Tech Note 355 How to use DASSIDirect DAServer Together with Siemens S7-200 PLC

All Tech Notes and KBCD documents and software are provided "as is" without warranty of any kind. See the Terms of Use for more information.

Topic#: 002006 Created: March 2004

Introduction

This technote will go over the steps necessary to setup communication between the new DASSIDirect DAServer and a Siemens S7-200 PLC.

The TechNote consists of 4 parts:

- Configure the S7-200 PLC
- Configure the DAServer
- Test the Communication
- Memory Areas in the S7-200 PLCs and How to Access Them Via DASSIDirect

Configure the S7-200 PLC

1. Start Step7-Micro/Win 32. Create a new project or load your existing one (Figure 1 below):



Figure 1: Step 7 - Micro/WIN32 - Project1 Dialog Box.

- 2. Click on **Tools** and then select **Ethernet Wizard**.
- 3. Step through the wizard:



Figure 2: Ethernet Wizard.

4. Click Yes when asked to use symbolic addressing. Otherwise the wizard cannot continue:

STEP 7-N	Micro/WIN 32
?	To use the wizard, the project must be compiled, and be in Symbolic Addressing Mode. Compile and use symbols now?
	la Nein

Figure 3: Click Yes.

5. Enter the module position of the CP243-1.

If you are uncertain about the position, click the button **Read Modules**. Otherwise you can enter the number directly:

Ethernet Wizard	
	This wizard will help you define the parameters for the CP243-1 Ethemet module. The wizard will then place this configuration in your project. Specify Module Position To configure the module, specify the module's position relative to the PLC. Click 'Read Modules' to search for installed CP243-1 Ethemet modules.
	Module Position Bead Modules Position Module ID
	<prev next=""> Cancel</prev>

Figure 4: Specify Module Position Dialog Box.

6. Enter the IP configuration of your CP243-1.

Especially during the startup phase of the project, I would recommend not to use a BOOTP server.

Let the module detect the connection type (Figure 5 below):

	Module Address
100****01	Prease select the address to assign to this CP243-1 module. Byour network provides a BUO IP server (a service that will automatically assign IP addresses at startup), you may choose to have an IP address automatically assigned.
E.	IP Address: 10 . 49 .129 . 35
	Subnet Mask: 255.255.0.0
10111 10 10	Gateway Address: 10 . 49 .128 . 1
1	T Allow the BOOTP server to automatically assign an IP address for the module.
	Module Connection Type
	Specify the communications connection type for this module.
	Auto Detect Communications

Figure 5: Module Address Configuration.

7. Enter the numbers of connections you want to configure for the CP243-1.

Default value is **0**, which would not allow communication. In this example I will use 2 connections (Figure 6 below):

Ethernet Wizard		×
100*** 01	Module Command Byte Determine the Q-address by counting the output bytes used by any I/O modules attached to the PLC before the CP243-1 module.	
TI	OB 2	
	many connections you wish to configure for this module.	
	Click 'Next>' to edit the connections for this configuration.	
	«Prev Next» Cancel	

Figure 6: 2 Connections.

Now you have to configure the connections. Connection 0 (Figure 7 below) will accept all incoming client requests:

Connection 0 (2 connections requested) C This is a Client Connection: Client connections reque server	est data transfers between the local PLC and a remote
Local Properties (Server) TSAP 10.00 This server will connection with an Operator Panel (OP). Accept all connection requests. Accept connection requests from the following clients only.	TSAP 10.00
Enable the Keep Alive function for this connection. Please specify a symbolic name for this client connect symbolically when initiating data transfers with the remo	ion. Your program can reference this connection ofe server.

Figure 7: Connection Configuration.

- 8. Always select **This is a Server Connection**. I recommend using the default TSAP's as suggested by Step7-Micro/Win32.
- 9. Click Next Connection (or Prev. Connection if available) to step through all the connections to configure them.

Note: If you plan to use such a connection, be sure that only one client tries to connect to the PLC via this connection at the same time. All other connection tries will be rejected.

Connection 1 accepts only requests from the specified client:

Configure Connections	×
You have requested 2 connection(s). For each connection, specify whether the connection should act as a client or server, and configure its associated properties.	
Connection 1 (2 connections requested) This is a Client Connection: Client connections request data transfers between the local PLC and a remote server This is a Server Connection: Servers respond to connection requests from remote clients. Local Properties (Server) TSAP 11.00 Remote Properties (Client) TSAP 11.00 Remote Properties (Client) TSAP 10.00	
Enable the Keep Alive function for this connection. Please specify a symbolic name for this client connection. Your program can reference this connection symbolically when initiating data transfers with the remote server.	
< Prev Connection Next Connection >]
OK Cancel]

Figure 8: Specified Client Requests.

