

How to ...

echochange

Setting up data exchange between a Siemens S7-300 and an Allen-Bradley ControlLogix, using Siemens SEND/RECEIVE functions and Allen-Bradley CIP Generic messages



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 Support: 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

Chapter 1	Introduction	5
Chapter 2	RSLogix 5000- create tags to send and receive data	6
Chapter 3	NetCon echo – echochange network settings	8
Chapter 4	Create data blocks to send and receive data	9
Chapter 5	RSLogix 5000 – send data to echochange	12
Chapter 6	STEP 7 – configure send and receive connections	18
Chapter 7	STEP 7 – program communications	28
Chapter 8	NetCon echo - establish the connections	45



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

- A ControlLogix 1756-ENBT or 1756-EN2T communications processor is used
- The S7-300 communicates raw data using the TCON, TSEND and TRCV functions
- The ControlLogix uses CIP Generic messages to send data
- The S7-300 DB2 data block (Data_to_CLX, 1 INT) is transferred onto the Data_from_S7_raw ControlLogix tag
- The Data_to_S7 ControlLogix tag is transferred onto the S7-300 DB1 data block (Data_from_CLX, integer array of 50 elements)

Software used

- Siemens SIMATIC Step 7 V5.4 SP5 with Open Communication Wizard V2.3 (http:// support.automation.siemens.com/WW/view/en/25209116)
- 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...**.

RSLogix 5000 - CL1_Demo	_Slot2 [1756-L63 20.13]* - [Con	troller Tags - CL1_I	Demo_Slot2(cont	roller
Pile Edit View Search	n Logic Communications To	ools Window H	elp	
8	00	- # 4 9 [F 🛛 🗬 🔍 🤆	2
	Path:	THIP-1\192.168.10	8\Backplane\2*	-
Offline 🛛 🚛 🖻 🖓	RUN LAL		41-44-63	(U)
No Forces	к			191.00
No Edits	AT	Favorite:	s 🖌 Add-On 🖌 Sa	fety
Redundancy 3.5	0			
Controller Organizer	- 4 × [s	Scope: 🗓 CL1_De	mo_Slot2 👻 S	how:
Tan Controller CL1_Der	mo_Slot2	Name 📰 🗠	Alias For	Bas
Controller Fai	👌 New Tag Ctrl+W	DI12	Local:0:1.Data.12	Loc
Power-tin Hat	6	DI13	Local:0:I.Data.13	Loc
Tasks	Monitor Tags	DI14	Local:0:1.Data.14	Loc
A MainTask	Edit Tags	DI15	Local:0:1.Data.15	Loc
MainProgra	Verify		Local:0:1.Data	Loc
E Task2	Export Tags		Local:1:0.Data	Loc
E Periodic_1s		DO0	Local:1:0.Data.0	Loc
Unscheduled	Print	D01	Local:1:0.Data.0	Loc

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

New Tag	Canal Street and Long Ton	X
Name:	Data_from_S7_raw	Create 🗸
Description:	A	Cancel
	•	Help
Туре:	Base Connection	
Alias For:	•	
Data Type:	INT	
Scope:	D CL1_Demo_Slot2	
External Access:	Read/Write	
Style:	Decimal 🔻	
Constant		
Open Con	iguration	



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

New Tag	ten Tag	Surgeries.	X	Constant	Style		
new rug	and them in the second				Decimal		
Name:	Data_to_S7		Create 🔫		Decimal		
	<u> </u>				Binary		
Description:		~	Cancel		Binary		
					Decimal	1	
			Help		Decimal		
		-			Decimal		
	B		Select Data Type	-2-	-	B	X
Туре:	Conn	ection	Data Types:				
Alias For:		-/	INT[50]				ОК
Data Type:	SINT	Ó	FILTER_NOTCH	SELECTO	в		Cancel
Scope:	D CL1_Demo_Slot2	•	FLIP_FLOP_D FLIP_FLOP_JK	_0222010			Help
External Access:	Read/Write	•	FUNCTION_GENE	RATOR			
Style:	Decimal	•	IMC				
Constant							
			Array Dimensions	8			
Open Cor	ntiguration		Dim 2	Dim 1	Dim	0	
	AB:1756_AI6_FI		0 💠	0	50	-	
	AB:1756_AI6_FI				_		
	AB:1756_A06		Show Data Typ	es by Group	S		
	10 1300 100						

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

ope: JOCL1_Demo_	Slot2 -	Show: All rags						
Name		Alias For	Base Tag	Data Type	Description	External Access	Constant	Style
⊞-Data_from_S7_raw			6	INT		Read/Write		Pecimal
⊞-Data_to_S7				INT[50]		Read/Write		Decimal
D10		Local:0:1.Data.0	Local:0:1.Data.0	BOOL		Read/Write		Decimal
DI1		Local:0:1.Data.1	Local:0:1.Data.1	BOOL		Read/Write		Decimal
DI2		Local:0:1.Data.2	Local:0:1.Data.2	BOOL		Read/Write		Decimal
DI3		Local:01 Data 3	Local:01Data 3	BOOL		Read/Write		Decimal

4. 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 \rightarrow 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.

	43		
Station Name		echochange SNA	ОК
MAC (Ethernet)	Address	00 A2 DA 23 06 7E	Cancel
TCP/IP Station	Paramete	r	Help
Use DHCP			
IP Address	192.168	.10.5	
Subnet Mask	255.255	.255.0	
Domain Name	sna.softi	ing.local	
Maximum thre	e Domain	Server Addresses	
DNS Addresses		192.168.10.4	
192.168.10.3		[·	
Maximum thre	e Router	Addresses	
Router Address	es	192.168.10.1	



4 Create data blocks to send and receive data

- 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 Memory Byte **10** in this example.

roperties - CPU 315-2 P	N/DP - (RO/S	52)		Đ
General Diagnostics/Clock Cycle/Clock Memory Rete	Startup Protecti entive Memory	on Interrupts	Synchronous Cycle Communication	Interrupts Web Cvclic Interrupts
Cycle	nage cyclically			
Scan cycle monitoring time	[ms]:	150	1	
Scan cycle load from com	nunication (%):	20	_	
Size of the process-image Size of the process-image	input area: output area:	128	-	
0B85 - call up at I/0 acce	ss error:	No OB85	call up	-
Clock Memory			0	
Clock memory Memory Byte:		10		
ОК			Cancel	Help

- 3. Create a data block for the data to be sent to the ControlLogix PLC.
 - a. Go back to the project window and add a new data block.





