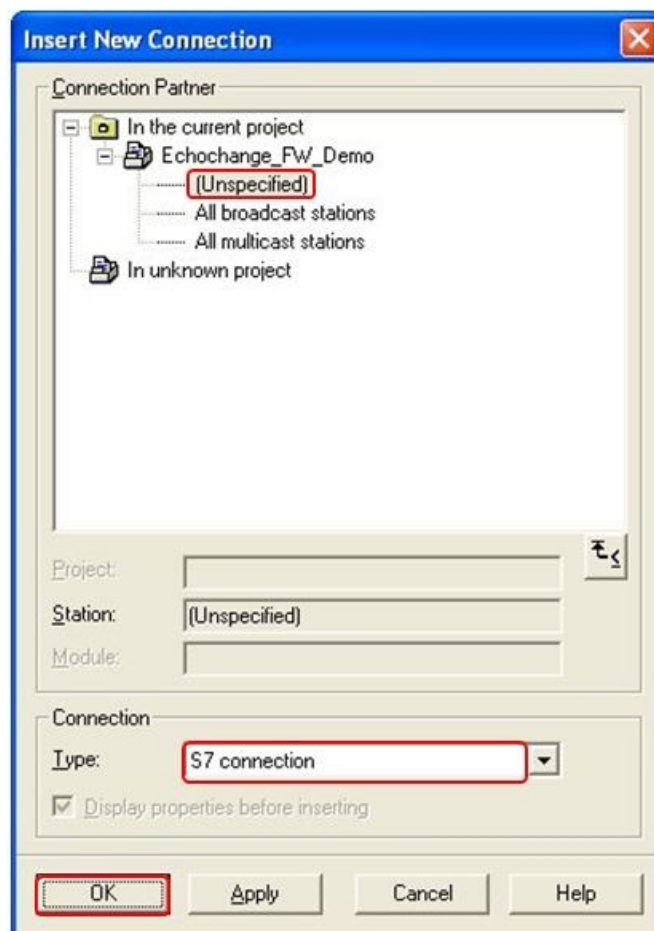
3. Create a new connection.
  - a. Open NetPro by clicking on **Options** → **Configure Network**.



- b. Right-click on the CPU block and choose **Insert New Connection**.



- c. Select **Unspecified** station, and **S7 connection** type.

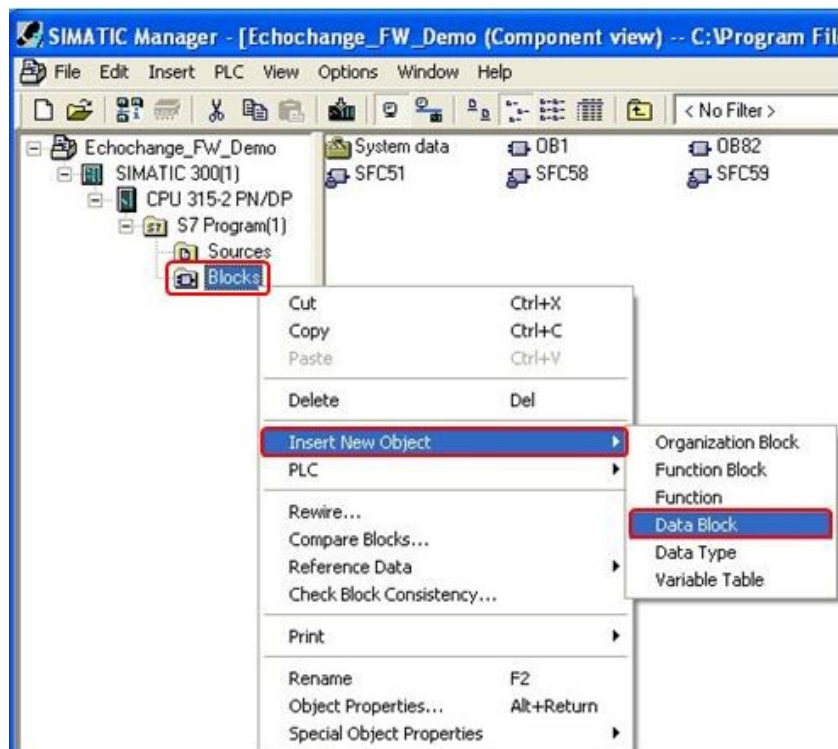


- d. In the next window enter the partner's IP address (echochange) and take note of the connection identifier (**W#16#1** in this case). Next, click on **[Address Details...]**.