10. If all connections are configured click **OK**.

During the startup phase of a project I would recommend not to use a CRC protection:

Ethernet Wizard	×
The subscription of the su	CRC Protection
100-01	The wizard can generate a CRC to help protect the module configuration from unintentional memory overwrites. However, this protection will also prevent your program from making modifications to the configuration at run time.
TI.	Yes, generate CRC protection for this configuration in the data block. (* No, do not generate CRC protection for this configuration.)
The second second	Keep Alive Interval When connected with a remote communications partner, or when communicating with STEP 7-Micro/WIN, the CP243-1 module can ensure the connection on a timed interval. Specify the interval for the Keep Alive function, in seconds.
	30 × sec.
	(Prev Next) Cencel

Figure 9: No CRC Protection.

The wizard now needs a range in the V-Memory where to store this information. Step7-Micro/Win32 will suggest a valid range.

11. Click on Suggest Address if you have planned to use this range for something else.

In this case Step7-Micro/Win32 will suggest another free range that has the correct size to hold this data:

Ethernet Wizard	Allocate Memory for Configuration The configuration block for this module requires 24 bytes of V-Memory. With the options you have chosen, the total size of the configuration is 159 bytes. Please specify a starting address where the configuration will be placed in the Data Block.
	The wizard can suggest an address that represents an unused block of V-memory of the correct size. Suggest Address VB0 through VB158
	(Prev Next) Cancel

Figure 10: Store Memory Allocation for Address.

12. Click Next.

The program now has enough information:

Ethernet Wizard		×
	The Ethernet Wizard will now generate the project components for your selected configuration, and make that code evailable for use by your program. Your requested configuration consists of the following project components: Subroutine "ETH0_CTRL" Subroutine "ETH0_CFG" The module configuration will be placed at (VB0 - VB158) in the Data Block Call the initialization and control subroutine "ETH0_CTRL" every scan. The CP243-1 module configuration must be downloaded to the PLC before use.	×
	<prev finish<="" td=""><td>Cancel</td></prev>	Cancel

Figure 11: Generate Project Components.

13. Click **Finish** to complete the configuration:

Finish		×
?	Complete the	Wizard Configuration?
	ļa	Nein

Figure 12: Complete the Wizard Configuration.

14. Click Yes.

In Step7-Micro/Win32 you should now see something like the following figure:

🗱 STEP 7-Mkro/WIN 32 - Project1 - [Data Block]
🖞 Ele Edit Yew BLC Debug Icols Windows Help
0 ●◎ 4 ● 2 × ● 8 ○ ● 1 1 1 1 ■ 1 1 1 ■ ● ● 第四目 # 5 ● 6 6 6
2=2= 戸沼田 ▲多多多 開設 00 ユュ ← → 3+ ○ 0
View Specific F
Tools
Modem Exponsion OF ETH // CP243-1 Ethernet Module Configuration block. Generated by the Ether
Webd LGETH VB0 'CP243' // Module ID for CP243-1 Ethernet module at posit 8 G Symbol VW5 16#006C // Length of CD8
EffemetWizerd System VB10 16#001 // Length of NVB System VB10 16#00 // Configuration Data Version VB10 16#00 // Project Configuration Version
Communications , User Configured
VD15 1600A318123 // TV Address for Hodule (10.49.129.35).
(P/T_0 (P/T0)
Block Size = 0 (bytes), 0 errors
Ready Lin 1, Col 0 INS

Figure 13: Step7_Micro/Win32 Window.

Step7-Micro/Win32 has created some new entries in the V-Memory, starting at the address as specified during the setup of the CP243-1.

15. Download the configuration to the PLC:



Figure 14: Download the Configuration.

16. Select all options and click **OK** (Figure 15 below):



Figure 15: Download Options.

17. Set the PLC to STOP mode in order to be able to download a new configuration:

STEP 7-	Micro/WIN	132	×
Place the PLC in STOP mode			
	ОК	Abbrechen	

Figure 16: PLC STOP Mode.

18. After the download don't forget to set the PLC to **RUN** mode again. This will not be done automatically:



Figure 17: PLC RUN Mode.