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

ame and type:	DB2 Shared [DB 💌
ymbolic Name:	Data_to_CLX	
ymbol <u>C</u> omment		
Created in Language:	DB	
Project path:		
itorage location If project:	C:\Program Files\Siemens\Ste	p7\s7proj\Echochan
	Code	Interface
)ate created:	10/08/2013 11:44:12 AM	
ast modified:	10/08/2013 11:44:12 AM	10/08/2013 11:44:12 AM
Comment		^

c. After clicking **[OK]**, open the new block and verify that the data type is an **INT** (integer).

Address	Name	Туре	Initial value
0.0		STRUCT	_
+0.0	DB_VAR	INT	0
=2.0		END_STRUCT	-

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

ieneral - Part 1 Gener	al - Part 2 Calls Attribu	ites	
Name and type:	DB1 S	hared DB	-
jymbolic Name:	Data_from_CLX		
Symbol Comment:			
Created in Language:	DB V		
Project path:			
Storage location	C:\Program Files\Sieme	ens\Step7\s7proi\Echochan	
or project.	Code	Interface	
Date created:	10/08/2013 11:41:17 A	м	
Last modified:	10/08/2013 11:41:17 A	M 10/08/2013 11:41:17	AM
Comment:			<u>^</u>
			2
		Cancel	Help
OK I			
OK			
OK Iress Name	Туре	Initial value Con	ment
OK Iress Name	Type	Initial value Con	ment
0K Iress Name 0.0 +0.0 DE_VAR	Type STRUCT ARPAY[050	Initial value Con	ment
0K 1ress Name 0.0 +0.0 DB_VAR *2.0	Type STRUCT ARPAY(050 INT	Initial value Con	ment

5. Close the data block configuration window and save the block



5 RSLogix 5000 – send data to echochange

- 1. Configure the send instruction to transmit data from the ControlLogix to the S7 through the echochange.
 - a. Open the **MainRoutine** from the **Controller Organizer** and add a rung to your program.



b. Add a MSG instruction on the newly created rung.





Ladder Element M	SG	Instruction Help >>
Name	Description	
-H Rung	(SOR-EOR)	
	(BST-BND)	
HH Branch Level	(NXB)	=
🗉 🛄 Alarms	10	
Bit		
Timer/Counter	5	
input/Output		
Input/Output	Message	
	Message Get System Value	
E SV H GSV H GSV H SSV	Message Get System Value Set System Value	
Input/Output H GSV H SSV Show Language E	Message Get System Value Set System Value Ilements By Groups	СК
Input/Output H-4 MSG H-4 GSV H-4 SSV Show Language E	Message Get System Value Set System Value Ilements By Groups	OK Cancel

c. On the MSG function block, right-click on Message Control to create a new tag.

Mess	Age Control 2 CEN)-	
×	Cut Instruction	Ctrl+X
	Copy Instruction	Ctrl+C
	Paste	Ctrl+V
	Delete Instruction	Del
	Add Ladder Element	Alt+Ins
	Edit Main Operand Description	Ctrl+D
	Save Instruction Defaults	
	Clear Instruction Defaults	
	Remove Force	
	<u>G</u> o To	Ctrl+G
	Instruction Help	F1



d. Name the new tag and verify its Data Type is set to **MESSAGE**.

New Tag		X
<u>N</u> ame:	send_to_echochange	Create 📐 🔻
<u>D</u> escription:	A	Cancel Help
Typ <u>e</u> :	Base Connection	
Alias <u>F</u> or:	•	
Data <u>T</u> ype:	MESSAGE	
<u>S</u> cope:	DCL1_Demo_Slot2	
E <u>x</u> ternal Access:	Read/Write	
St <u>y</u> le:	•	
Constant		
Open ME	SSAGE Configuration	

e. Open the **Message Configuration** dialog box by clicking on [...], as shown in the figure below:

	MSG	
_	- Message - (EN)	-
	Message Control send_to_echochange	
	View Configuration D	ialog



- f. In the **Configuration** tab, verify that the message type is set to **CIP Generic** and the service type to **Set Attribute Single**.
- g. Choose the tag created in chapter 2, step 2 (Data_to_S7) as the Source Element and specify how many bytes will be transferred. In this example we will send the complete array consisting of 50 integers, so we chose a Source Length of 100 bytes.
- h. In our example, we enter 1 for Class, Instance and Attribute. Service Code is set to 10 by default.

Configuration	* Communication Tag ype: CIP Generic	•	
Service Type: Service Code: Instance:	Set Attribute Single II (Hex) Qlass: 1 (Hex) Attribute: 1 (Hex)	Source Element Source Length: Destination Elemen	Data_to_S7 100 (Bytes) nt New Tag
) Enable	O Enable Waiting O Start	O Done	Done Length: 0



i. Switch to the **Communication** tab and enter the path to establish the connection with the echochange. The path consists of the symbolic name of the local module, the local port (port from which the message exits), target IP address and target port number.

essage Configuration	- send_to_echoo	hange		-
Configuration* Comm	unication* Tag			
Path: ETH_M	ocal, 2, 192.168.10.	5:44818		Browse
ETH_Id	cal, 2, 192.168.10.5	5:44818		
Broadcast	•			
Communication Met	hod	11		
© CIP © D∐•	<u>Channel</u> :	'A'	Destination	Link:
CIP With Source ID	Source Link	0	Destination	Node: 0 (Octal)
Connected		Cach <u>e</u> C	Connections •	Large Connection
O Enable O Enab	ole Waiting	O Start	O Done	Done Length: 0
D Error Code: Error Path: Error Text	Extended	Error Code:		Timed Out *
	[ок	Cancel	Apply Help

ETH_local, 2, 192.168.10.5:44818

ETH_local: symbolic name of the Ethernet communications processor



- 2: Ethernet port from a 1756-ENET module
- 192.168.10.5: echochange IP address
- 44818: echochange TCP port



- j. Add an **Examine Off** instruction to trigger the data transmission. Select the enable (**EN**) bit of the message instruction.