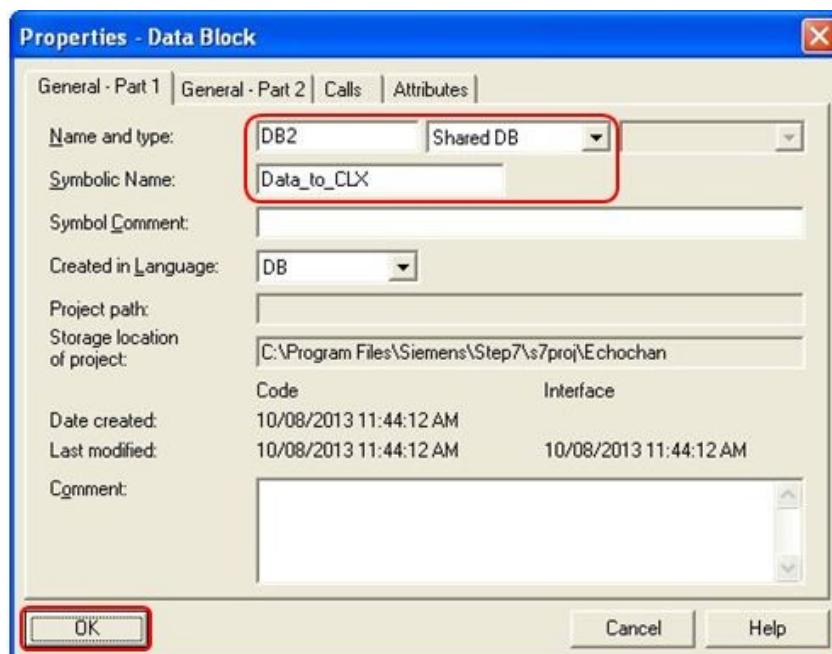
- e. For simplicity, match the Local **Rack/Slot** and **Connection Resource** values on the Partner side. In our case, this resulted in TSAPs 10.02 for both Local and Partner.

- f. Click **[OK]** in this window and in the previous one.

4. Create a data block for the data to be sent to the ControlLogix PLC (1 byte).
  - a. Go back to the project window and add a new data block.

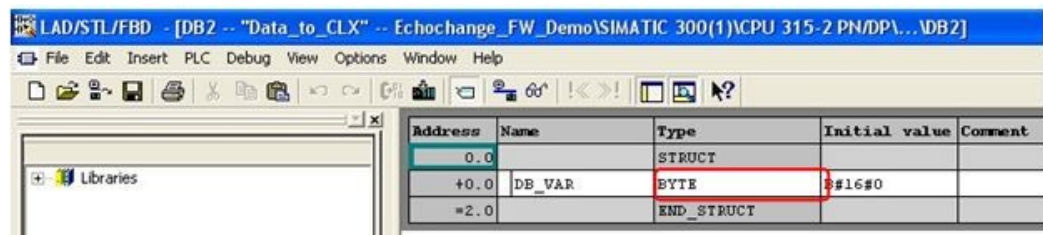


- b. Enter the **Name and type** and the **Symbolic Name**.

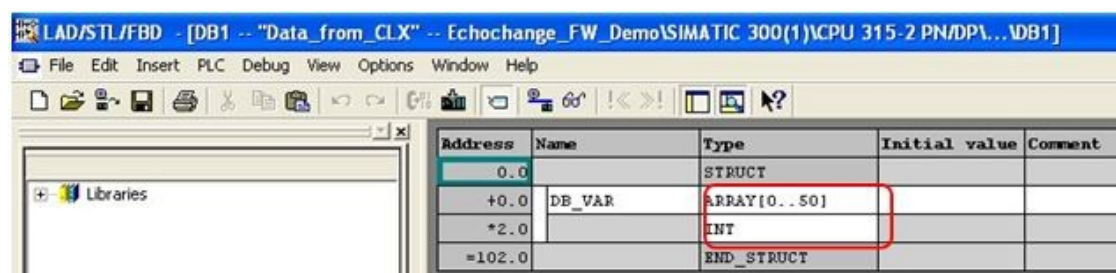
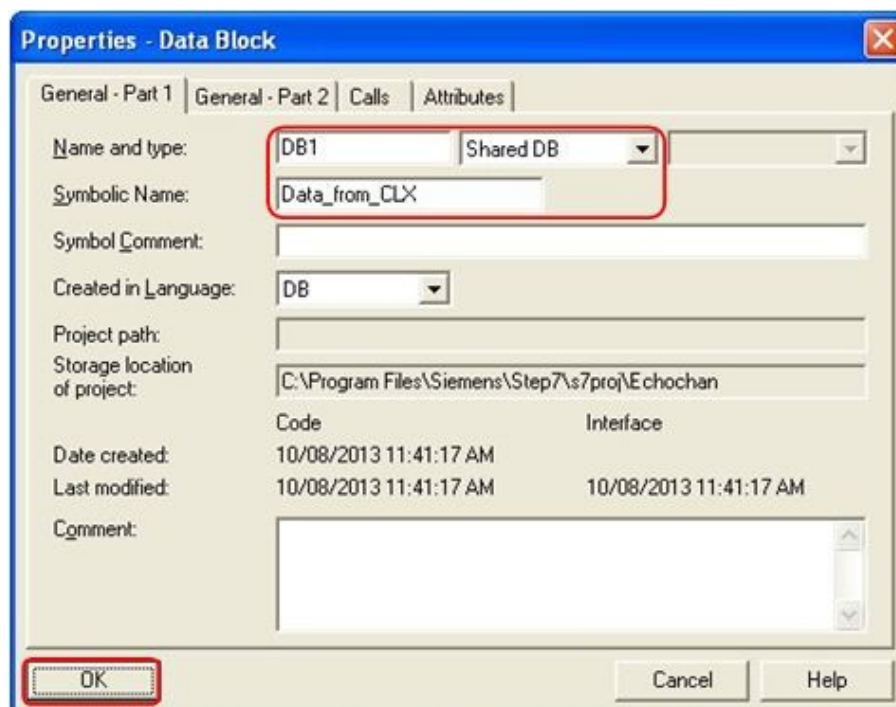




- c. Double-click the new block and select **Byte** for the data type.