The PLC side is now configured.

Configure the DAServer

Configuration of the DAServer is quite easy.

Since we have created two connections in the PLC, we will also create two connections in DASSIDirect DAServer.

Connection 0 will correspond to the connection as created in Figure 7:

🖉 SMC - [ArchestrA System Management	Console (PETERC4QA2003)\DAServer Manager\Default Group\L	oca 💶 🗆 🗙
Eile Action Yiew Help		
⇔ → 🗈 🖬 🗙 😭		
hestrA System Management Console (PETERC- Galaxy Database Manager DAServer Manager Default Group Default Group Def	Node Type: S7Cp Delimiter: S7200_allowall Parameters Device Groups Processor Type: \$7 FLC Network Address: 10.49.129.35 Local TSAP 10	
Log Viewer Platform Manager	Remote TSAP Remote Rack No: 0 Remote Slot No: 0 Connection Resource: 10 Reply Timeout: 15000 ms	
	ConnectionTimeout: 30000 ms	•

Figure 18: DASSIDirect Connection 0.

The Remote TSAP in DASSIDirect must be the Local TSAP in the PLC configuration, and vice versa.

Connection 1 shown in Figure 19 (below) will correspond to the connection as created in Figure 8:

2 SML - [ArchestrA System Manageme	nt Console (PETERC4QA2003)\DAServer Manager\Default Group\Lo	(a <u>- D X</u>
Elle Action View Help		
⇔ ⇒ 🗈 🖬 🗙 🔮		
hestrA System Management Console (PETERC Galaxy Database Manager DAServer Manager Default Group Default Group Configuration Conf	Node Type: S7Cp Delimiter: S7200_spec Parameters Device Groups Device Items Processor Type: S7 PLC Network Address: 10.49.129.35 Local TSAP 10 . Remote TSAP Remote Rack No: 0 Remote Slot No: 0 .	
₹►	Connection Resource: 11 Reply Timeout: 15000 ms ConnectionTimeout: 30000 ms Alarms & Events	

Figure 19: DASSIDirect Connection 1.

The single parts of the remote TSAPs in DASSIDirect are called **Remote Rack No**, **Remote Slot No** and **Connection Resource**. This naming convention was meant for S7-300 and S7-400 PLC's.

For S7-200 PLC's it is a little bit different. The **Remote Slot No** (which is the remote slot number of the CP, not the CPU) in the DASSIDirect configuration corresponds to the **Module Position** as seen in **Figure 4**. The **Remote Rack No** should always be **0**.

Another way to figure out the correct values for the remote TSAP's in DASSIDirect is described in the following information:

Figure 20 (below) shows you how to figure out the meaning of the single parts of the remote TSAP in DASSIDirect from a local TSAP configuration in the PLC as seen in **Figure 7** or **Figure 8** (The TSAP in Figure 20 do not correspond to the TSAP's as used in the above example. I wanted to have different numbers for Rack and Slot in Figure 20!).

TSAP:	02.	03	(hexad	lezi	mal)	
Conn. Res.: OS			Rack	0	Slot	3

Figure	20:	Remote	TSAP.
--------	-----	--------	-------

The TSAP in Figure 20 would correspond to the remote TSAP value in Figure 7 or Figure 8.

In this example the value for **Connection Resource** in DASSIDirect would be **02**, the value for the **Remote Rack No** would be **0** and the **Remote Slot No** would be **3**.

Now create your device group(s), if necessary.

For a DDE/SuiteLink connection you need to have at least one device group per connection, for OPC it is not necessary. For to test the communication as described below, please enter a device group called **\$7200**.

The connection between DASSIDirect and the S7-200 PLC should now be ready to test.

Test the Communication

In previous TechNotes we used wwclient for testing. This tool will no longer be installed when installing ArchestrA (A²) products like InTouch 8.0 or any DAServer. If you have InTouch 7.11 installed you will find this tool under Program Files/Wonderware FactorySuite/Common and you can use this tool.

- Create a simple InTouch application with one tag.
 Select Special/Access Names from the main menu.
- 3. Select Add.

The Add Access Name dialog box appears:

dd Access Name		
Access S7-200		OK
Node Name:		Cancel
Application Name:		
DASSIDirect		
DASSIDirect		
DASSIDirect <u>Topic Name:</u> [S7200]		-
DASSIDirect <u>Topic Name:</u> [S7200] Which grotocol to use		
DASSIDirect <u>Iopic Name:</u> [S7200] Which protocol to use © DDE	C SuiteLink	
DASSIDirect Iopic Name: S7200 Which protocol to use ODE When to advise server	C SuiteLink	

Figure 21: Add Access Name.