13	III 🖶 🛱 💩 💌 🧆	
	send_to_echochange.EN	Message Message Control send_to_echochange CR>

k. Download the project to the controller.





6 STEP 7 – configure send and receive connections

1. Click on the **Blocks** folder and start the **Open Communication Wizard**.



2. Click **[OK]** when this dialog box comes up.





3. Make sure **New** is selected in the following dialog box and click **[Next]**:

en Commun	ication Wize	ard		
List of conne	ections			
ID (16#)	DB/UDT	Connection type	Connection name	Remote IP
<	1 (New O Change (🔿 Сору	Export
		< Previo	Next >	Finish

4. Select **ISO on TCP** as the protocol layer and click **[Next]**.

Open Communication W	izard		
Connection type			
Choose the type of your	connection.		
◯ TCP natio	re		
Open, con	ection-oriented communication	n as per RFC 793.	
ISO on T Open, con	CP rection-oriented communication	as per RFC 1006.	
O UDP Open, con	nectionless communication as p	per RFC 768.	
	Paning		euros.
Into	< Previous	Next>	Finish



5. Make sure both options below are checked and click the **[Next]** button.

Open Communicat	ion Wizard	
Communication	Partners	
Which communic	cation parter would you like to configure?	
	Communication partner A S7-300 or S7-400	Communication partner B
	Only communication partner <i>I</i> Communication partner B is n	A shall be configured. tot a S7-CPU (eg. a PC).
Info	< Previous	Next > Finish

6. The **Connection ID** is automatically set by the wizard with the first free value. Name the connection and verify the connection establishment is set to **Active**. Select the **Used interface** (CPU type) from the dropdown menu. Under the **Communication partner B** section, enter the IP address of the echochange and click [Next].

Open Communication V		
Communication Parts	ners	
Enter the properties of	rie communications partners.	
Connection ID:	Communication partner A W#16# 0001	Communication partner B
Connection name:	REC_from_CLX	
Connection establishment	Active O Passive	🔿 Active 🛞 Passive
Connection	Unspecified connection partner	Unspecified connection partner
IP address:	000.000.000.000	192.168.10.5
Used interface: 1	CPU 314/315/317; I 🛩	v
Rack / Slot of the CPU:	0 / 2	0 / 0
Rack / Slot of the CP	0 / 3	0 / 0
Info	< Previous	Next > Finish



- Enter TSAP extensions for Communication Partner A and Communication Partner
 B. Use the same on both for simplicity. Names can be up to 6 characters long.
- 8. Take note of the TSAP-IDs in the bottom half of the window. These are generated from the TSAP extensions you entered. Click **[Next]**.

Open Communication		
Connection parameters f	ers or the connection.	
	Communication partner A	Communication partner B
TSAP extension:	Specify TSAP ID	Specify TSAP ID
ASCII	TCP-1	TCP-1
O HEX	5443502D31	5443502D31
Local TSAP-ID: Communication partner A:	CPU 3xx (FW < V2.7), CPU 4 B0025443502D31	4xx (FW < V5.2)
Communication partner B:	CPU 3xx (FW < V2.7), CPU 4 5443502D31	4xx (FW < V5.2) [5]
Info	< Previous	Next > Firish

9. Choose a free data block (DB11 in our example) where the wizard will store the communication parameters and click **[Next]**.

Open Commun	ication Wiz	ard 🔲 🗖 🔀
Choose desti Select STEF	nation proj 7 project, bloc	ect :k folder and block.
STEP 7 project	C:\Program	Files\Siemens\Step7\s7proj\Echoch_2
Communicati	on partner A	
Name:	DB11	(absolute or symbolic, eg. UDT 65 or "TCON_PAR")
Block folder:	Echochang	e_SR_0CW_Demo\SIMATIC 300(1)\CPU 315-2 PN/DP\
Communicati	on partner B	
Name:		(absolute or symbolic, eg. UDT 65 or "TCON_PAR")
Block folder.		
Info		<pre></pre>



10. Click **[Next]** in the following window:

en Communi verview List of conne	cation Wiza	rd		
ID (16#)	DB/UDT	Connection type	Connection name REC_from_CLX	Remote IP 192.168.10.5
<				
Click "Next" to	continue	Previou	n Navt S	Enish

11. The wizard will compile the connection parameters. Make sure that everything finishes without errors or warnings. Since we want to configure a second connection for sending data, check **Configure another connection**, and then click **[Finish]**.

Open Communication Wizard			
Compilation report			
Would you like to configue yet another	connection?		
The following UDTs / DBs were compiled s	uccessfully: DR1	1	
Compilation report:			
10/21/13 11:29:26: COMPILE vo Compile: Echochange SB OCW De	n tmpA2.avl	00(1))CPU 315-	2
PN/DP\S7 Program(1) \Sources\t	apA2.awl		
compiler result: 0 hrior(s),	J warning(s)		
			4
Configure another connection.			
		C	
Info	< Previous	Next >	Finish



12. Configuration of the send connection starts here. Click [Next].



13. Click **[OK]** to continue.



14. Make sure New is selected in the following dialog box and click [Next].

ID (16#)	DB/UDT	Connection type	Connection name	Remote IP
1	DB11	ISO on TCP	REC_from_CLX	192.168.10.5
67 h		1		



15. Select ISO on TCP and click [Next].

ard	🛛
nnection.	
ction-oriented communication as per RFC 7	93.
Ction-oriented communication as per RFC 1	006.
ctionless communication as per RFC 768.	
	_
	ard nnection. ction-oriented communication as per RFC 7 ction-oriented communication as per RFC 1 ctionless communication as per RFC 768.