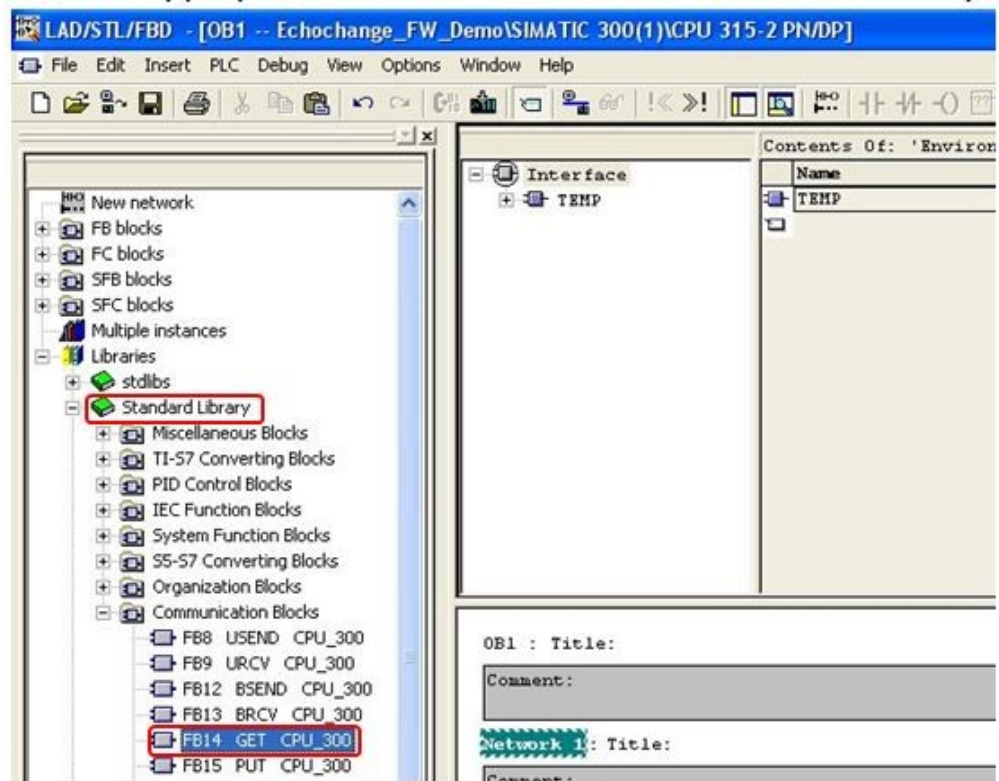


- d. Close the data block configuration window and save the block.
5. Create a data block for the data to be read from the ControlLogix PLC, following the same method described previously. This will be an integer array of 50 elements.

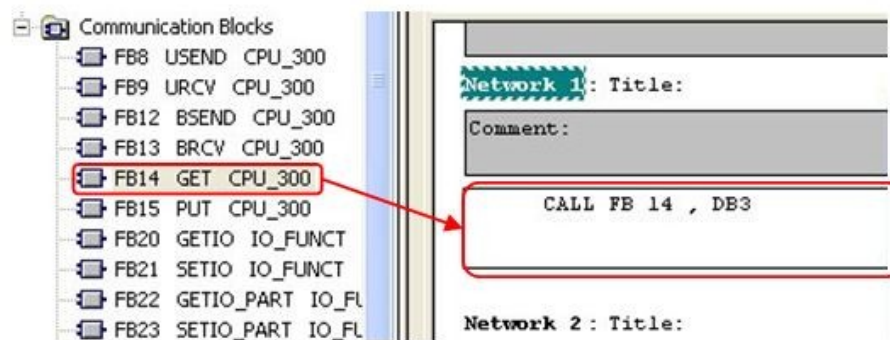


Close the data block configuration window and save the block.

6. Add a GET function on your program to read data from the ControlLogix PLC.
  - a. Find the appropriate GET function under the **Standard Library**.

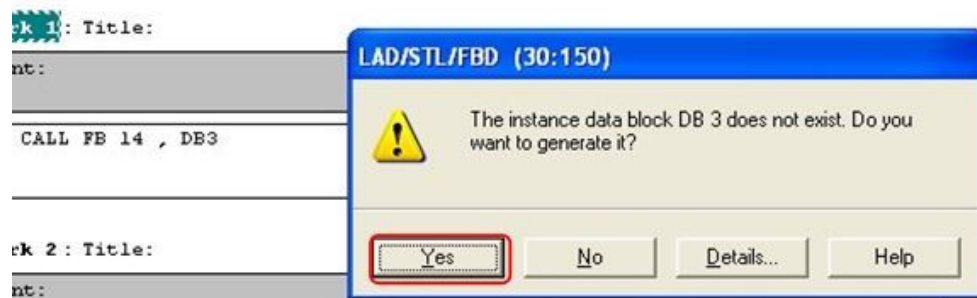


- b. Drag FB14 to the program and assign an unused DB number (DB3 in our example).