- 4. Enter a meaningful name in the Access field.
- 5. Leave the Node Name blank if you have InTouch and DASSIDirect on the same node.

Otherwise enter the nodename of the PC where DASSIDirect resides.

The Topic Name must match the Device Group name as configured in DASSIDirect.

6. Select Special/Tagname Dictionary and select New.

Tagname Dictionary		×		
C Main C Details C Alarms C Details & Alarms C Members				
New Restore Delete Save <<	Select > Cancel C	llose		
Tagname: MB1	Lype: I/O Integer			
Group: \$System	C Read only C Read Write			
Comment: AccessLevel				
□ Log Data □ Log Events □	Retentive Value 🔲 Retentive Para	meters		
Initial ⊻alue: 0	Mig EU: 0	Mag EU: 255		
Deadband: 0	Min Ra <u>w</u> . 0	Max Raw: 255		
Eng Units:	Log Deadband: 0	Conversion ← Linear ← Square Root		
Access Name: \$7-200				
Item: MB1		Use Tagname as Item Name		

Figure 22: New Tagname.

- 7. Select Type: I/O Integer and the access name as created in the previous steps.
- 8. Use the item **MB1** because it always exists in all S7-200 PLCs.

Use this tag in the window you just created.

- 9. Activate DASSIDirect in the System Management Console (SMC).
- 10. Start WindowViewer.
- 11. Open the Diagnostic in the SMC to verify the communication:



Figure 23: Check Communication from the SMC.

If you see a Time value that is changing, and a Client Quality of 00C0, your communication is fine.

Memory Areas in the S7-200 PLCs and How to Access Them Via DASSIDirect

S7-200 PLCs have the following memory areas:

Memory area	Mnemonic in PLC	Item in DASSIDirect
Flags (Merker)	M, MB, MW, MD	Flag item syntax
V-Memory	V, VB, VW, VD	Datablock item syntax (only DB1!)
Timer	Т	Timer item syntax
Counter	С	Counter item syntax
Input	I, IB, IW, ID	Input item syntax
Output	Q, QB, QW, QD	Output item syntax

For more information about the detailed item syntax for DASSIDirect, please refer to the DASSIDirect online documentation.

Note: The following DASSIDirect items are NOT supported when connecting to a S7-200 PLC:

- Peripheral Input Bytes
- Peripheral Output Bytes
- Block Items
- Alarms & Events

Examples: If you want to read the memory area as seen in **Figure 13**, you have (amongst other things) the following possibilities:

V-Memory	Value in PLC DASSIDirect Ite	m Value in	
----------	------------------------------	------------	--

Address		Syntax	Client	
VB0	'CP243'	DB1,S0,5	'CP243'	String with 5 characters, starting at byte 0
VW5	16#006C	DB1,W5	108	Unsigned Integer (2 bytes), starting at byte 5
VW7	16#0014	DB1,W7	20	Unsigned Integer (2 bytes), starting at byte 7
VB9	16#01	DB1,B9	1	Unsigned Integer (1 byte), starting at byte 9
VB10	16#00	DB1,B10	0	Unsigned Integer (1 byte), starting at byte 10
VW11	16#0000	DB1,W11	0	Unsigned Integer (2 bytes), starting at byte 11
VW13	16#AC04	DB1,W13 or DB1,X13.0 to DB1,X14.7	44036	Unsigned Integer (2 bytes), starting at byte 13 since this is some kind of bit-pattern, where each bit represents one special setting, you can also access the single bits directly.
VD15	16#0A318123	DB1,D15 or DB1,B15 DB1,B16 DB1,B17 DB1,B18	171016483 10 49 129 35	Unsigned Integer (4 bytes), starting at byte 15 since this is an IP address, it's probably better to access this value as four independent bytes, and concatenate them to a string like 10.49.129.35 in InTouch.

P. Crämer

The Tech Note is published occasionally by Wonderware Technical Support. Editors: Sabrina Abercromby and Kelly Hauser; Publisher: Wonderware Corporation, 100 Technology Drive, Irvine CA 92618. There is also technical information on our software products at www.wonderware.com/support/mmi, our WonderFax fax-on-demand system at (949) 450-5050, and the Comprehensive Support Knowledge Base.