16. Make sure both options below are checked and click [Next].

Open Communicat	ion Wizard	
Communication	Partners	
Which communic	ation parter would you like to configure?	
	Communication partner A	Communication partner B
	\$7-300 or \$7-400	Non-Simatic station
	Only communication partner A Communication partner B is n	A shall be configured. ot a S7-CPU (eg. a PC).
Info	< Previous	Next > Finish



17. The **Connection ID** is automatically set by the wizard with the first free value. Name the connection and verify the connection establishment is set to **Active**. Select the **Used interface** (CPU type) from the dropdown menu. Under the **Communication partner B** section, enter the IP address of the echochange and click [Next].

Open Communication	Wizard	
Communication Part	ners	
Enter the properties of	the communications partners.	
Connection ID:	Communication partner A	Communication partner B
Connection name:	SEND_to_CLX	
Connection establishment:	Active O Passive	Active () Passive
Connection	Unspecified connection partner	D Unspecified connection partner
IP address:	000.000.000.000	192.168.10 .5
Used interface:	CPU 314/315/317;1 🗸	×
Rack / Slot of the CPU:	0 / 2	0 / 0
Rack / Slot of the CP:	0 / 3	0 / 0
Info	< Previous	Next > Finish

- Enter TSAP extensions for Communication Partner A and Communication Partner
 B. Use the same on both for simplicity. Names can be up to 6 characters long.
- 19. Take note of the TSAP-IDs in the bottom half of the window. These are generated from the TSAP extensions you entered. Click **[Next]**.

onnection parameters	for the connection	
	Communication partner A	Communication partner B
SAP extension:	Specify TSAP ID	Specify TSAP ID
💿 ASC	I TCP-2	TCP-2
O HEX	5443502D32	5443502D32
.ocal TSAP-ID:	6	~
	CPU 3xx (FW < V2.7), CPU 4	4xx (FW < V5.2)
Communication partner A	E0025443502D32	[7]
	CPU 3xx (FW < V2.7), CPU 4	4xx (FW < V5.2)
Communication partner E	5443502D32	(5)



20. Choose a free data block (DB21 in our example) where the wizard will store the communication parameters and click **[Next]**.

en commun	Ication wize	aro E LL
hoose desti	nation proj	ect
Select STEP	7 project, bioc	:K tolder and black.
STEP 7 project	C.\Program	Files\Siemens\Step7\s?proj\Echoch_2
Communicati	on partner A	
Name:	DB21	(absolute or symbolic, eg. UDT 65 or "TCON_PAR")
Block folder:	Echochang	e_SR_0CW_Demo\SIMATIC 300(1)\CPU 315-2 PN/DP\]
Communicati	on partner B	
Name:		(absolute or symbolic, eg. UDT 65 or "TCDN_PAR")
Block folder:		

21. Click [Next].

List of connect	tions			
ID (16#)	DB/UDT	Connection type	Connection name	Remote IP
1	DB11	ISO on TCP	REC_from_CLX	192.168.10.5
<				



22. The wizard will compile the connection parameters. Make sure that everything finishes without errors or warnings and click **[Next]**.



23. Click [Finish] to close the wizard.



7 STEP 7 – program communications

1. If you don't already have it, create OB100 (this is a special start-up OB that is executed before OB1).



2. Type in **OB100** for the name and click **[OK]**.



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

eneral - Part 1 Gener	al - Part 2 Calls Attributes	
Name:	OB100	
Symbolic Name:		
Symbol Comment:		
Created in Language:	STL	
Project path:		
Storage location	C:\Program Files\Siemens\Ste	p7\s7proj\Echoch_2
	Code	Interface
Date created:	10/21/2013 11:45:43 AM	
.ast modified:	10/21/2013 11:45:43 AM	10/21/2013 11:45:43 AM
Comment:		^
		×



3. We want to initialize the TCON function at startup (warm restart). Open OB100 and pick a free memory bit (M0.0 in our example) to be used to trigger the connection establishment. We want to set the bit to **1** in OB100:



4. This is the LAD equivalent:



- 5. Save and close OB100.
- 6. Create a new function block for the receive connection.





7. Name the function block (FB1 in this example) and click [OK].

<u>N</u> ame:	FB1	Mul. Inst. Cap.
Symbolic Name:		
Symbol <u>C</u> omment:		
Created in Language:	STL 💌	
Project path:		
Storage location of project:	C:\Program Files\Siemens\Ste	p7\s7proj\Echoch_2
	Code	Interface
Last modified:	10/21/2013 11:51:17 AM	10/21/2013 11:51:17 AM
C <u>o</u> mment:		4
orrage location of project Date created: Last modified: Comment:	C:\Program Files\Siemens\Ste Code 10/21/2013 11:51:17 AM 10/21/2013 11:51:17 AM	p7\s7proj\Echoch_2 Interface 10/21/2013 11:51:17 AM

8. Open the newly created function block to program it. Add a call to FB65 TCON, found under Libraries → Standard Library → Communication Blocks.

SFC blocks Multiple instances Standard Library Standard	Therface The Ins The OUT The	Contents Of: 'Environment\Im
Granization Blocks GranizationBlocks Granization Blocks Granization Blocks	FB1 : Title: Comment:	
FB14 GET GPU_300 FB15 PUT CPU_300 FB20 GETTO IO_FUNCT FB21 SETTO JO_FUNCT FB22 GETTO_PART IO_FUN FB23 SETTO_PART IO_FUN FB63 SEEND COMM FB64 TRCV COMM	Comment:	

9. After adding FB65 to your program, assign an unused DB number (**DB13** in this example).





10. After hitting Enter, click [Yes] on the following dialog box:



- 11. Set the parameters for the TCON function block.
 - a. Right-click on ID and select Insert Symbol.