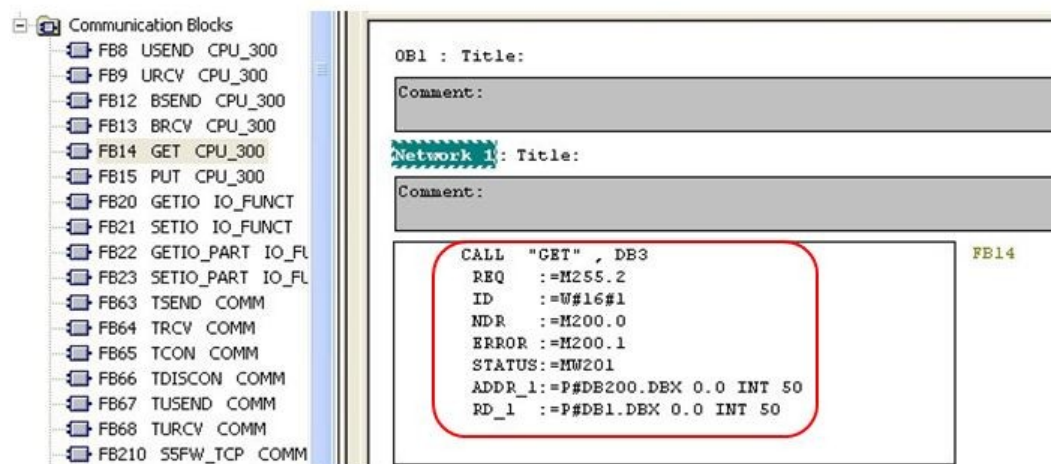


Note that in our example we use STL language, but the same programming can be done in any language.

- c. Press **Enter**, then click **[Yes]** in the following dialog box:



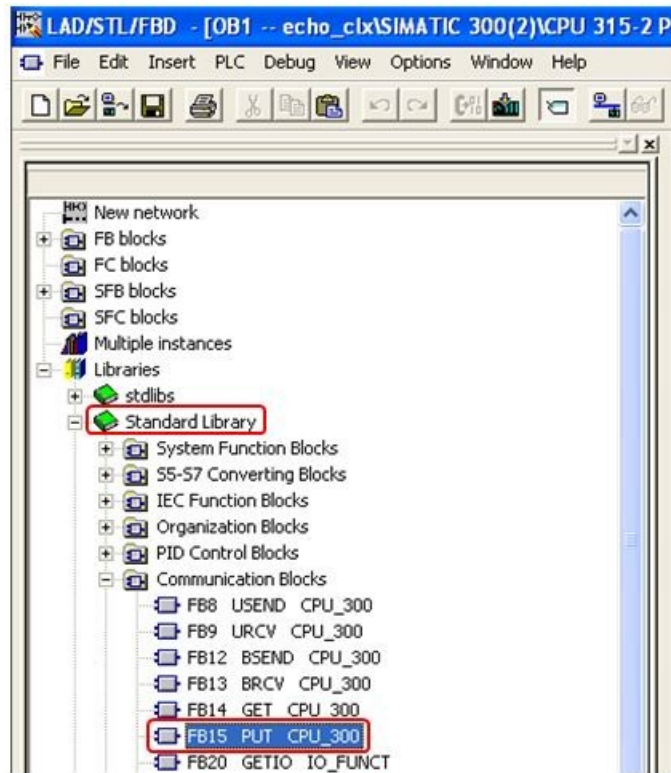
- d. Enter the required parameters for the GET function.



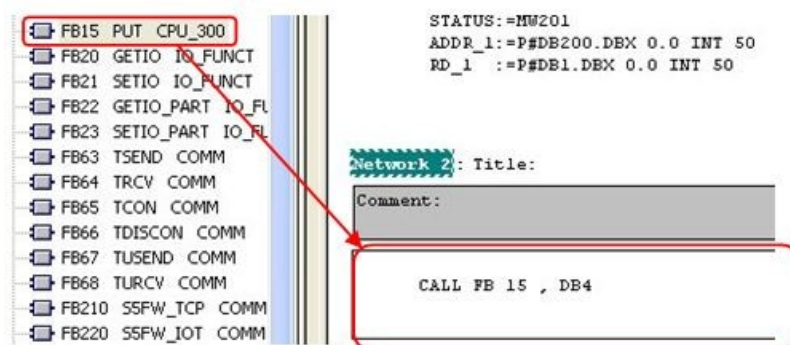
- REQ: control parameter request, activates the data exchange on a rising edge. We use the clock byte we created in chapter 4 step 2 c. We use M255 in our example, and bit 2's frequency is 2.5 Hz.
- ID: this is the connection identifier from chapter 4 step 3 d (W#16#1).
- NDR: new data received status bit. Pick an unused memory location to store this status. We selected M200.0.
- ERROR: error status bit. Pick an unused memory location to store this status. We selected M200.1.
- STATUS: status word. Pick an unused memory location to store this status. We selected MW201.
- ADDR\_1: pointer to the area on the partner CPU that is to be read. This address will be mapped in the echochange.  
P#DB200.DEX 0.0 INT 50: pointer to data block 200, byte offset 0, bit offset 0, of data type INT, 50 elements.
- RD\_1: pointer to the area on the local CPU in which the read data is entered.  
P#DB1.DEX 0.0 INT 50: pointer to data block 1 (Data\_from\_CLX), byte offset 0, bit offset 0, of data type INT, 50 elements.