Comment:		
CALL "TCON REQ :=M	, DB13 0.0	FB6
ID :=	Paste	Ctrl+V
BUSY :=	Delete	Del
STATUS :=	Insert Network	Ctrl+R
CONNECT: =	Insert Symbol	Ctrl+J
	Go To Object Properties	Alt+Return
	Change to Multiple Instance	Call

b. Browse to the id symbol of the DB structure chosen in chapter 6 step 8. In our example it is **DB11.0UCW_1.id**. Double-click on it.

REQ :=MO.O ID :=		FB65	00	nnect		
DB11.0UCW_1.id						
🗄 🚇 Data_from_CLX	DB	1	DB	1		
🗄 🕒 Data_to_CLX	DB	2	DB	2		
🕀 🔀 DB11	DB 11		DB11			
E BB11.0UCW_1	Struct	5	DB11.	DBD	0	
DB11.0UCW_1.active_est	Bool		DB11.	DBX	5.0	
DB11.0UCW_1.block_length	Word		DB11.	DBW	0	#!REC_from_CLX!#
DB11.0UCW 1.connection type	Byte		DB11.	DBB	4	
1 DB11.0UCW_1.1d	Word		DB11.	DBW		
DB11.0UCW_1.local_device_id	Byte		DB11.	DBB	6	
<pre>#DB11.0UCW_1.local_tsap_id[]</pre>	Array	[116] Of Byte	DB11.	DBD	12	
DB11.0UCW_1.local_tsap_id_len	Byte		DB11.	DBB	7	
DB11.0UCW l.next staddr[]	Array	[16] Of Byte	DB11.	DBD	56	



c. Right-click on CONNECT and select Insert Symbol.

DONE BUSY	:=DB11.DBWZ :=M1.0 :=M1.1		
ERROR	:=M1.2 :=MW2		
CONNECT	Paste	Ctrl+V	
	Delete	Del	
	Insert Network	Ctrl+R	
	Insert Symbol	Ctrl+J	
	Go To		,
	A second state and state state and state state state and state stat state state s	10 m an	

d. Select the whole structure that was created in chapter 6 step 8 by double-clicking on it. In our example this is **DB11.OUCW_1**.

CALL "TCON", DB13 REQ :=H0.0 ID :=DB11.DBW2 DONE :=H1.0 BUSY :=H1.1 ERROR :=H1.2 STATUS :=HW2 CONNECT:=		FB65	Connect	
DB11.0UCW_1	DB	1	DR 1	
Data to CLX	DB	2	DB 2	
	DB 11	_	DB11	
DB11.0UCW_1	Struc	5	DB11.DBD 0	
- □ (DB13	FB 65	and the second sec	DB13	
 € @DB21 	DB 21		DB21	
	FB 1		FB1	
B PB2	FB 2		FB2	
(TCON	FB	65	FB 65	Connect

e. For the REQ parameter, enter the memory bit we set to **1** in chapter 6 step 3 (**M0.0**). Choose any unused bit or word respectively for the output parameters DONE, BUSY, ERROR and STATUS.

etwork 1: Title:		
CALL "TCON", DB13 REQ :=M0.0 ID :=DB11.DBW2 DONE :=M1.0 BUSY :=M1.1 PDPOP :=W1.2	FB65	Connect
STATUS :=MW2 CONNECT:=DB11.OUCW_1	P#DB11.DBX0.0	



- 12. Call the **TRCV** function.
 - a. Insert a new network into function block FB1.



b. Add a call to FB64 TRCV, found under Libraries \rightarrow Standard Library \rightarrow Communication Blocks.



c. After adding FB64 to Network 2, assign an unused DB number (**DB14** in this example).

FB21 SETIO IO_FUNCT	Network 2: Title:	
ER23 SETIO PART TO FUN	Comment:	
EB63 ISEND COMM		
FB64 TRCY COMM	C aver m cr morel	
FB65 TCON COMM	CALL PD 64 , DD14	
FB66 TDISCON COMM		
FB67 TUSEND COMM	<u> </u>	



d. After hitting Enter, click [Yes] on the following dialog box:



- 13. Set the parameters for the TRCV function block.
 - a. Right-click on ID and select Insert Symbol.

ment:		
CALL "TRCV" EN_R :=T	, DB14 RUE	FB6
ID := LEN :=	Paste	Ctrl+V
NDR := BUSY :=	Delete	Del
ERROR :=	Insert Network	Ctrl+R
RCVD_LEN: =	Insert Symbol	Ctrl+J
DATA :=	Go To Object Properties	► Alt+Return
	Change to Multiple Instance	Call

b. Browse to the id symbol of the DB structure chosen in chapter 6 step 8. In our example it is **DB11.OUCW_1.id**. Double-click on it.

comment:			
CALL "TRCV", DB14 EN_R :=TRUE ID :=	FB64	Receive Data	
DB11.0UCW_1.id			
Data_from_CLX	DB 1	DB 1	
BData_to_CLX	DB 2	DB 2	
🕀 🕒 DB11	DB 11	DB11	
😑 🔚 DB11.0UCW_1	Struct	DB11.DBD 0	
DB11.0UCW_1.active_est	Bool	DB11.DBX 5.0	
DB11.0UCW_1.block_length	Word	DB11.DBW 0	#!REC_from_CLX!#
DB11.0UCW 1.connection type	Byte	DB11.DBB 4	
DB11.0UCW_1.id	Word	DB11.DBW 2	
DB11.0UCW_1.local_device_id	Byte	DB11.DBB 6	
<pre>#DB11.0UCW_1.local_tsap_id()</pre>	Array [116] Of Byte	DB11.DBD 12	
DB11.0UCW_1.local_tsap_id_len	Byte	DB11.DBB 7	
<pre>DB11.0UCW_1.next_staddr[]</pre>	Array [16] Of Byte	DB11.DBD 56	
DB11.0UCW_1.next_staddr_len	Byte	DB11.DBB 11	
DB11.0UCW_1.rem_staddr[]	Array [16] Of Byte	DB11.DBD 34	
DB11.0UCW_1.rem_staddr_len	Byte	DB11.DBB 9	
<pre>#DB11.0UCW_1.rem_subnet_id[]</pre>	Array [16] Of Byte	DB11.DBD 28	



- c. Set EN_R to TRUE.
- d. Set LEN to 0.
- e. Set DATA with a pointer to the area where the received data will be stored. In chapter 4 step 4 we created DB1 for this purpose and the pointer to this area is P#DB1.DBX0.0 BYTE 100.
- f. Choose any unused bit or word respectively for the remaining output parameters NDR, BUSY, ERROR, STATUS and RCVD_LEN.

mment:		
CALL "TRCV", DB14 EN_R :=TRUE ID :=DB11.DBW2	FB64	Receive Data
LEN :=0 NDR :=M1.3 PHOY ==N1.4		
ERROR :=M1.5 STATUS :=MW4		
RCVD_LEN: =MW6 DATA :=P#DB1.DBX0.0 BYTE	100	

- g. Save and close FB1.
- 14. Create a new function block for the send connection.