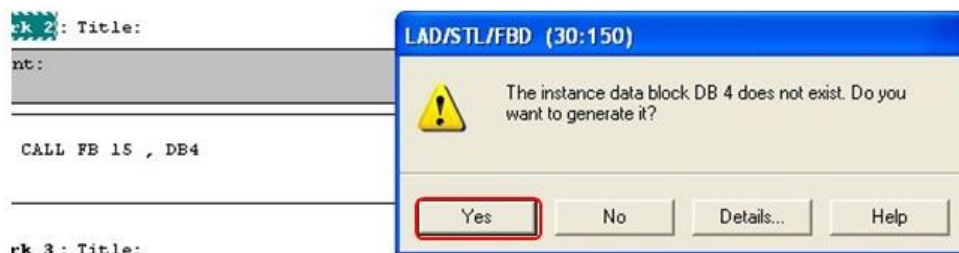
7. Insert a new network on your program and add a PUT function to write data to the ControlLogix PLC.
  - a. Find the appropriate PUT function under the Standard Library.



- b. Drag FB15 to the program, and assign an unused DB number (DB4 in our example).



- c. Press **Enter**, then click **[Yes]** in the following dialog box:



8. Enter the required parameters for the PUT function.

Network 2 : Title:

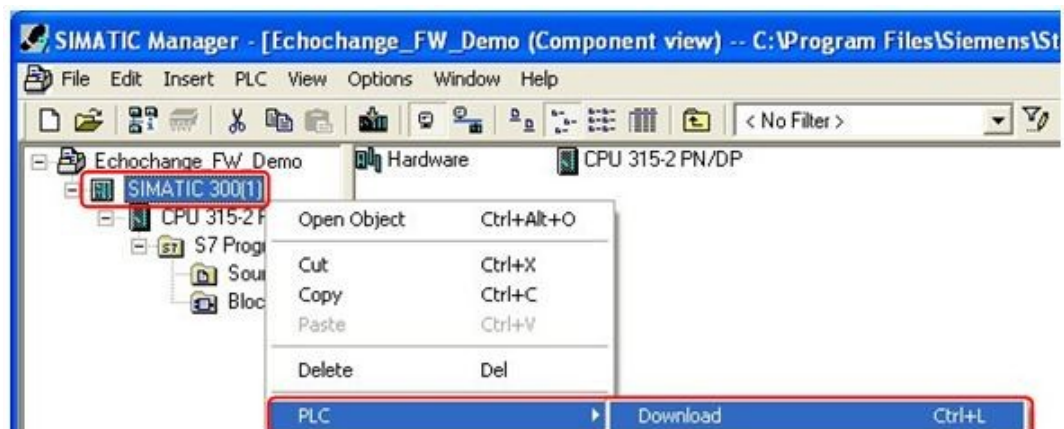
Comment:

```
CALL "PUT" , DB4
REQ    :=M255.2
ID     :=W#16#1
DONE   :=M200.2
ERROR  :=M200.3
STATUS:=MW203
ADDR_1:=P#DB100.DBX 0.0 BYTE 1
SD_1   :=P#DB2.DBX0.0 BYTE 1
```

FB15

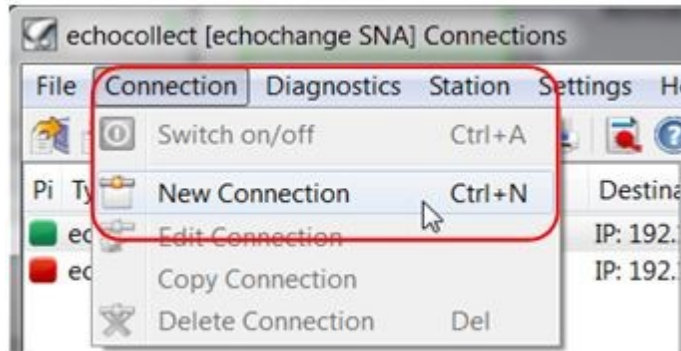
-- Write Data to a Re

- REQ: control parameter request, activates the data exchange on a rising edge. We use the clock byte we created in chapter 4 step 2.c. We use M255 in our example, and bit 2's frequency is 2.5 Hz.
  - ID: this is the connection identifier from chapter 4 step 3.d (W#16#1).
  - DONE: done status bit. Pick an unused memory location to store this status. We selected M200.2.
  - ERROR: error status bit. Pick an unused memory location to store this status. We selected M200.3.
  - STATUS: status word. Pick an unused memory location to store this status. We selected MW203.
  - ADDR\_1: pointer to the area on the partner CPU to which the data will be written. This address will be mapped in the echoschange.  
P#DB100.DBX0.0 BYTE 1: pointer to data block 100, byte offset 0, bit offset 0, of data type BYTE, 1 element.
  - SD\_1: pointer to the area on the local CPU which contains the data to be sent.  
P#DB2.DBX0.0 BYTE 1: pointer to data block 2 (Data\_to\_CLX), byte offset 0, bit offset 0, of data type BYTE, 1 element.
9. Download the project to the S7-300 device.