15. Name the function block (FB2 in this example) and click [OK].

Name:	FB2	🗹 Mul. Inst. Cap.
Symbolic Name:		
Symbol <u>C</u> omment:		
Created in Language:	STL -	
Project path:		
Storage location of project:	C:\Program Files\Siemens\Ste	p7\s7proj\Echoch_2
	Code	Interface
Date created:	10/21/2013 12:26:00 PM	
Last modified:	10/21/2013 12:26:00 PM	10/21/2013 12:26:00 PM
D <u>o</u> mment:		2

16. Open the newly created function block to program it. Add a call to FB65 TCON, found under Libraries → Standard Library → Communication Blocks.

KAD/STL/FBD - [FB2 Echochange_SR_0	CW_Demo\SIMATIC_300(1)\CPU 315-2 PN/DP]
File Edit Insert PLC Debug View Options	Window Help	
	🏫 🔁 📲 66° !«>	! 🔲 🖪 🔛 카카 -0 🖾 나 🎝 1
SFB blocks SFC blocks SFC blocks Multiple instances Standard Library Standard Library Miscellaneous Blocks TL-57 Converting Blocks D D Control Blocks	Interface IN IN IN IN IN IN IN TIN TIN TIN TIN TIN	Contents Of: 'Environment\Ir Name IN OUT IN_OUT STAT TEMP
System Function Blocks System Function Blocks Grogenization Blocks Grogenizatio	FB2 : Title: Comment: Network 1: Title: Comment:	
FB64 TRCV COMM FB65 TCON COMM FB66 TDISCON COMM		

17. After adding FB65 to your program, click **[No]** on the following dialog box:

!\	The object	:t 'FB65' already exists. Do you want to it?
<u>B</u> ena	sme	Adjust Attributes



18. Assign an unused DB number (DB23 in this example).





19. After hitting **Enter**, click **[Yes]** on the following dialog box:



- 20. Set the parameters for the TCON function block.
 - a. Right-click on ID and select Insert Symbol.

Comment:		
CALL "TCON REQ :=N	, DB23 0.0	FB6
DONE :=	Paste	Ctrl+V
BUSY := ERROR :=	Delete	Del
STATUS :=	Insert Network	Ctrl+R
connact	Insert Symbol	Ctrl+J
	Go To Object Properties	ب Alt+Return
	Change to Multiple Instance	Cal



b. Browse to the id symbol of the DB structure chosen in chapter 6 step 18. In our example it is **DB21.OUCW_1.id**. Double-click on it.

Conment:						
Network 1: Title:				_		
Comment:						
CALL "TCON", DB23 REQ :=M0.0 ID :=		FB65	Cor	mect		
DB21.0UCW_1.id						
Data_from_CLX	DB	1	DB	1		
@Data_to_CLX	DB	2	DB	2		
🖲 🔀 DB11	DB 11		DB11			
🕀 🚇 DB13	FB 65		DB13			
⊞ @DB14	FB 64		DB14			
🕀 🚇 DB 21	DB 21		DB21			
E BDB21.0UCW_1	Struct		DB21.1	DBD	0	
DB21.0UCW_1.active_est	Bool		DB21.1	DBX	5.0	
DB21.0UCW_1.block_length	Word		DB21.1	DBW	0	#!SEND_to_CLX!#
DB21.0UCW 1. connection type	Byte		DB21.1	DBB	4	
DB21.0UCW_1.id	Word		DB21.1	DBW	2	
DB21.0UCW_1.local_device_id	Byte		DB21.1	DBB	6	
<pre>DB21.0UCW_1.local_tsap_id[]</pre>	Array	[116] Of Byte	DB21.1	DBD	12	
DB21.0UCW_1.local_tsap_id_len	Byte		DB21.1	DBB	7	
DB21.0UCW_1.next_staddr[]	Array	[16] Of Byte	DB21.1	DBD	56	
DB21.0UCW_1.next_staddr_len	Byte		DB21.1	DBB	11	
DB21.0UCW_1.ren_staddr[]	Array	[16] Of Byte	DB21.1	DBD	34	
DB21.0UCW_1.ren_staddr_len	Byte		DB21.1	DBB	9	
<pre>DB21.0UCW_1.rem_subnet_id[]</pre>	Array	[16] Of Byte	DB21.1	DBD	28	

c. Right-click on CONNECT and select Insert Symbol.

CALL "TC	ON" , DB23	FB65
REQ :	=M0.0	
ID :	=DB21.DBW2	
DONE :	=M11.0	
BUSY :	=M11.1	
ERROR :	=M11.2	
STATUS :	=HW12	
CONNECT:	Paste	Ctrl+V
L	— Delete	Del
	Insert Network	Ctrl+R
	Insert Symbol	Ctrl+J
	Go To	•
	Object Properties	Alt+Return
	Change to Multiple Instance	Call

d. Select the whole structure created in chapter 6 step 18 by double-clicking on it. In our example this is **DB21.OUCW_1**.