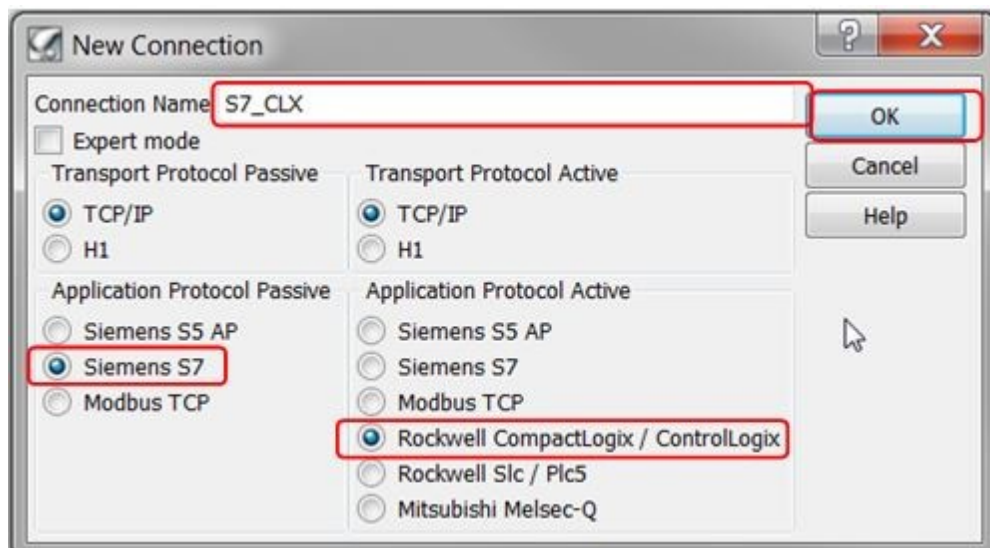


## 5 NetCon - establish the connections

1. Create a new connection.

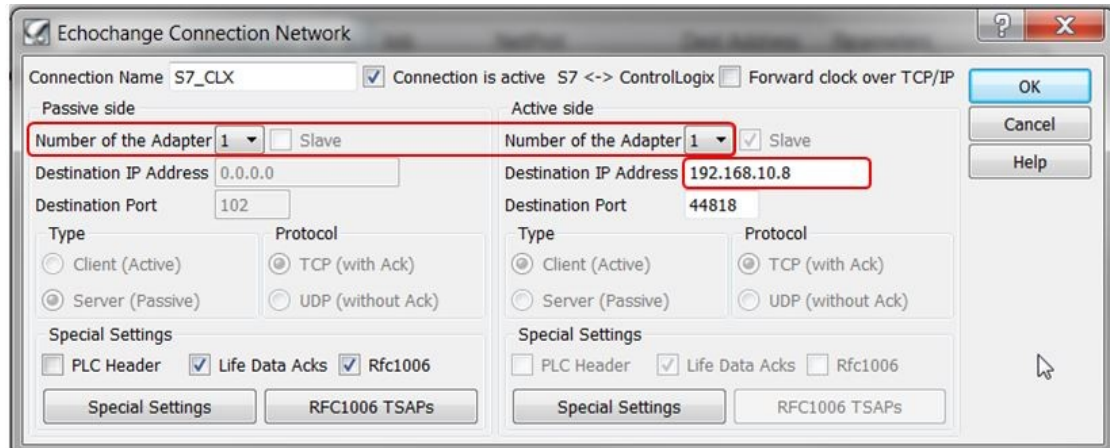


2. Enter a connection name and select **Siemens S7** under the **Application Protocol Passive** section and **Rockwell CompactLogix/ControlLogix** under the **Application Protocol Active** section.

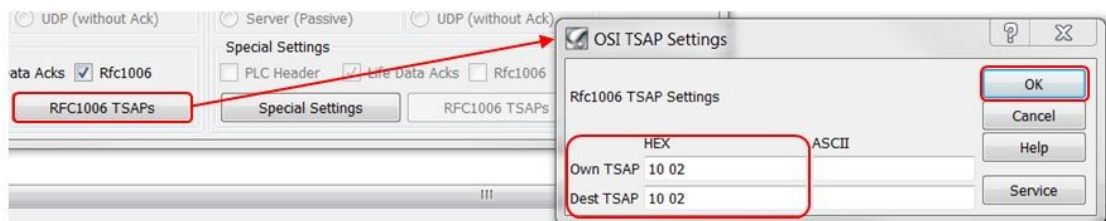


3. Since we are only using port 1 (Eth 1), we will leave the **Number of Adapter** at 1.

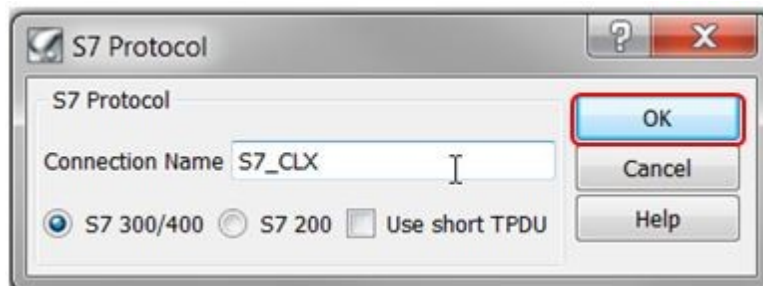
- Enter the IP of the ControlLogix PLC on the **Destination IP Address**, under the **Active side** section.



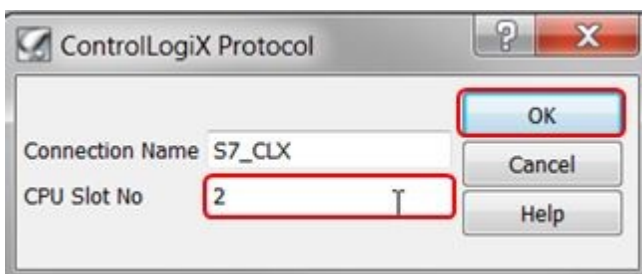
- Click on **RFC1006 TSAPs** and enter the TSAPs chosen in Step 7. See chapter 4 section 3 e.



- After clicking **[OK]** on the **Echochange Connection Network** window, click **[OK]** on the next window:

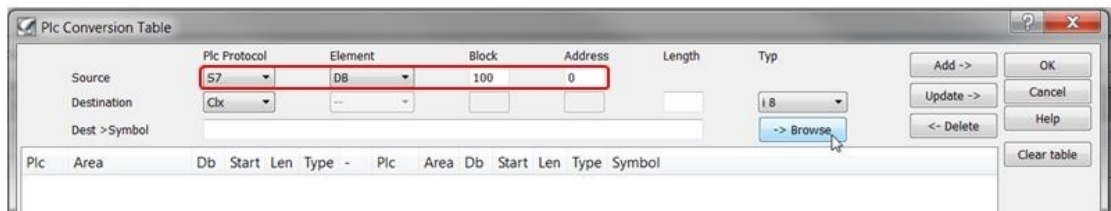


- Enter the ControlLogix CPU slot. In our case we use slot 2.

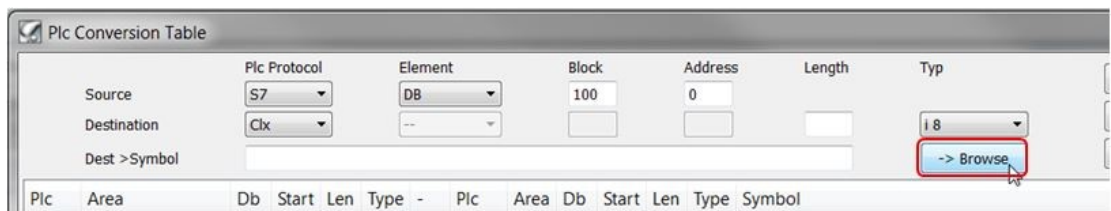


- The **Plc Conversion Table** will come up. Here you will link the S7-300 data areas with the ControlLogix tags.

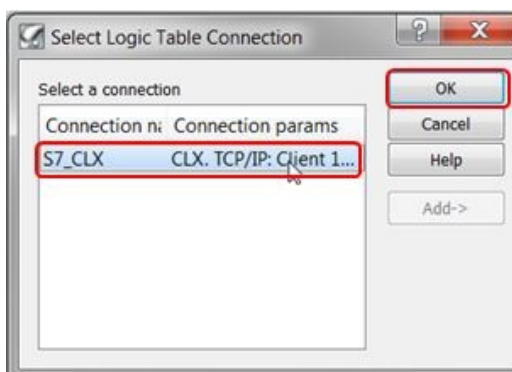
9. First map the S7 data area to be written to the ControlLogix tag. We used P#DB100.DBX0.0 BYTE 1 for the ADDR\_1 parameter in our PUT instruction, so in the echochange we specify **DB100** with **0** offset as follows:



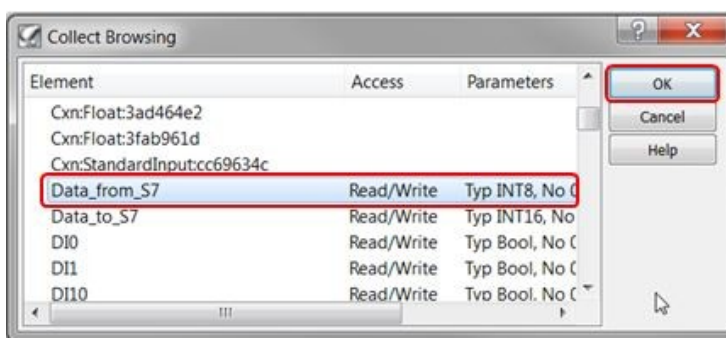
10. Click on [**→ Browse**] to select the respective ControlLogix tag.



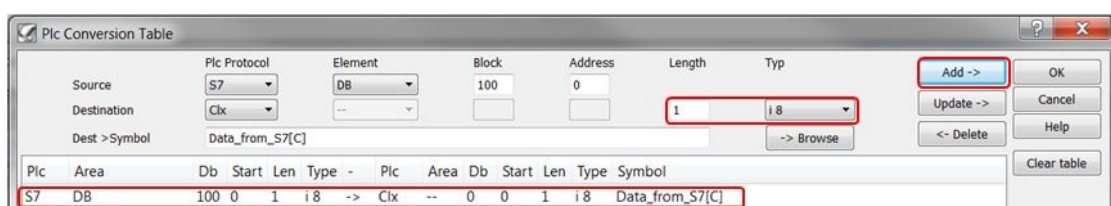
11. Select the connection you created and click [**OK**].