REQ :=M0.0 ID :=DB21.DBW2 DONE :=M11.0 EUSY :=M11.1 ERROR :=M11.2 STATUS :=HW12 CONNECT:=			
DB21.0UCW_1			
	DB 1	DB 1	
@ Data_to_CLX	DB 2	DB 2	
🕀 🕒 DB11	DB 11	DB11	
🕀 🕒 DB13	FB 65	DB13	
⊕ ⊕DB14	FB 64	DB14	
🕀 🚇 DB21	DB 21	DB21	
DB21.0UCW_1	Struct	DB21.DBD 0	
E DB23	FB 65	DB23	
🕒 FB1	FB 1	FB1	
D FB2	FB 2	FB2	
TCON	FB 65	FB 65	Connect
TRCV	FB 64	FB 64	Receive Data



21. For the REQ parameter, enter the memory bit we set to 1 in chapter 6 step 3 (**M0.0**). Choose any unused bit or word respectively for the output parameters DONE, BUSY, ERROR and STATUS.

Comment:		
CALL "TCON", DB23 REQ :=M0.0 ID :=DB21.DBW2 DONE :=M11.0 BUSY :=M11.1	FB65	Connect
ERROR :=M11.2 STATUS :=MW12 CONNECT:=DB21.0UCW_1	P#DB21.DBX0.	0

- 22. Call the **TSEND** function
 - a. Insert a new network into function block FB2.

Connent	: 0		
Network	×		
	Cut	Ctrl+X	
Connen	Copy	Ctrl+C	
	Paste	Ctrl+V	
1	Delete	Del	FB65
C	Insert Netw	ork Ctrl+R	
	Edit Symbol	s Alt+Return	
	STATUS := M	W12	
	CONNECT := D	B21. OUCW 1	P#DB21.D



b. Add a call to FB63 TSEND, found under Libraries → Standard Library → Communication Blocks.



c. After adding FB63 to Network 2, assign an unused DB number (**DB24** in this example).



d. After hitting Enter, click [Yes] on the following dialog box:





- 23. Set the parameters for the TSEND function block.
 - a. Right-click on ID and select Insert Symbol.

Comment:		
CALL "TSEN REQ :=N	ND", DB24 10.7	FB6
ID := LEN :=	Paste	Ctrl+V
DONE := BUSY :=	Delete	Del
ERROR :=	Insert Network	Ctrl+R
DATA :=	Insert Symbol	Ctrl+3
	Go To Object Properties	Alt+Return
	Change to Multiple Instance	Call

b. Browse to the id symbol of the DB structure chosen in chapter 6 step 18. In our example it is **DB21.OUCW_1.id**. Double-click on it.

Comment:			
CALL "TSEND", DB24 REQ :=M10.7 ID :=	FB63	Send Data	
DB21.0UCW_1.id			
🗄 🕒 Data_from_CLX	DB 1	DB 1	
Data_to_CLX	DB 2	DB 2	
🕀 🕒 DB11	DB 11	DB11	
₩ @ DB13	FB 65	DB13	
🗄 🕮 DB14	FB 64	DB14	
🗄 🔀 DB21	DB 21	DB21	
😑 🔚 DB21.0UCW_1	Struct	DB21.DBD 0	
DB21.OUCW_1.active_est	Bool	DB21.DBX 5.0	
DB21.0UCW_1.block_length	Word	DB21.DBW 0	#!SEND_to_CLX!#
DB21.0UCW 1.connection type	Byte	DB21.DBB 4	
DB21.0UCW_1.id	Word	DB21.DBW 2	
DB21.0UCW_1.local_device_id	Byte	DB21.DBB 6	
DB21.0UCW_1.local_tsap_id[]	Array [116] Of Byte	DB21.DBD 12	
DB21.0UCW_1.local_tsap_id_len	Byte	DB21.DBB 7	
<pre>#DB21.0UCW_1.next_staddr[]</pre>	Array [16] Of Byte	DB21.DBD 56	
DB21.0UCW_1.next_staddr_len	Byte	DB21.DBB 11	
BDB21.0UCW_1.rem_staddr[]	Array [16] Of Byte	DB21.DBD 34	
TE DR21 OHON 1 yam staddy lan	Rota	DR21 DRR 9	



- c. Set the REQ parameter to the clock memory bit enabled in chapter 4 step 2. We use **M10.7** in this example, which has a frequency of 0.5Hz.
- d. Set LEN to 2.
- e. Set DATA with a pointer to the data area that will be sent. In chapter 4 step 3 we created DB2 for this purpose and the pointer to this area is P#DB2.DBX0.0 BYTE 2.
- f. Choose any unused bit or word respectively for the remaining output parameters DONE, BUSY, ERROR and STATUS.



- g. Save and close FB2.
- 24. Program OB1 to call FB1 (receive) and FB2 (send).
 - a. Locate FB1 within the FB blocks.



b. After adding FB1 to your program, assign an unused DB number (**DB12** in this example).

OB1 :	"Main Program Sweep (Cycle)"	
Conner	nt:	
Networ	1 : Title:	
Comme	nt:	
_	CALL FB 1 , DB12	
		_



c. After hitting Enter, click [Yes] on the following dialog box:



d. Locate FB2 within the FB blocks.



e. After adding FB2 to your program, assign an unused DB number (**DB22** in this example).



f. After hitting Enter, click [Yes] on the following dialog box:





25. Reset the memory bit M0.0 which was used to indicate the start-up of the CPU and trigger the connection establishment of both connections.

ment:			
work 1	Title	:	
ment:			
CALL	FB	1 , DB12	
CALL	FB FB	1 , DB12 2 , DB22	
CALL CALL	FB FB N	1 , DB12 2 , DB22 0.0	

26. This is the LAD equivalent:



27. Download the project to the S7-300 device.

💹 SIMATIC Manager - [Echochange_SF	R_OCW_Demo (C	omponent	view) C:\Pros	gram Files\Si
File Edit Insert PLC	View Options	Window Help			
🗅 😂 🎛 🛲 🐰 🗉	b 🔒 🕍 🛛 👁	9 9 5 E		< No Filter >	•
Echochange_SR_00	W_Den Den Hard	ware 🚺 CF	PU 315-2 PN/D	P	
E- CPU 315-2	Open Object	Ctrl+Alt+O	1		
- 57 F10	Cut	Ctrl+X			
Blo	Сору	Ctrl+C			
	Paste	Ctrl+V			
	Delete	Del			
	PLC	•	Download		Ctrl+L
	Print	•	Compile and Copy RAM t	Download Objects to ROM	i



8 NetCon echo - establish the connections

- 1. Create a new connection for the communication direction $CLX \rightarrow Echochange \rightarrow Siemens$.
 - a. Click on **Connection** \rightarrow **New Connection**.