12. Select the tag and click [**OK**].

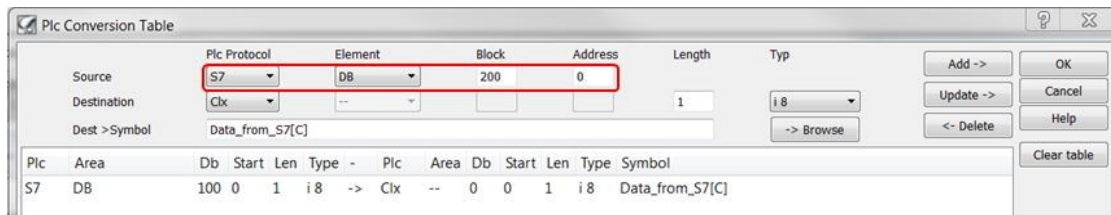


13. Enter the data length and type (1, byte) and click on [**Add →**] to add the mapping to the conversion table.

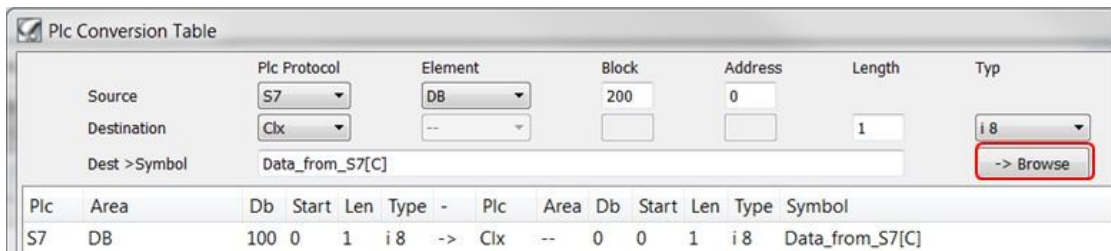




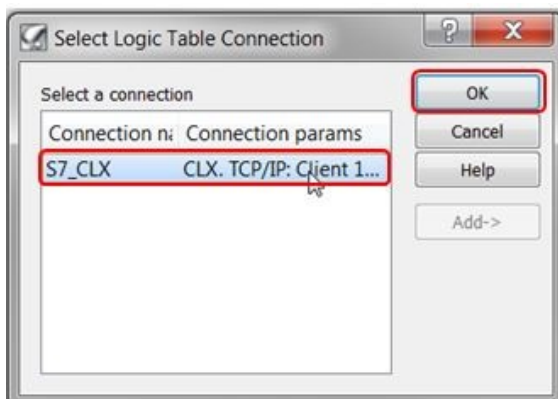
14. Now map the S7 data area used to read from the ControlLogix tag. We used P#DB200.DBX0.0 BYTE 1 for the ADDR\_1 parameter in our GET instruction, so in the echochange we specify **DB200** with **0** offset as follows:



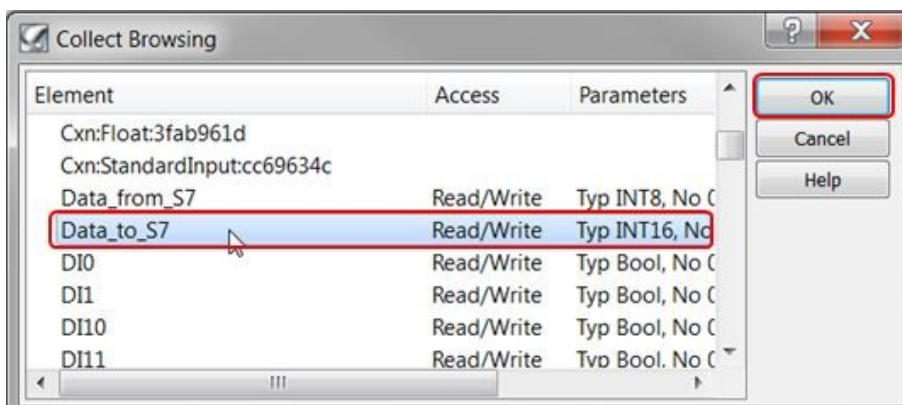
15. Click on [**→ Browse**] to look for the respective ControlLogix tag.



16. Select the connection you created and click [**OK**].



17. Select the tag and click [**OK**].



18. Enter the data length and type (50, Integer) and click on **[Add →]** to add the mapping to the conversion table.

| Plc | Area | Db  | Start | Len | Type | -> | Plc | Area | Db | Start | Len | Type | Symbol          |
|-----|------|-----|-------|-----|------|----|-----|------|----|-------|-----|------|-----------------|
| S7  | DB   | 100 | 0     | 1   | i 8  | -> | Clx | --   | 0  | 0     | 1   | i 8  | Data_from_S7[C] |

19. The conversion table should look like this:

| Plc | Area | Db  | Start | Len | Type | -> | Plc | Area | Db | Start | Len | Type | Symbol            |
|-----|------|-----|-------|-----|------|----|-----|------|----|-------|-----|------|-------------------|
| S7  | DB   | 100 | 0     | 1   | i 8  | -> | Clx | --   | 0  | 0     | 1   | i 8  | Data_from_S7[C]   |
| S7  | DB   | 200 | 0     | 50  | i 16 | -> | Clx | --   | 0  | 0     | 50  | i 16 | Data_to_S7[L0.50] |

20. Click **[OK]** to confirm. Setup is now complete.



**Softing Industrial Automation GmbH**

Richard-Reitzner-Allee 6

85540 Haar / Germany

Tel: + 49 89 4 56 56-0

Fax: + 49 89 4 56 56-488

Internet: <http://industrial.softing.com>

Email: [info.automation@softing.com](mailto:info.automation@softing.com)

Support: [support.automation@softing.com](mailto:support.automation@softing.com)