- b. Enter a connection name and enable Expert mode.
- c. After Expert mode is checked, a warning message will pop up. Click [Yes].



d. Select Send/Receive under the Application Protocol Left section and Rockwell CompactLogix/ControlLogix under the Application Protocol Right section.

New Connection	? X	
Connection Name S7-rec-from-CLX		ок
Expert mode		
Transport Protocol Left	Transport Protocol Right	Cancel
TCP/IP	TCP/IP	Help
🔘 н1	🔘 н1	
Application Protocol Left	Application Protocol Right	
Send/Receive	Send/Receive	
Siemens S5 AP	Siemens S5 AP	
Siemens SZ	Siemens SZ	
Modbus TCP	Modbus TCP	
Rockwell CompactLogix / ControlLogix	Rockwell CompactLogix / ControlLogix	
C Rockwell Slc / Plc5	Rockwell Sic / Pic5	
Mitsubishi Melsec-Q	Mitsubishi Melsec-Q	



- e. Since we are only using port 1 (Eth 1), we will leave the **Number of Adapter** at **1**. On the **Right side** (ControlLogix), uncheck **Slave** and select **Server (Passive)** type. On the **Left side** under **Special Settings**, check **Rfc1006**.
- f. Click on the [RFC1006 TSAPs] button.



g. Enter the HEX values of the TSAPs created by the **Open Communication Wizard** in chapter 6 step 7.

🕼 OSI TS	OSI TSAP Settings		? X
Pfc1006 TS	AP Settings		ОК
RICIUU ISAP Settings		Cancel	
	HEX	ASCII	Help
Own TSAP	54 43 50 2D 31	TCP-1	
Dest TSAP	E0 02 54 43 50 2D 31		Service



Note

Own TSAP = Partner B TSAP

Dest TSAP = Partner A TSAP

h. After clicking **[OK]** in the **Echochange Connection Network** window, enter the ControlLogix CPU slot. In our case it's 2.

🕼 ControlLogi	(Protocol	? ->
		ок
Connection Name	S7-rec-from-CLX	Cancel
CPU Slot No	2	Help



i. Enter the ControlLogix tag name created in chapter 2 step 2 (used to send data to the S7-300) and enable **Raw Data Endian Convert**.



- 2. Create a new connection for the communication direction **Siemens** \rightarrow **Echochange** \rightarrow **CLX**.
 - echocollect [echochange SNA] Connections Connection Diagnostics Station File Settings He 3 Switch on/off Ctrl+A 10 Pi Destina **New Connection** Ctrl+N IP: 192. ec Edit Connection IP: 192. e ec Copy Connection **Delete Connection** Del
 - b. Enter a connection name and enable **Expert mode**. After **Expert mode** is checked, a warning will pop up. Click **[Yes]**.



a. Click on **Connection** \rightarrow **New Connection**:



c. Select **Send/Receive** under the **Application Protocol Left** section and **Rockwell CompactLogix/ControlLogix** under the **Application Protocol Right** section. Then click **[OK]**.

S New Connection	A DESCRIPTION OF TAXABLE	? X
Connection Name S7-send-to-CLX		ОК
Expert mode		
Transport Protocol Left	Transport Protocol Right	Cancel
TCP/IP	TCP/IP	Help
🔘 н1	🔘 н1	
Application Protocol Left	Application Protocol Right	
Send/Receive	Send/Receive	
Siemens S5 AP	Siemens S <u>5</u> AP	
Siemens SZ	Siemens SZ	
Modbus TCP	O Modbus TCP	
Rockwell CompactLogix / ControlLogix	Rockwell CompactLogix / ControlLogix	
Rockwell Slc / Plc5	Rockwell Slc / Plc5	
Mitsubishi Melsec-Q	Mitsubishi Melsec-Q	

- d. Since we are only using port 1 (Eth 1), we will leave the **Number of Adapter** at **1**.
- e. On the **Right side** (ControlLogix), select **Client (Active)** type and enter the IP address of the ControlLogix PLC.
- f. On the Left side under Special Settings, check Rfc1006.
- g. Click on the [RFC1006 TSAPs] button.

Connection Name S7-s	end-to-CLX 🔽 Connecti	ion is active Raw <-> ControlLogix Forward clock over TCP/IP	ОК
Left side	L - Chun	Right side	Cancel
Destination IP Address	0.0.0.0	Destination IP Address 192,168,10.8	Help
Destination Port	2000	Destination Port 44818	
Туре	Protocol	Type Protocol	
Client (Active)	TCP (with Ack)	Client (Active) TCP (with Ack)	
Server (Passive)	O UDP (without Ack)	Server (Passive) UDP (without Ack)	
Special Settings		Special Settings	
PLC Header	Life Data Acks 🔽 Rfc1006	PLC Header 🗸 Life Data Acks 🕅 Rfc1006	
Special Settings	REC1006 TSAPs	Special Settings REC1006 TSAPs	



h. Enter the HEX values of the TSAPs created by the Open Communication Wizard in chapter 6 step 17.





Note

Own TSAP = Partner B TSAP

Dest TSAP = Partner A TSAP

i. After clicking **[OK]** in the **Echochange Connection Network** window, enter the ControlLogix CPU slot. In our case it's 2.

		ОК
Connection Name	S7-send-to-CLX	Cancel
CPU Slot No	2	Liele

j. Enter the ControlLogix tag name created in chapter 2 step 1 (used to receive data from the S7-300) and enable **Raw Data Endian Convert**.



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 Support: support.automation@softing.com